Renewable Energy Focus 48 (2024) 100520

Contents lists available at ScienceDirect

Renewable Energy Focus

journal homepage: www.journals.elsevier.com/renewable-energy-focus

Research paper

Federated transfer learning with orchard-optimized Conv-SGRU: A novel approach to secure and accurate photovoltaic power forecasting

Syed Muhammad Salman Bukhari [a], Syed Kumayl Raza Moosavi [b], Muhammad Hamza Zafar [c], Majad Mansoor [e,f], Hassan Mohyuddin [a], Syed Sajid Ullah [d], Roobaea Alroobaea [h], Filippo Sanfilippo [c,g,*]

[a] *Department of Electrical Engineering, Capital University of Science and Technology, Islamabad, 44000, Pakistan*
[b] *SEECS, National University of Sciences and Technology, Islamabad, 44000, Pakistan*
[c] *Department of Engineering Sciences, University of Agder, Grimstad, 4879, Norway*
[d] *Department of Information and Communication Technology, University of Agder, Grimstad, 4879, Norway*
[e] *Department of Automation, University of Science and Technology of China, 28796, China*
[f] *MIT Scale Network, Ningbo China Institute for Supply Chain Innovation, Ningbo, 315000, China*
[g] *Department of Software Engineering, Kaunas University of Technology, Kaunas, 51368, Lithuania*
[h] *Department of Computer Science, College of Computers and Information Technology, Taif University, Taif, 11099, Saudi Arabia*

ARTICLE INFO

Keywords:
Privacy-preserving
FL
TL
PV power forecasting
Deep learning

ABSTRACT

Accurate photovoltaic (PV) power forecasting is pivotal for optimizing the integration of RES into the grid and guaranteeing proficient energy management. Concurrently, the sensitive nature of data obtained from individual PV systems underscores paramount concerns regarding data privacy and security. In this manuscript, we introduce an innovative approach for PV power forecasting that addresses these concerns, deploying federated learning (FL) combined with TL. This is orchestrated via a hybrid deep learning model, denominated as Federated transfer learning (TL) Convolutional Neural Network with Stacked Gated Recurrent Unit (FL-TL-Conv-SGRU). To optimize the performance of the Conv-SGRU model, we employ the OA for hyperparameter tuning, a novel bio-inspired technique inspired by orchard gardening practices. This algorithm presents a distinctive interplay between exploration and exploitation in the hyperparameter space, potentially elevating the model's performance. Our exposition covers eight disparate datasets from PV systems, which are judiciously split into two cohorts, safeguarding data privacy. Through the prism of FL, we ensure data security by orchestratively training the Conv-SGRU model over distributed datasets. This strategy allows tapping into the shared wisdom across the datasets, all the while ascertaining individual data remains localized, boosting model generalization and predictive prowess. Additionally, TL is invoked to benefit from pre-trained feature representations, facilitating effective knowledge transmission across diverse PV setups with unique characteristics and locales. The put-forth FL-TL-Conv-SGRU design amalgamates the essence of FL, TL, convolutional neural networks, and stacked gated recurrent units. This ensemble aids in deciphering spatial–temporal intricacies intrinsic to PV power generation. Through empirical analyses, we evince that our FL-TL-Conv-SGRU model transcends conventional forecasting paradigms, emphasizing its adeptness in delivering meticulous forecasts over a range of PV installations. Our results accentuate the bifurcated importance of the federated TL framework: a capability for collaborative training with an unwavering commitment to data privacy, and a proficiency in exploiting decentralized data. This strategy is particularly salient given the shifting regulatory milieu centered on data safeguarding and confidentiality. As we transition towards a world more reliant on renewable energy, our proposed stratagem promises to be a cornerstone for efficient, sustainable energy management, heralding a future replete with green energy.

* Corresponding author at: Department of Engineering Sciences, University of Agder, Grimstad, 4879, Norway.
E-mail address: filippo.sanfilippo@uia.no (F. Sanfilippo).

https://doi.org/10.1016/j.ref.2023.100520
Received 26 September 2023; Received in revised form 8 November 2023; Accepted 23 November 2023
Available online 25 November 2023

Nomenclature

Abbreviation

RES	Renewable Energy Sources
CNN	Convlutional Neural Networks
GRU	Gated Recurrent Unit
OA	Orchard Algorithm
NMSE	Normalized Mean Squared Error
MAE	Mean Absolute Error
RE	Relative Error
SCADA	Supervisory Control and Data Acquisition
SGD	Stochastic Gradient Descent
GDPR	General Data Protection Regulation
SVR	Support Vector Regression
ARIMA	Auto-Regressive Integrated Moving Average
ReLU	Rectified Linear Unit
MLP	MultiLayer Perceptron
RNN	Recurrent Neural Network
LSTM	Long Short Term Memory
DNN	Deep Neural Network
FL	Federated Learning
RMSE	Root Mean Squared Error
TL	Transfer Learning
ANN	Artificial Neural Network

Symbols

R^2	R square
a_n	Update Gate for n layer
h_n	Hidden State for n layer
$\hat{f}(s_i)$	Normalized Objective Function Value
s_i	Seedling i
r_n	Reset Gate for n layer
$\hat{g}(s_i)$	Growth Rate for seedling i
$\nabla\mathcal{L}\left(a_t\right)$	Batch gradient
C_t	Customers
F	Filters
x_o^l	Output of Convolutional Layer
$f(.)$	Activation function
$k_{io,fl}^1$	Kernel filter
$\bar{T}$	Average value of the true output
$\bar{P}$	Average value of the expected output

Subscripts

n	layer number
i	No. of the seedling from the population

Greek letters

θ	Global model Parameters
α	Weight of the fitness function
β	Weight of the growth rate
η	Learning rate

1. Introduction

How to produce a clean and sustainable energy supply has emerged as a crucial topic that requires attention due to the simultaneous pressures of the steady depletion of non-RES and the more noticeable ecological and environmental challenges [1]. Also, RES has drawn more attention recently as a result of the rising energy demand. One of the most promising sources of renewable energy is photovoltaic (PV) electricity. International Energy Agency (IEA) [2] data indicates that by 2018, the yearly average expansion rate for photovoltaic (PV) energy was at 36.5%. Projections suggest that by the year 2022, there will be an approximate increase of 120 gigawatts in global PV infrastructure. However, a PV system's power output fluctuates and is intermittent since weather patterns are always changing. The fluctuation makes the PV integration difficult and poses possible dangers to a power grid system. Therefore, precise PV power generation forecasting is crucial since it may significantly help with electricity and power grid scheduling. For predicting PV power, a great deal of research has been conducted. Physical performance models were first commonly taken into consideration. These models simply use Numerical-Weather-Prediction (NWP) data and module settings instead of historical generation data. PV power forecasting also uses statistical techniques like SVR [3] and ARIMA [4]. DNN has emerged in recent years as a potential prediction-making method. Deep learning techniques may effectively uncover nonlinear correlations between input variables by structuring many hidden network layers. CNN [5], RNN [6], and LSTM are a few of the DNN approaches utilized to boost the performance of PV forecasting. PV plants who want to adopt this strategy must first gather a large amount of historical data since DNNs frequently suffer from overfitting without enough training data. However, sharing historical data is problematic in light of recently passed data security and privacy rules like the GDPR [7] and the Health Insurance Portability and Accountability Act (HIPAA) [8]. This information is often important for the power industry.

Therefore, gathering significant quantities of data from several plants to build a decent model is frequently not possible. Federated learning (FL) provides an effective solution to the privacy challenges associated with data collection. It allows edge devices to perform local training, and then only the model updates are sent to a centralized server for the aggregation of a global model. The whole training and running procedure does not include the transmission of raw training data between devices, potentially protecting privacy. The Federated Averaging (FedAvg) optimization technique is typically used for update aggregation [9]. A central server sends the global model to a group of clients that are chosen at random during a normal FedAvg iteration, and it then waits for committed results. The client devices communicate trained weights back to the central server after performing local changes (using gradient descent) over many iterations. An optimized model can subsequently be produced when the central server averages out all the individual weights to refresh the overarching global model. To optimize model configurations and better cater to the statistical nuances and facility-specific details inherent in diverse power facilities, the OA offers a promising avenue. Traditional FL aims to construct a universal model by amalgamating data from multiple sources. While this broad approach has its merits, it occasionally misses out on addressing the unique variations and intricacies of individual power facilities. This sometimes leads to models that might not perform optimally for certain setups. Introducing TL into this mix can offer a significant enhancement. By embedding transfer learning (TL) within the FL paradigm, models trained globally can be meticulously fine-tuned according to the specific traits of local datasets. This means the resultant model is more adept at addressing site-specific variations, improving its efficacy for individual scenarios.

However, integrating the principles of the OA provides an additional layer of optimization. OA, with its growth, screening, and grafting procedures, facilitates an efficient navigation of the model's hyperparameter space, ensuring a more tailored fit for the task. Our research dives deep into this innovative amalgamation of FL, TL, and OA. The objective is to effectively address the challenges posed by data heterogeneity, which conventional FL models occasionally grapple with (see Fig. 1).

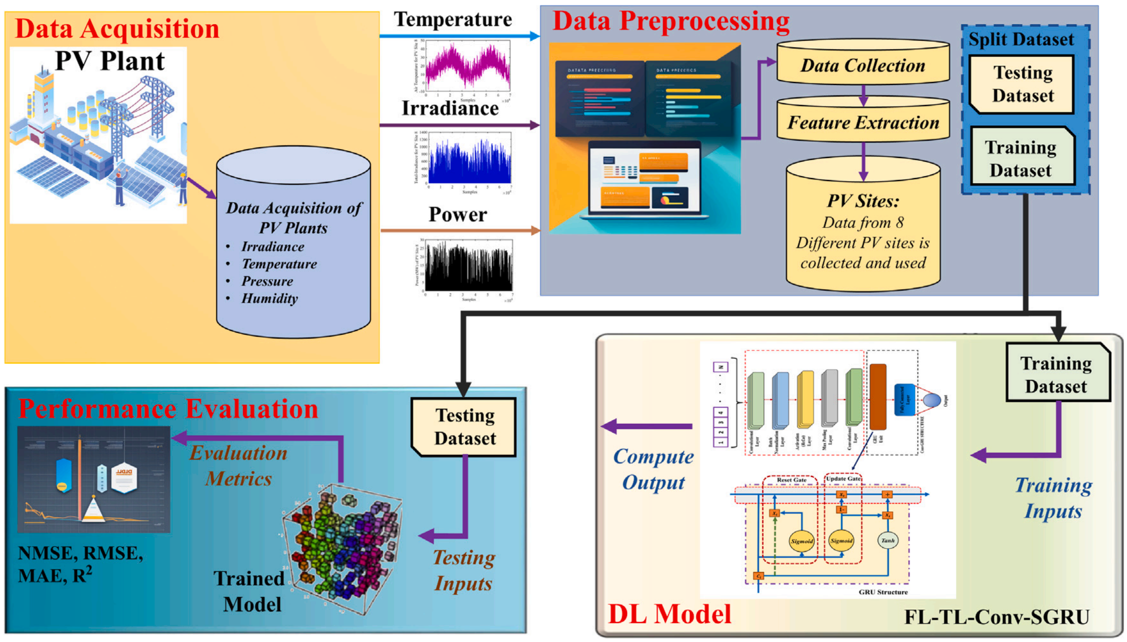

Fig. 1. Flow diagram of Proposed technique.

1.1. Contributions and paper organization

In this study, we introduce a novel approach for short-term PV power forecasting that harnesses the synergistic capabilities of the CNN-Stacked GRU (CNN-SGRU) model, a framework that intertwines TL with an FL environment (FL), and the OA. Our method exploits edge computing to elevate forecasting precision, all the while addressing the challenges posed by data heterogeneity and privacy. The strategy we have employed involves partitioning the data from two distinct groups of PV sites for training within the FL context. Initially, the model is pre-trained using data from the first group. Crucially, the OA is utilized to optimize the model parameters during this phase, ensuring enhanced convergence and accuracy. The consolidated weights derived from this phase are preserved, readying them for subsequent deployment on the second data group, epitomizing the TL paradigm.

In this study, we make the following significant contributions:

1. **Innovative Forecasting Methodology:** We present an advanced approach for photovoltaic (PV) power forecasting by harmoniously integrating FL with TL. This strategy not only addresses the challenges of data privacy and security but also advances the efficient integration of renewable energy into the grid.
2. **Hybrid Deep Learning Architecture:** We unveil the CNN integrated with a Stacked GRU (Conv-SGRU). This hybrid architecture adeptly discerns both spatial and temporal intricacies present in PV system datasets. By design, the model can elucidate intricate spatiotemporal dynamics characteristic of PV power generation.
3. **FL with Orchard-Optimized Tuning:** We deploy the OA for hyperparameter tuning of the Conv-SGRU model, ensuring its optimal performance. Alongside this, our FL framework provides collaborative model training across distributed datasets, all the while upholding the sanctity of data privacy.
4. **Robust Empirical Assessment:** Through meticulous empirical evaluations, we substantiate the Conv-SGRU model's superiority over traditional forecasting techniques. Our results spotlight its consistent accuracy and reliability, even when confronted with PV installations that span diverse geographies and characteristics.
5. **Contribution to Secure Energy Management:** In resonance with the shifting paradigms around data privacy and protection regulations, our research paves the way for both sustainable and secure energy management systems. By doing so, we enhance the capabilities of integrating renewable sources into the global energy grid, marking a stride towards a more sustainable energy future.

The paper is organized as follows: In Section 2, we provide a comprehensive review of relevant research about the prediction of PV power and TL. Section 3 outlines the problem statement, introduces the proposed CNN-SGRU model, and presents the FL framework incorporating TL and OA. Our choice of the dataset for this study is expounded upon in Section 4, along with a detailed account of the experimental methodology and resultant findings. Finally, in Section 5, we offer conclusive remarks and identify promising avenues for future research (see Table 1).

2. Related work

The concept of 'time horizon' in PV power forecasting refers to the interval between actual and anticipated operating times [15]. Time horizons can be broadly categorized into very short-term, short-term, medium-term, and long-term, each serving different operational needs within the power system [16]. For instance, real-time energy dispatching and PV plant monitoring typically rely on very short-term forecasts, ranging from seconds to 30 minutes [17]. Medium-term forecasts, spanning a few hours to days, facilitate the maintenance and operation of power systems incorporating PV energy. Long-term forecasts, covering days to months, are often used for seasonal planning and new PV plant constructions. Notably, much of the existing research focuses on extremely short-term and short-term forecasts due to their critical role in power scheduling and PV transaction planning. This paper primarily centers on short-term forecasting.

From a modeling perspective, PV power forecasting techniques can be grouped into three main categories: physical, statistical, and ANNs. Physical methods aim to develop mathematical models grounded in the fundamental principles of PV power generation. These models often consider a range of variables, from internal characteristics of the PV plant – such as panel installation angle and conversion efficiency –

Table 1
Literature review of FL-based power forecasting.

Ref.	Year	Technique	Summary	Privacy preserved	Centralized	TL
[10]	2020	FL based on DL using VBI	A federated probabilistic forecasting scheme for solar irradiation that uses deep learning and variational Bayesian inference (VBI). The approach ensures data privacy as training data remain on local IoT devices, and only forecasting models are shared.	✓	x	x
[11]	2021	FL	A PV power prediction model using FL to balance data sharing and privacy protection.	✓	x	x
[12]	2022	DFA	The paper presents a solution to the challenge of residential-level short-term load forecasting (STLF) in light of privacy concerns and the non-identical and independent distribution (non-IID) of data from different houses.	✓	x	x
[13]	2023	MLP-FL-based BTM PV forecasting model	To train a single model using data from many BTM sites, the method for FL power forecasting for PVs presented in this paper employs FL as a decentralized collaborative modeling methodology.	✓	x	x
[14]	2023	FL-LSTM	The unique decentralized collaborative modeling method of FL-based wind energy forecasting is provided in this study. It allows for the training of a single model using data from several wind farms without jeopardizing data security or privacy.	✓	x	x
Our model	2023	OA-FL-TL-Conv-SGRU	In this work, we integrate the OA with a federated TL framework using a stacked convolutional and GRU network. The model is optimized on one PV data group and then transferred for refinement on another.	✓	✓	✓

to external meteorological factors like solar radiation, temperature, and wind speed. Known as analytical models [18], these physical approaches usually incorporate four to five variables and are particularly useful for long-term forecasting [19]. However, the implementation of physical methods is fraught with challenges due to the diversity of plant-specific variables and the frequent absence of detailed information from PV module manufacturers. Statistical methods for PV power forecasting establish relationships between input variables and the forecasted output through statistical algorithms. These models have often demonstrated superior performance compared to their physical counterparts. One prominent statistical model is the Auto-Regressive Moving Average (ARMA), widely adopted for time-series forecasting. For example, Hassanzadeh utilized an enhanced ARMA model to forecast hourly power generation in a 75 kW PV installation [20]. Variations of the ARMA model, such as ARIMA [4] and ARMAX—Auto-Regressive Moving Average with Exogenous Inputs [21], have also gained traction in the field. Other popular statistical techniques include SVR [3] and Exponential Smoothing (ES) [22], further expanding the toolkit for accurate PV power forecasting.

ANNs, and more specifically DNNs, have emerged as a promising area in the realm of PV power forecasting, offering strong capabilities for nonlinear representation and generalization [23]. Among DNNs, the Multilayer Perceptron (MLP) stands out as a commonly used technique. Other variations include the Radial Basis Function Neural Network (RBFNN), which employs radial basis functions as activation functions and has been applied for day-ahead PV power generation forecasting [24]. Yona et al. employed a combination of MLP, RBFNN, and RNN to predict daily power outputs for a 20 kW PV system [23]. Notably, LSTM networks, a specialized form of RNN, have gained prominence. Multiple studies have demonstrated the efficacy of LSTM-based models in accurately forecasting PV power generation on an hourly basis [25]. Overall, empirical evidence suggests that LSTMs tend to outperform various alternative approaches in this domain. While numerous models and strategies have been proposed for PV power forecasting, they often fall short in addressing key issues such as data heterogeneity and privacy concerns. For instance, some existing FL methods for PV power forecasting have primarily focused on decentralized collaborative modeling, leveraging multi-layered perceptrons to ensure data privacy and security [13]. Others have extended FL to deep reinforcement learning for ultra-short-term wind power forecasting, aiming for higher prediction accuracy but still not fully addressing data heterogeneity [26]. Yet another study introduced a semi-asynchronous FL framework for short-term solar output, employing a CNN-LSTM model to improve the framework's effectiveness [27]. In response to the prevalent issues of data heterogeneity and privacy in existing models, we present the OA-FL-TL-Conv-SGRU framework. This innovative solution cohesively merges the OA, FL, and TL. By operating in a decentralized manner via FL, we ensure rigorous data privacy across multiple edge devices. Additionally, the inclusion of TL, bolstered by OA, enables our framework to adeptly handle diverse datasets. Not only does our approach showcase superior forecasting accuracy as gauged by metrics like RMSE, MAE, and MAPE, but it also optimally leverages edge computing resources. With its modular design, it promises ease of updates, establishing itself as a scalable solution for future PV power forecasting challenges. This integration, therefore, offers a comprehensive solution, skillfully balancing top-tier accuracy, data privacy, and effective management of varied datasets

3. Proposed method description

We introduce the OA-assisted FL-TL CNN Stacked GRU (OA-FL-TL-Conv-SGRU) model, a sophisticated approach tailored for the effective analysis and prediction of PV power generation data.

3.1. Federated learning

When training DNN models, FL offers a mechanism to solve data privacy issues [9]. Traditional DNN models rely largely on the availability of large amounts of training data to function well. However, it is frequently impossible to send this data from several devices to a single server due to data privacy concerns. Leveraging the principles of distributed machine learning, FL enables joint model training across multiple devices without compromising data confidentiality. As edge computing gains traction – processing information nearer to its source – edge servers have evolved, creating an optimal environment for implementing FL methods. In the FL process, each participating device trains the model using its localized data, while only the learned parameters are transmitted to a central server for consolidation. With this technique, there is no need to aggregate raw data, potentially raising privacy concerns.

Algorithm 1 Basic FL

Procedure FEDERATED_LEARNING (Server, Clients)
Initialize θ
for each round t in total_rounds **do**
  Select a subset of clients C_t to participate in training
  for each client k in C_t in parallel **do**
    Retrieve θ from server
    Update local model parameters by training on local data with CNN-SGRU
    Compute and send model updates $\Delta\theta_t$ to the server
  end for
  Compute global model update $\Delta\theta$ = average($\Delta\theta_t$ for each k in C_t)
  Update global model parameters $\theta = \theta + \Delta\theta$
end for
return Global model parameters θ
End Procedure

FedAvg, a common FL optimization technique, minimizes communication rounds by mandating that clients conduct many local epochs before interacting with the main server. The algorithm for FedAVG is shown in Algorithm 1 Although FedAvg has been improved over time [28], the heterogeneous nature of edge computing systems makes it difficult to deploy its synchronous method. To deal with the temporary nature of edge nodes and related network challenges, research efforts have been focused on asynchronous aggregation techniques [29]. Additionally, data in FL contexts frequently exhibits statistical heterogeneity, with significant variations in data characteristics and volume amongst devices. This suggests that not all participant data may match the current global model. Personalization approaches, which modify the basic model to match particular data, have thus become an important study area. TL [30], multi-task learning [31], and meta-learning approaches [32] are some of the strategies [33] that largely modify the global model utilizing local input from the client.

Model updates are transmitted between these nodes in a method that protects the privacy of the local data in FL, where the model training process is often dispersed among numerous nodes, each with its local data. A mathematical representation that outlines the actions involved in the iterative updating of the local and global models may be used to formalize this distributed learning configuration. We designate the customers (C), the regional datasets (D), the model parameters (a), and the learning rate (η) using particular notations in this form. Let us examine the specific equations and their interpretation, which form the basis of the FL-SCNN-SGRU model we have suggested (see Fig. 2).

All or a portion of the clients are chosen at the beginning of the training phase, and the most recent global model parameters are distributed to the clients. With gathered local data D_k, C_k does optimization over many iterations, such as adaptive moment estimation (Adam). Updated local model parameters include:

$$a^k_{t+1} \leftarrow a_t - \eta \nabla \mathcal{L}\left(a_t\right), \tag{1}$$

Then updates are transmitted to the server, where safe aggregation is carried out:

$$a_{t+1} \leftarrow w_t - \sum_{k=1}^{K} \frac{n_k}{n} a^k_{t+1}, \tag{2}$$

where $n_k = \left|D_k\right|, n = \left|D_1 \cup \cdots \cup D_k\right|$. The technique is then done once more. The data that has been gathered from the various sensor nodes is pre-processed and distributed to each client manually. When the global model is integrated with the obtained model updates, a new global model is produced and made available to source clients. Algorithm 1 shows the general form of a FL approach where:

- The procedure FEDERATED_LEARNING takes in a Server and a set of clients. θ are the global model parameters initialized on the server.
- A subset of the C_t is picked in each round of the algorithm to take part in the training.
- The global model parameters are fetched from the server by each client in C_t, who then trains their local model on local data, figures out the model's update, and transmits it to the server.
- To update the global model parameters, the server first calculates the global model update by averaging all of the k received from each client in C_t.
- The revised global model parameters are then returned when the procedure is performed for a set number of cycles.

3.2. Convolutional neural network (CNN)

CNN, which was first developed for image processing, has also shown promise in the management of one-dimensional sequential data. 1D-CNNs, which are specialized modifications of the basic 2D CNN design, are used to realize this capacity [34]. In a conventional 1D CNN, activation functions like the Rectified Linear Unit (ReLU) are used to add nonlinearity to the model after layers of convolution and pooling procedures. Dropout layers and normalization strategies are frequently implemented into the design to prevent overfitting. If all other variables stay constant, the fundamental benefit of 1D CNNs is that they require less computing power than their 2D equivalents. Due to this, they are especially well suited for real-time applications and their implementation on cheap, portable devices [35]. They have shown to be particularly useful in specialized applications, such as patient ECG monitoring, civil engineering constructions, time-series forecasting, and high-power electrical circuits and motors when the data exhibits severe signal fluctuations and insufficient labeling. In essence, the primary distinction between 1D CNNs and 2D CNNs is the type of inputs used; 1D CNNs use 1D arrays as input, whereas 2D CNNs use matrices. This change makes 1D CNNs particularly effective at handling sequential data. The following mathematical formulae may be used to explain the processes of a 1D CNN:

$$x^l_{o,fl} = f\left(\sum_{\mathrm{im}} \mathrm{x}^{\mathrm{l}-1}_{\mathrm{i}} * \mathrm{k}^{\mathrm{l}}_{\mathrm{io,fl}} + \mathrm{y}^{\mathrm{l}}\right) \tag{3}$$

$$x^l_o = f\left[\max\left(\sum_{im} x^{l-1}_i\right) + y^l\right] \tag{4}$$

$$x^l_o = f\left(x^{l-1}_i * z^l_{io} + y^l\right), \tag{5}$$

The parameters y and z can be learned. Indeed, the way 1D CNNs integrate many machine-learning tasks into a single, well-rounded process is one of its most intriguing features. To improve the accuracy of classification and regression issues, this holistic strategy combines feature extraction, regression, and classification tasks. The ability to identify patterns and learn high-level characteristics from sequential data, such as time series, has been demonstrated by this approach to be quite successful. Fundamentally, the advantages of 1D CNNs are their computational effectiveness and simplicity. They can effectively handle enormous quantities of data by conducting a succession of 1D convolutions, simply the linear weighted sum of two 1D arrays. Because of the much lower computational cost compared to more complicated designs, 1D CNNs are an appealing option for applications demanding real-time processing or having limited computer resources. Furthermore, both the

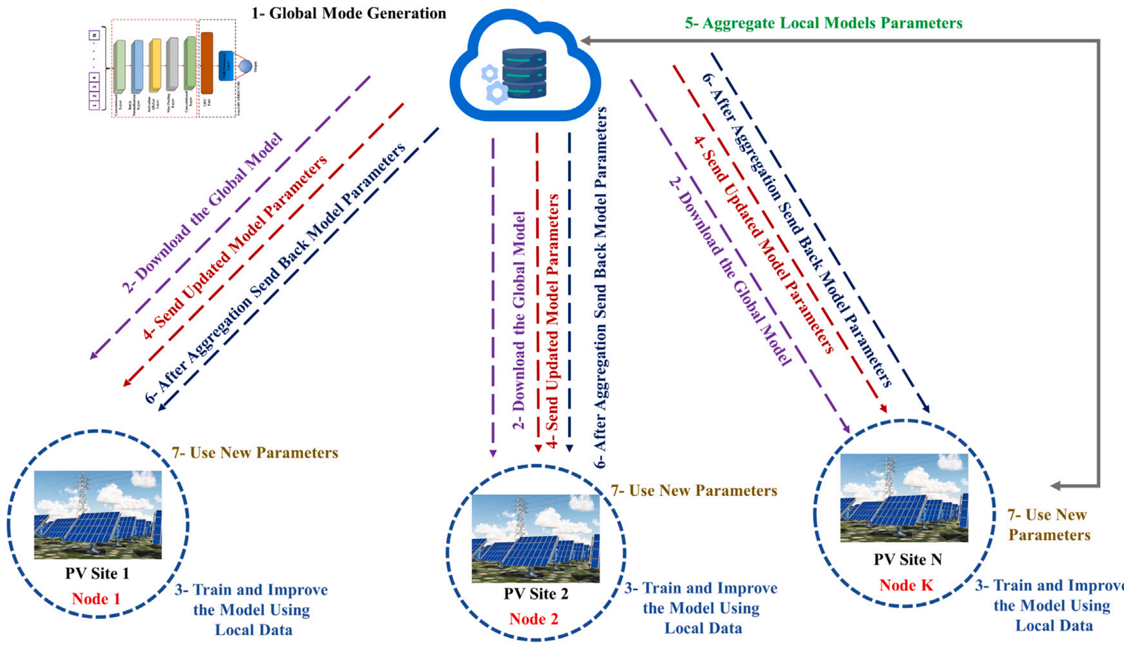

Fig. 2. FL approach for PV Power forecasting.

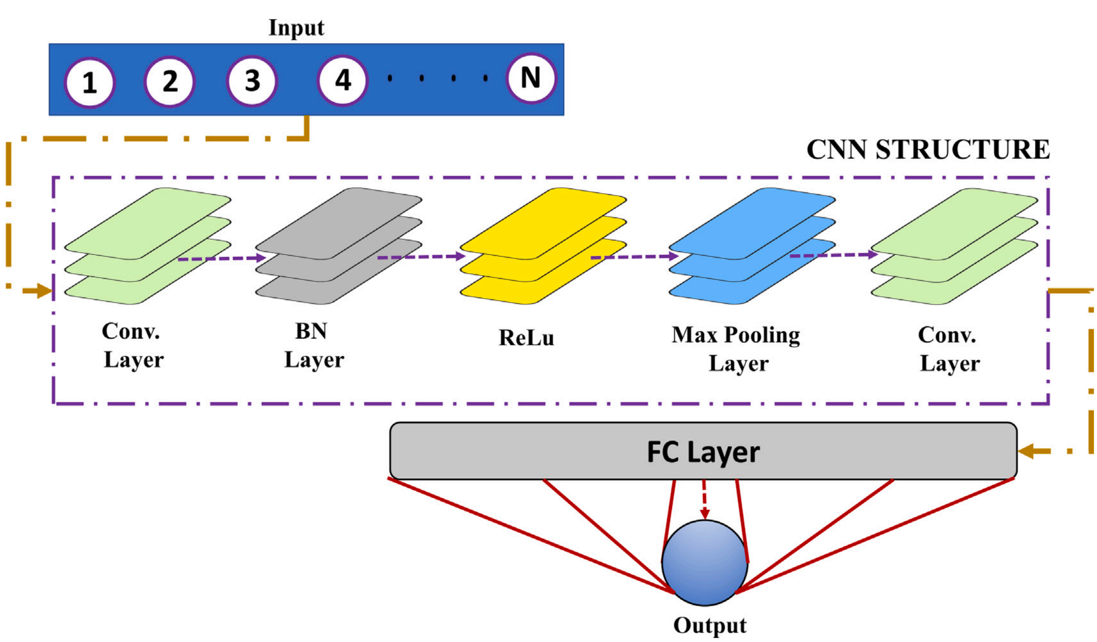

Fig. 3. Structure of CNN.

forward and backpropagation processes may be computed effectively due to the linear nature of convolution operations. This is because these operations may be carried out concurrently across several elements of the input data, considerably accelerating the learning process as a whole. This special quality of 1D CNNs shortens training time and makes it possible to analyze huge datasets quickly, thereby increasing the usefulness and adaptability of this model architecture.

3.3. Gated recurrent unit

A GRU serves as a specific type of RNN, utilizing gates to control and manage the data exchange between different neurons in the network. Due to its capacity to reduce the issues of disappearing and expanding gradients, which frequently trouble standard RNNs, the GRU has seen an increase in utilization. (a_n) and (r_n) are the two different types of gates used by GRUs. These two gates determine how the GRU interprets and stores data over time [36]. The memory unit that enables the reset gate to store and use part of the previously processed information is represented by the $\widetilde{h}_n$ term in Equation. Both of the aforementioned gates lack independent memory units [37]. Eq. (1)'s (a_n) controls how much information from the past must be sent to the future. Given that it determines how much previous information is preserved or forgotten, this is comparable to the forget and input gates in an LSTM.

$$a_n = \sigma\left(w_a x_n, w_{ha} h_{n-1} + b_a\right) \tag{6}$$

According to Eq. (2), (r_n) regulates how much of the previous hidden state goes towards the upcoming hidden state. In essence, the reset gate establishes how much of the past should be remembered and

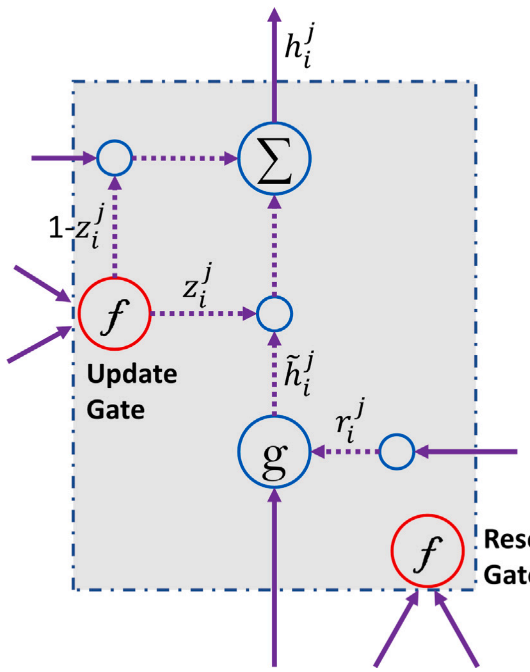

Fig. 4. Structure of GRU.

how much should be discarded.

$$r_n = \sigma\left(w_r x_n, w_{hr} h_{n-1} + b_r\right) \tag{7}$$

Combining these gates enables GRUs to efficiently capture long-term relationships in sequence data, which makes them well-suited for a variety of applications like time-series prediction, natural language processing, and more. Eq. (3) demonstrates how to compute the hidden unit $\tilde{h}_n$ by using a hyperbolic tangent activation function on the weighted sum of the reset gate's output and the prior hidden state.

$$\tilde{h}_n = \tanh\left(w_h \cdot \left[r_n \otimes h_{n-1}, x_n\right] + b_h\right) \tag{8}$$

Finally, as stated in Eq. (4), h_n is calculated. This is the result of adding the weights provided by the update gate to the prior hidden state and the current candidate hidden state.

$$h_n = \left(1 - z_n\right) \otimes h_{n-1} + z_n \otimes \tilde{h}_n \tag{9}$$

3.4. Conv-SGRU

CNN and GRU architectures combine their best qualities in the Conv-SGRU model, which is especially focused on improving the forecasting of PV power generation. Through the use of CNNs and GRUs, it successfully detects both local patterns and temporal relationships in the input data.

The Conv1D layer employs one-dimensional convolution techniques to extract important information from the input sequence, such as weather variables. The GRU layer is then fed these extracted characteristics. By design, the GRU layer examines these properties while properly taking into consideration the temporal order of the input.

The Conv-SGRU model can train quickly and provide precise predictions about PV power generation by taking into account both local and temporal information. This makes it ideally suited for time-series data analysis and power output predictions. Additionally, the Conv-SGRU model is equipped to handle intricate patterns and relationships in data thanks to the combination of both architectures, improving its predictive powers (see Table 2).

3.5. Hyperparameter tuning

Deep learning models, celebrated for their versatility and power, are intricately sensitive to their configuration settings, commonly known as hyperparameters [38]. Unlike weights and biases, hyperparameters are not updated during training but define the model's architecture and its training process [39]. Parameters like learning rate, dropout rate, and the number of neurons in a layer considerably influence a model's performance. Proper hyperparameter selection can be pivotal between a model efficiently achieving high accuracy and one that either overfits or stagnates. Traditional methods like grid search or random search, although well-established, are often computationally burdensome and might not capture the full potential of configurations [40]. This accentuates the quest for superior hyperparameter optimization techniques in the deep learning domain.

Within the broad spectrum of optimization techniques, various algorithms have gained prominence over the years. Gradient-based methods like SGD and its offshoots, such as Adam [41] and RMSprop [42], have been extensively adopted. These techniques dynamically tweak hyperparameters during training based on gradient information. Evolutionary and genetic algorithms also offer population-based optimization strategies [43]. More recently, Bayesian optimization [44] emerged as a probabilistic model-based approach, pinpointing hyperparameters that likely enhance validation performance with minimal evaluations. However, the evolving landscape of optimization research beckons newer techniques, introducing the OA [45].

3.5.1. Introducing the OA

The OA emerges as a novel meta-heuristic optimization method inspired by the meticulous progression and nurturing practices inherent in fruit gardening. Just as a thriving orchard results from systematic care – including activities such as irrigation, fertilization, and grafting – OA similarly navigates the solution space with distinct operations. To provide a clearer and more tangible understanding of how the OA operates, we present a visual representation of its process.

Seedlings Initialization: At the inception of the algorithm, an 'orchard' is initialized by planting 'seedlings', which in the context of hyperparameter tuning, represent potential solutions. These seedlings are typically randomized initial configurations.

Seedling Growth: Analogous to nurturing a plant, each seedling undergoes a growth phase, wherein its neighborhood in the solution space is explored. This can be envisioned as a local search. If this growth leads to a better solution (a more promising hyperparameter configuration), the current seedling is updated. Mathematically, this is often represented by evaluating the objective function within the vicinity of the current solution.

Screening: Periodically, the seedlings are screened or ranked based on their performance. In the context of hyperparameter tuning, this might involve evaluating model performance on a validation set for each seedling's configuration. The performance is often quantified using a combination of the objective function and a growth rate, captured as:

$$F_j = \alpha \cdot \hat{f}_j + \beta \cdot \hat{g}_j$$

Grafting: One of the unique stages in OA is grafting. Here, medium-performing seedlings (or solutions) are enhanced by grafting parts from high-performing seedlings. This cross-over operation combines attributes of good solutions to potentially produce even better ones.

Replacement: Seedlings that continually underperform, analogous to trees that do not bear fruit even after years of care, are replaced with new ones. This introduces fresh genetic material (or new configurations) into the population.

Elitism: To ensure that quality solutions are not lost in subsequent iterations, the top-performing seedlings are retained unchanged in the next generation, a concept known as elitism in evolutionary algorithms.

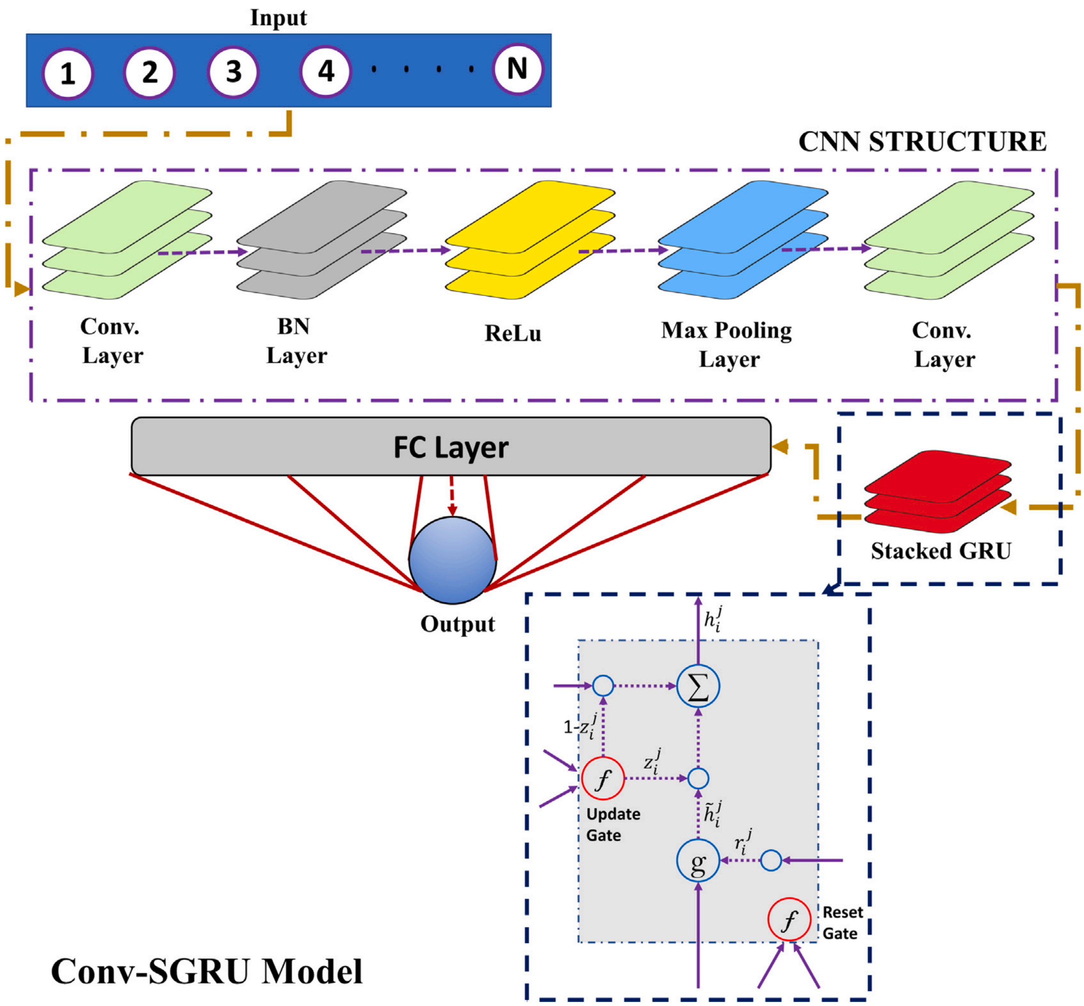

Fig. 5. Detailed Conv-SGRU structure.

Table 2
Range of hyperparameters of Conv-SGRU.

Model name	Hyper parameters	Range (To initialize)
Convolutional layers	No. of Units in 1 Layer Filter size in each Layer Activation	[20–29] [1–7] ['LeakyReLU', 'ReLU', 'Tanh']
GRU layers	No. of Hidden Nodes/Neurons	[10–500]
Dense layer	Nodes	[10–500]
Learning configuration	Learning rate Dropout rate	[10-5–10-1] [0, 0.7]

The algorithm iteratively applies these operations until a stopping criterion is met. This could be a predefined number of iterations, a threshold improvement value, or any other suitable measure. At each iteration, the orchard is refined, moving closer to optimal or near-optimal solutions.

What sets OA apart is its dual ability to explore and exploit the solution space. The seedling growth ensures focused improvement (exploitation) while grafting and replacement processes ensure diverse exploration. This balance, modeled after nature's way of nurturing orchards, offers a promising avenue for efficient hyperparameter tuning, potentially outperforming classical methods in both speed and solution quality.

Emerging from the rich gamut of optimization techniques, the OA presents itself as a bio-inspired meta-heuristic contender. Drawing parallels from the meticulous care of orchards, OA introduces a unique optimization approach. Just as orchard keepers nurture seedlings, ensuring they mature into fruitful trees, OA's premise is to refine potential solutions, guiding them to optimal outcomes.

3.5.2. Mathematical representation of OA

In the realm of the OA, seedlings or solutions are represented by vectors of decision variables. Let us define:

- S as a set of seedlings (solutions).

Table 3
Optimized hyperparameters of proposed DL model using OA.

Model name	Hyper parameters	Optimized using OA
Convolutional layers	No. of Units in 1st Layer Layers	150
	Filter size in each Layer	3
	Activation	'ReLU'
GRU layers	No. of Hidden Nodes/Neurons in 1st Layer	150
	No. Hidden nodes in 2nd layer	100
Dense layer	Nodes	100
Learning configuration	Learning rate	10^{-2}
	Dropout rate	0.5

- Each seedling s_i in S characterized by its objective function $f(s_i)$ and growth rate $g(s_i)$.
- The combined optimality index for ranking seedlings as:

$$F(s_i) = \alpha \cdot \hat{f}(s_i) + \beta \cdot \hat{g}(s_i)$$

To better comprehend the OA and its intricate steps, we present its pseudo-code in Algorithm 2.

Algorithm 2 Pseudo-code of OA

```
% Seedlings initialization
for i = 1 to N do
    Create an orchard by planting some seedlings
    Set number of strong trees in screening
    Set number of weak trees in screening
    Set number of trees that need grafting
    Set the growth-years number (GYN)
    Set α (The weight of fitness function)
    Set β (The weight of growth rate)
    Evaluate the fitness of seedlings
end for
% Seedlings growth through the search space
while termination criteria is not met do
    for i = 1 to N do
        Growth of the seedlings
        Screening based on fitness function and growth rate
        Grafting
        Replacement of the weak seedlings by the new ones
        Elitism
    end for
end while
```

3.5.3. Applying OA to Conv1D and GRU hyperparameter optimization

When channeling the principles of OA to the domain of Conv1D and GRU architectures, we represent neural network hyperparameters as the algorithm's seedlings. Here, each seedling embodies a specific configuration of hyperparameters, such as the number of filters for the Conv1D layer or the number of units for the GRU layer.

Through the operations of the OA, such as growth (a local search in the hyperparameter space) and grafting (combining parts of strong hyperparameter configurations into medium-strength ones), we aim to find an optimal configuration for the Conv-SGRU model. By steering this optimization journey with the mathematical underpinnings of OA, we promise enhanced model performance and efficient convergence (see Fig. 3).

The architecture of Conv-SGRU brings together the strengths of convolutional layers and GRU layers, necessitating intricate hyperparameter tuning. Inherent hyperparameters span the convolutional layers' dynamics, encapsulating the number of filters, kernel size, stride, padding, and activation function. Similarly, for the GRU layers, pertinent hyperparameters entail several hidden units, activation function, dropout rate, and recurrent dropout rate. Furthermore, training parameters like learning rate, batch size, and epoch count are pivotal for model convergence and performance.

Traditionally, tuning approaches such as grid search and random search, albeit prevalent, might not explore the full spectrum of potential hyperparameter combinations. Techniques like Bayesian optimization, on the other hand, add probabilistic efficiency but may not always guarantee global optima. Amidst this backdrop, the OA offers a novel, bio-inspired meta-heuristic optimization strategy tailored for Conv-SGRU.

Incorporating the principles of OA for Conv-SGRU, we depict each hyperparameter configuration as an OA seedling. The nurturing process, reflecting stages of growth, grafting, and replacement, mirrors hyperparameter optimization. Each seedling matures based on performance on a validation set or via cross-validation techniques.

The procedure begins with demarcating hyperparameters like learning rate, batch size, number of layers, number of filters, dropout rate, and activation function. Subsequently, guided by the OA dynamics, the search space for these hyperparameters is explored and exploited. During this course, OA's growth phase iteratively refines configurations via local search. Grafting mingles attributes from promising configurations, while weaker configurations are supplanted with fresh candidates.

The culmination of the OA process yields an optimal hyperparameter combination. Adopting this configuration, the Conv-SGRU model is trained exhaustively on the entire training dataset. Post-training, the model's mettle is assessed on an independent test set. This final step is paramount to ensure the robustness of the Conv-SGRU model and validate the efficacy of the OA in hyperparameter tuning (see Fig. 4).

3.6. TL in FL

DNNs are susceptible to overfitting, particularly when confronted with limited training data. This limitation often leads to high performance on the training set but poor generalization to new, unseen data. In a conventional centralized machine learning setup, the data aggregation required for model training usually occurs at a central server. This strategy raises concerns about data privacy and incurs high communication overhead. However, by integrating TL within a FL framework, we overcome several challenges traditionally associated with DNN training:

- FL negates the need for uploading local data from each edge device to a central server, thereby minimizing the risk of privacy infringements and legal issues.
- Models trained using an FL approach often outperform their locally-trained counterparts due to their ability to leverage diverse data across multiple edge devices.
- The distributed nature of FL allows for efficient use of computational resources, as multiple edge devices collaborate in real-time for model training.

Incorporating TL into our FL framework further enhances model performance by enabling the network to generalize from previous tasks, thereby ameliorating the issue of data scarcity and heterogeneity. As a result, the integrated FL-TL approach provides a robust, efficient, and privacy-preserving solution for tackling complex machine learning challenges (see Fig. 5).

Traditional FL is constrained by the requirement that training data owned by many organizations share the same feature space. In actuality, in sectors like banking or healthcare, this is never the case. Federated TL (FL-TL) was suggested to address this flaw [46]. FL-TL allows for the creation of specialized feature spaces for each participant, making it applicable to real-world situations. TL, a paradigm that is already well-liked in image analysis [47], serves as the model for FL-TL. In this situation, machine learning models that have been trained on a sizable dataset for one problem/domain are applied to another problem/domain that is similar but distinct [48]. Interrelation between several domains is a key factor in TL effectiveness. When discussing FL, organizations from the same industry are often the stakeholders in the same data federation. As a result, the FL architecture is appropriate for applying TL. Federated TL is at the nexus of two distinct but quickly developing fields: information privacy and machine learning. This is a general form of Federated TL shown in Algorithm 3 where:

- Starting from a pre-trained Source Model, this Federated TL technique initializes the global model parameters.
- Then, it chooses a subset of clients to take part in each round of training.
- Each client downloads the most recent global model parameters from the server, and then trains on local data to perfect its local model.
- After computing the modifications to their local models, the clients give the server the updates.
- These changes are averaged by the server, which then updates the entire model.
- This procedure is repeated several times.
- The revised global model parameters are returned at the conclusion of training.
- Utilizing the advantages of both TL and FL is the main concept of FL. Updating the model locally keeps the customers' data private while yet allowing them to take advantage of the information stored in the pre-trained model (see Fig. 6).

Algorithm 3 Basic Federated TL (FTL)

```
Procedure FEDERATED_TRANSFER_LEARNING (Source_Model, Clients)
Initialize global model parameters θ from Source_Model
for each round t do
  Select a subset of clients C_t to participate in training
  for each client k in C_t in parallel do
    Retrieve global model parameters θ from server
    Fine-tune local model parameters by training on local data
    Compute and send model updates Δθ_t to the server
  end for
  Compute global model update Δθ = average(Δθ_t for each k in C_t)
  Update global model parameters θ = θ + Δθ
end for
return Global model parameters θ
End Procedure
```

3.7. FL-Conv-SGRU

The Federated Conv-SGRU (FL-Conv-SGRU) model addresses the particular issues of data decentralization and privacy protection inherent in an FL scenario while using the benefits of the Conv-SGRU architecture.

The Conv-SGRU model is trained in FL by a large number of clients, each holding their private dataset relevant to PV power generation. This collaboration occurs without explicit data exchange. To train the model while maintaining data privacy, each client does local calculations on its dataset. The Conv-SGRU model is generally used in these calculations to extract temporal and semantic patterns from the data.

The clients then send the model changes to the central server rather than their private information. These changes are collected by this central server to improve the overall model. The FL-Conv-SGRU model may produce an accurate and dependable model for forecasting PV power generation by repeatedly iteration this local computation and global aggregation until convergence. The FL-Conv-SGRU model is a very practical option for forecasting in federated contexts since it successfully uses decentralized data while simultaneously maintaining data privacy.

3.8. OA-FL-TL-Conv-SGRU

The OA Enhanced TL Federated Conv-SGRU (OA-TL-FL-Conv-SGRU) model leverages the principles of Conv-SGRU, FL, TL, and the OA.

TL is the technique of leveraging knowledge from one task or dataset to enhance performance on a different, yet related task or dataset. Specifically, it involves pre-training the Conv-SGRU model on a larger, related dataset (e.g., another PV power generation dataset or a related time-series dataset) and then fine-tuning this model on a specific target dataset using the FL framework.

This pre-training step produces a robust initial model capable of identifying general patterns and features applicable across various datasets. This initial model is then refined on each client's unique dataset in a FL environment, thereby ensuring data privacy and efficient use of decentralized data. Before the fine-tuning step, the OA is applied to optimize the model parameters, which can potentially enhance the model's prediction accuracy and convergence speed.

By integrating TL into the FL paradigm, we can potentially expedite model convergence while simultaneously enhancing the performance of the Conv-SGRU model on the target PV power generation dataset, all while preserving the privacy and decentralized nature of the data.

A generalized depiction of the OA Enhanced Federated TL with a CNN and Stacked GRU (OA-FL-TL-CNN-SGRU) is illustrated in Algorithm 4.

The procedure is distinctively apt for the FL-TL-CNN-SGRU model, a Federated TL paradigm integrating the CNN with the Stacked GRU (SGRU).

The algorithm's modus operandi initiates by establishing the global model parameters via a pre-trained Source Model. In each subsequent round, a subset of clients is cherry-picked for participation. Every client fetches the latest global model parameters from the central server. Leveraging the CNN-SGRU architecture, each client refines its local model based on its proprietary dataset. Once the local model adjustments are determined, these updates are communicated back to the server. The server, in its capacity, averages out these updates, reflecting these changes onto the global model. This cyclical process is reiterated until a convergence criterion is achieved, culminating in the final global model parameters.

This methodology's signature differentiator compared to traditional Federated TL algorithms is the engagement of a tailored model architecture, i.e., CNN-SGRU, enabling clients to finetune their local models optimally.

4. Experimental setup

4.1. Dataset pre-processing and statistics

The datasets created for this study's evaluation of the suggested model (OA-FL-TL-Conv-SGRU) were based on details relevant to solar

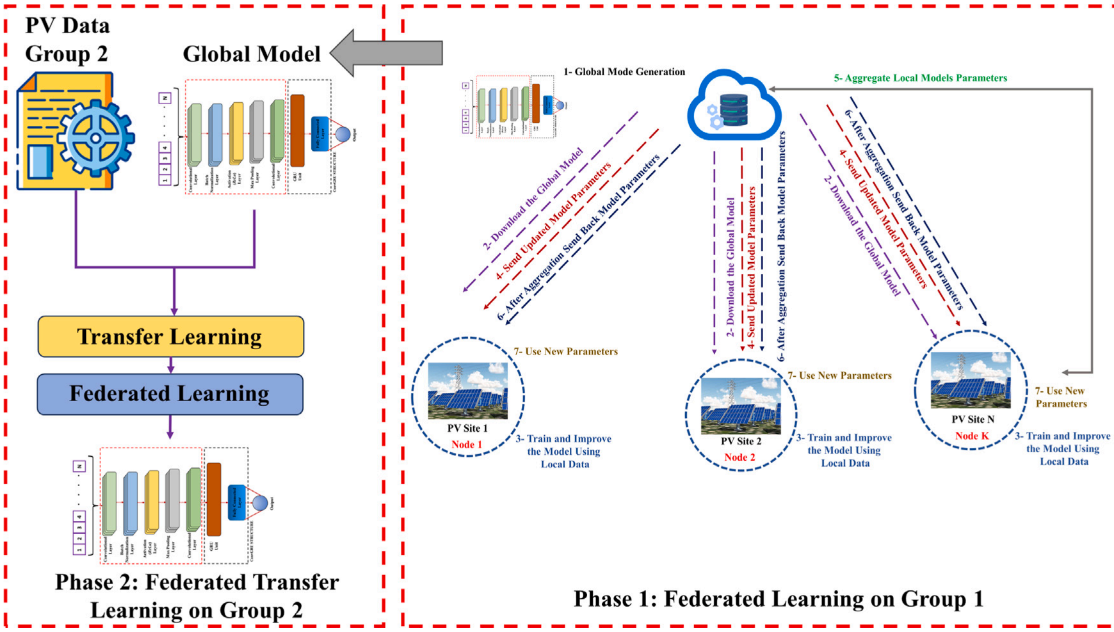

Fig. 6. Federated TL approach for PV Power forecasting.

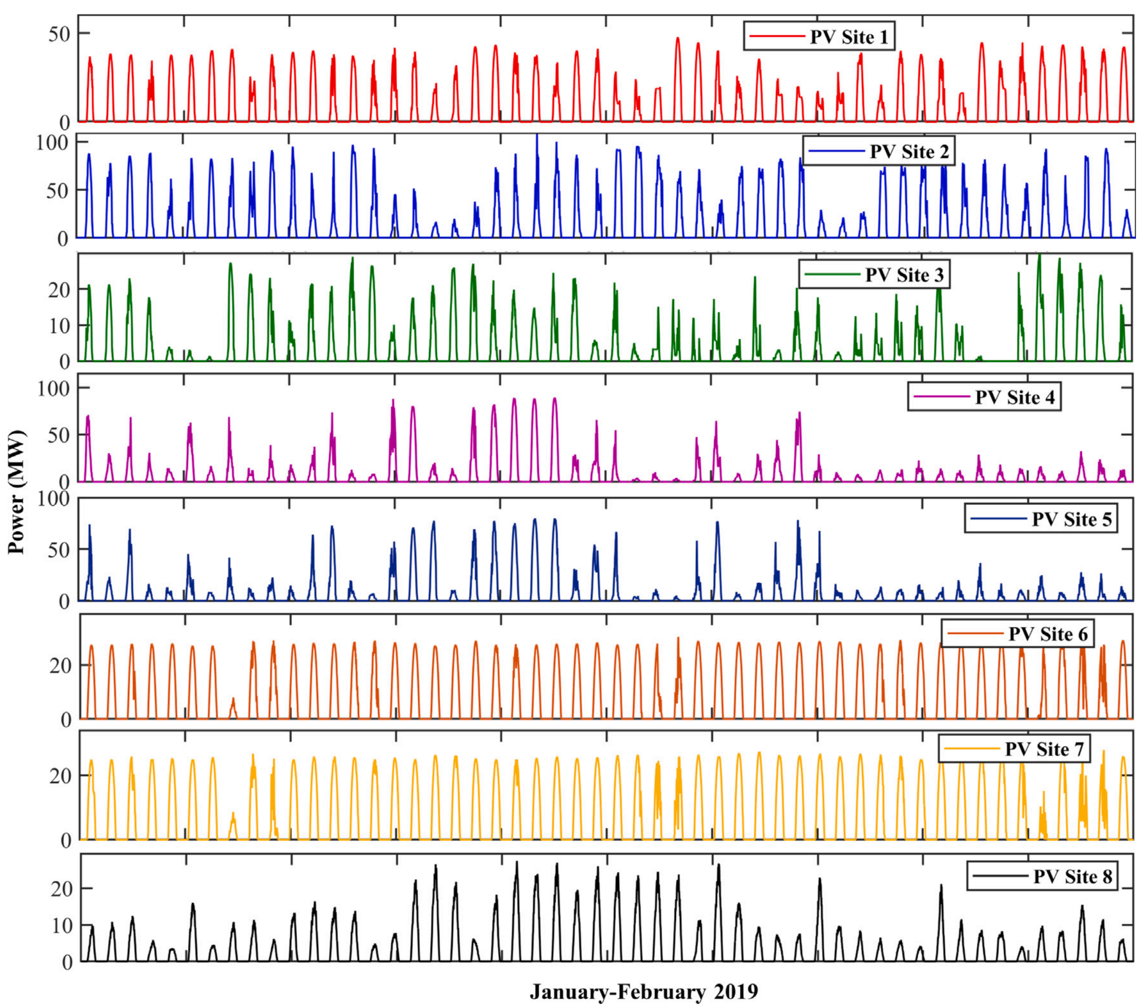

Fig. 7. Variation of power for all 8 PV sites for two months.

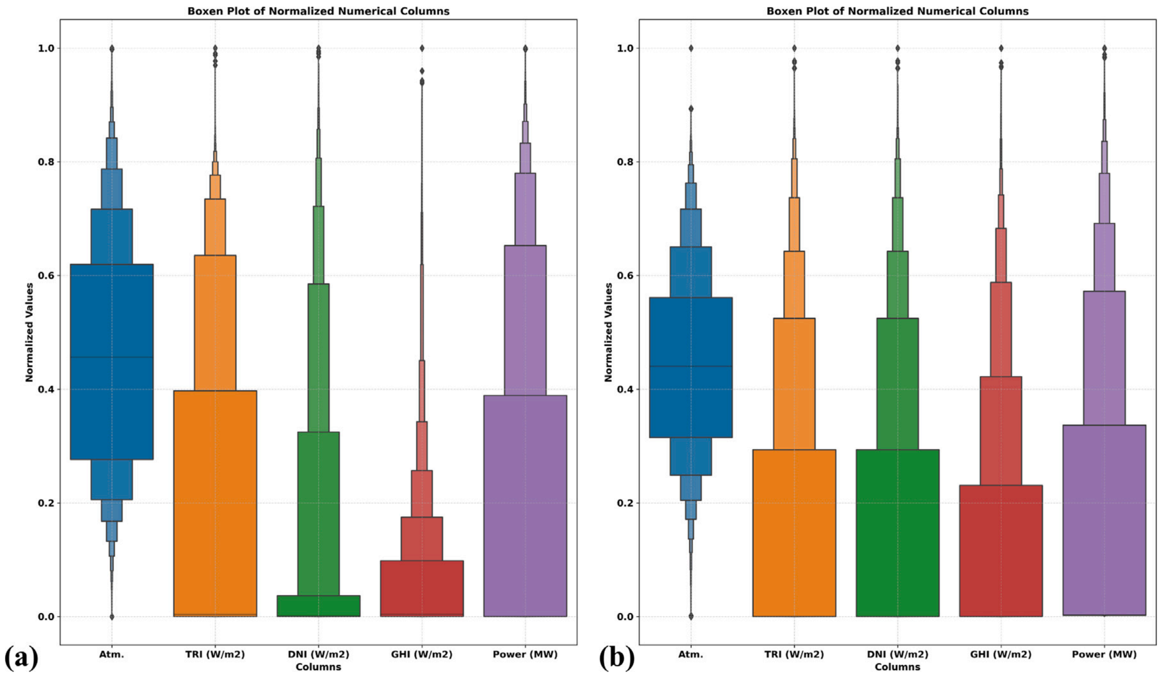

Fig. 8. Boxen plot for (a) Solar site 1 (b) Solar site 2 (Normalized column values).

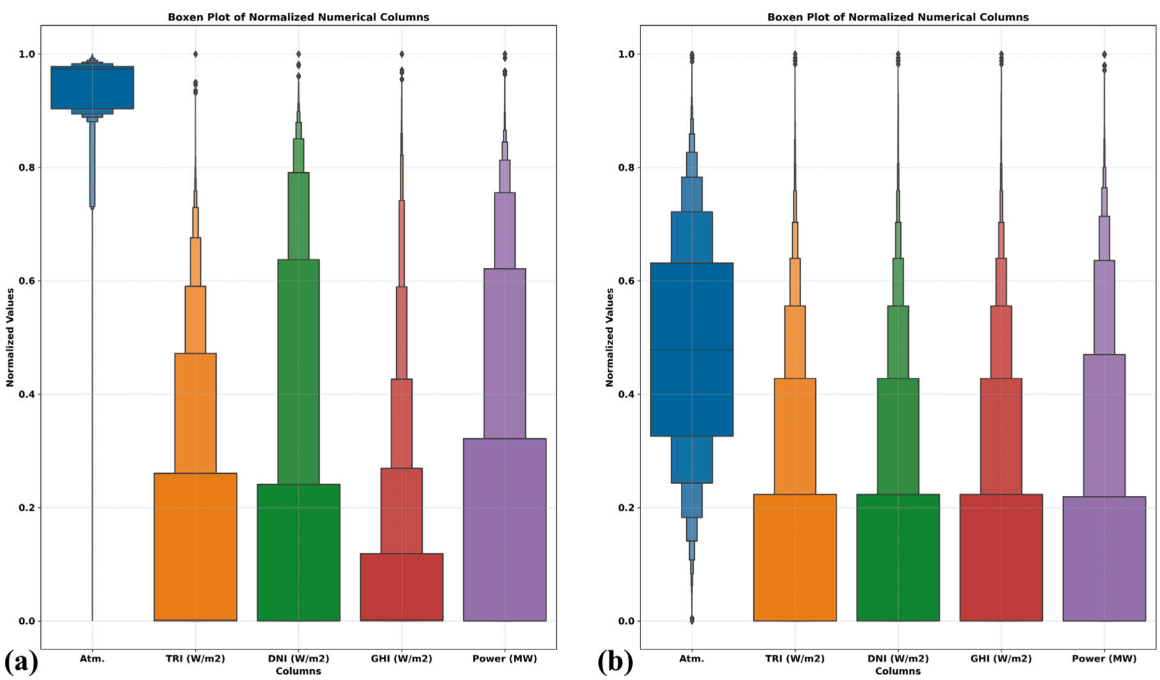

Fig. 9. Boxen plot for (a) Solar site 3 (b) Solar site 4 (Normalized column values).

energy. The Chinese State Grid gathered these statistics in 2021 when real-time monitoring of the weather and electricity output was carried out every 15 min. The procedure for data gathering and data value correction is explained in this document. To further verify the variability of power over time is shown in Fig. 7. The utilization of this data demonstrates tremendous promise for improving demand response (DR) programs for the electricity grid and creating data-driven forecasting techniques for the production of renewable energy. Additionally, Table 4 offers statistical summaries of the solar data values for each of the eight sites. These comprise information like the mean, standard deviation, and greatest and lowest values. This thorough statistical analysis is important for understanding the trends and variability in the solar data and is thus a key step in the modeling and forecasting process that follows Table 3.

Furthermore, we present the boxen plot, which was formerly known as a "letter value" plot because of the enormous number of quantiles that are designated as "letter values" it displays. When a non-parametric representation of a distribution is plotted, similar to a box

Table 4
Data statistics for all solar sites.

Statistical measures		Output power (MW)	Solar irradiance	Direct normal irradiance	Global horizontal irradiance	Atm. pressure (hPa)
Solar Site 1	Mean	9.669	266.2	93.25	67.69	913.367
	Min.	0	0	0	0	894
	Max.	48.32	1359	980	989	936.3
	Std Dev.	13.7	367.89	200.78	111.98	8.74
Solar Site 2	Mean	19.56	169.3	122.154	78.3	861.03
	Min.	0	0	0	0	844.51
	Max.	109.36	1041.93	751.75	561.8	936.3
	Std. Dev.	27.94	248.078	178.99	117.59	6.14
Solar Site 3	Mean	5.48	182.82	100.734	69.3	1016.03
	Min.	0	0	0	0	994.8
	Max.	29.11	1117	760	656	1038.6
	Std. Dev.	8.234	294.58	185.03	101.88	9.323
Solar Site 4	Mean	16.45	150.15	139.51	20,84	1011.37
	Min.	0.44	0	0	0	928.59
	Max.	114.68	1237	1010.27	150.96	1100.3
	Std. Dev.	27.45	253.43	210.68	31.48	33.22
Solar Site 5	Mean	14.51	164.6	148.1	115,27	1012
	Min.	0	0	0	0	990.7
	Max.	99.55	1467	1962	1208	1039
	Std. Dev.	23.89	273.74	235.13	203.42	9.94
Solar Site 6	Mean	6.36	243.08	215.15	53.93	830.67
	Min.	0	0	0	0	389.82
	Max.	32.24	1365.4	1179.8	296.2	846.07
	Std. Dev.	9.17	355.44	337.61	69.35	4.61
Solar Site 7	Mean	5.41	206.08	182.99	108.68	842,93
	Min.	0	0	0	0	398.2
	Max.	29.78	1393.73	1095.4	1125.13	867.1
	Std. Dev.	8.04	299.91	306.83	190.63	24.4
Solar Site 8	Mean	4.23	163.24	142.02	21.22	956.41
	Min.	0	0	0	0	881.4
	Max.	29.41	1214.54	1056.65	157.89	1037.78
	Std. Dev.	6.52	245.4	213.49	31.9	30.53

Algorithm 4 Federated TL with OA and CNN-SGRU (FL_OA_TL_CNN_SGRU)

```
Procedure FL_OA_TL_CNN_SGRU(Source_Model, Clients)
Initialize global model parameters θ from Source_Model
for each round t do
  Select a subset of clients C_t to participate in training
  for each client k in C_t in parallel do
    Retrieve global model parameters θ from server
    Perform OA for parameter refinement
    for i = 1 to N do
      Growth of the seedlings (adjust parameters)
      Screening based on validation performance (select best parameters)
      Grafting (combine features or layers from best-performing trees)
      Replace weak seedlings (parameters) with new ones
    end for
    Fine-tune local model parameters on local data with CNN_SGRU architecture
    Compute and send model updates Δθ_t to the server
  end for
  Compute global model update Δθ = average(Δθ_t for each k in C_t)
  Update global model parameters θ = θ + Δθ
end for
return Global model parameters θ
End Procedure
```

plot, all features correspond to real data. The distribution's shape, particularly in the tails, is better understood by showing more quantiles. The boxen plots are shown in Figs. 8 and 9.

4.2. Measurement and uncertainties in dataset acquisition

This section delves into the intricacies of our data collection and validation process, emphasizing the methodologies employed to ensure the accuracy and reliability of the dataset. In this regard, the discussion begins with an overview of our measurement methods, which involved the meticulous use of SCADA systems for data acquisition, allowing for precise monitoring and analysis of solar energy generation. Furthermore, our selection of geographically diverse solar stations across various terrains and climatic conditions is highlighted, underscoring our commitment to capturing a comprehensive representation of solar power generation. The technical validation procedures are also outlined, showcasing our thorough approach to handling missing data and outliers, with a focus on maintaining data quality. Additionally, the section addresses the presence of uncertainties in the field data, detailing how we transparently handle these uncertainties while preserving the fidelity of our dataset.

- Measurement Methods: The data for the solar stations was meticulously collected through the utilization of SCADA systems, renowned for their precision in the monitoring, control, and analysis of processes. Over a two-year span, from 2019 to 2020, data was consistently recorded at 15-minute intervals, enabling in-depth insights into the patterns of solar energy generation.

- Geographical and Climatic Diversity: A comprehensive representation of solar power generation across diverse terrains and climatic conditions was our primary objective. This entailed the selection of stations from North, Central, and Northwest China, encompassing a variety of landscapes, including deserts, mountains, and plains.
- Technical Validation: Recognizing the paramount significance of data quality and accuracy, our efforts in data pre-processing were characterized by diligence. A multi-faceted approach was employed to address missing data and outliers, incorporating techniques such as upward/downward completion, linear interpolation, and moving averages. Moreover, we have provided explicit clarification regarding the criteria for outlier detection, based on the interquartile range, while acknowledging the distinct challenges posed by the fluctuating nature of renewable energy.
- Uncertainties: Transparent disclosure of missing values and outliers in the dataset is a key feature of our approach. A comprehensive account of their occurrence rate is detailed in Table 6. Given the inherent uncertainties associated with field data, we have adopted a conservative strategy in our data pre-processing, ensuring the integrity of our dataset.

4.3. Objective function and evaluation parameters

The suggested approach is trained and validated using an objective function (fitness function). Lesser values of the utility function illustrate how closely the model's data projections match reality. As a result, the fitness function determines the prediction accuracy. The MSE is the most used fitness function and can be expressed as follows:

$$F.F = \frac{1}{n}\sum_{i=1}^{n}(T_i - P_i)^2, \tag{10}$$

where T_i, P_i, and n represent the true values, anticipated values, and the tally of samples, respectively.

The evaluation of alternative models also employs a number of error indexes. NMSE may be used to verify the degree of dispersion of the results as shown in Eq. (11). Forecast deviation indicators MAE and RMSE are provided in the equations Eq. (12) and (13), respectively. The correlation between actual and predicted values may also be found using the R^2 value, as indicated in Eq. (14).

$$NMSE = \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{11}$$

$$MAE = \frac{1}{M}\sum_{a=1}^{M}|T_a - P_a| \times 100\% \tag{12}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{13}$$

$$R^2 = \frac{\sum_{a=1}^{M}(T_a - \bar{T})(P_a - \bar{P})^2}{\sum_{a=1}^{M}(T_a - \bar{T})\sum_{a=1}^{M}(P_a - \bar{P})}, \tag{14}$$

4.4. Proposed model based prediction model

In order to extract valuable characteristics from the acquired datasets, an initial preprocessing step is executed. Following this, the data is partitioned into training and testing subsets. Special attention is given to dataset 3 as it mirrors the annual variation in solar power generation.

To effectively navigate the search space of possible model configurations, the OA is introduced. Within the context of the OA, seedlings, representing individual model configurations, are planted. These seedlings grow, are screened based on their fitness, and eventually evolve, imitating the learning process. This helps in determining the model's hyperparameters, replacing the traditional hit-and-trial approach, and ensuring that the CNN-Stacked GRU (CNN-SGRU) model is efficiently tailored for the task.

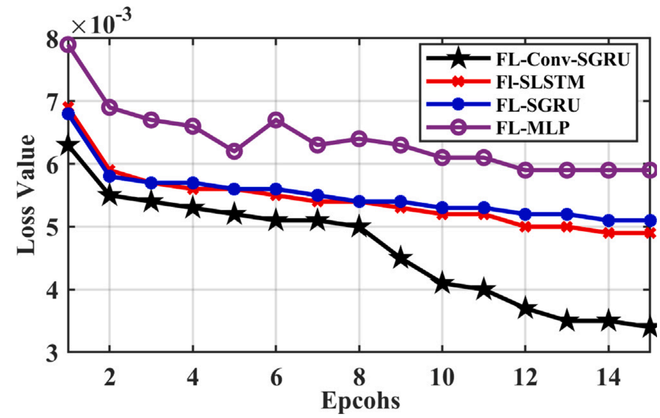

Fig. 10. Comparison of training loss vs epochs on Group 1 dataset using FL.

Once the optimal model configuration is identified through OA's iterative process, the model is trained using the identified hyperparameters and subsequently tested to forecast solar power generation. Various metrics are employed to gauge the accuracy and reliability of these forecasts, offering a multifaceted insight into the model's capabilities. Each metric unravels different aspects of the model's performance, ensuring a thorough evaluation.

For the analysis, two distinct dataset groups, Group 1 and Group 2, are considered. Each group comprises four unique datasets. The underlying philosophy of TL informs the structure of these groups. Group 1 datasets act as source datasets, aiding in the model's initial training. Here, the model imbibes essential patterns, which get encapsulated within its weights. As training progresses to Group 2 datasets, these pre-learned weights form the foundational knowledge. Thus, while Group 1 datasets are used for preliminary model training, Group 2 capitalizes on the "transferred" learnings.

5. Results

In this section, we examine the experimental findings made possible by applying the suggested model and the comparison techniques. The model's correctness is determined using a variety of statistical indicators, which provides a thorough assessment of the model's performance. For background, two groups of datasets representing various locales are used in our experimental setting. There are four separate datasets in each category. In a TL scenario, Group 1 acts as the retraining dataset, and the learned model's weights on this group are then used to initialize the learning on Group 2. Let us examine the outcomes for each group independently now.

5.1. Comparative analysis on Group 1: Without TL

The presented tables i.e. Tables 5 and 6 showcase the comparative evaluation of competing techniques, including FL with different neural architectures (OA-FL-Conv-SGRU, FL-SLSTM, FL-SGRU, FL-MLP), on Group1 PV datasets during both Summer and Winter seasons. The metrics employed for assessment encompass NMSE, MAE, R2, and RMSE, which collectively gauge the predictive accuracy and reliability of these models. Notably, in both seasons, OA-FL-Conv-SGRU emerges as the superior performer, consistently exhibiting the lowest NMSE (Summer: 0.0113, Winter: 0.0122), MAE (Summer: 0.0251, Winter: 0.0248), and RMSE (Summer: 0.0301, Winter: 0.0304), while attaining the highest R2 (Summer: 94.62%, Winter: 94.50%) values compared to its counterparts (see Fig. 10).

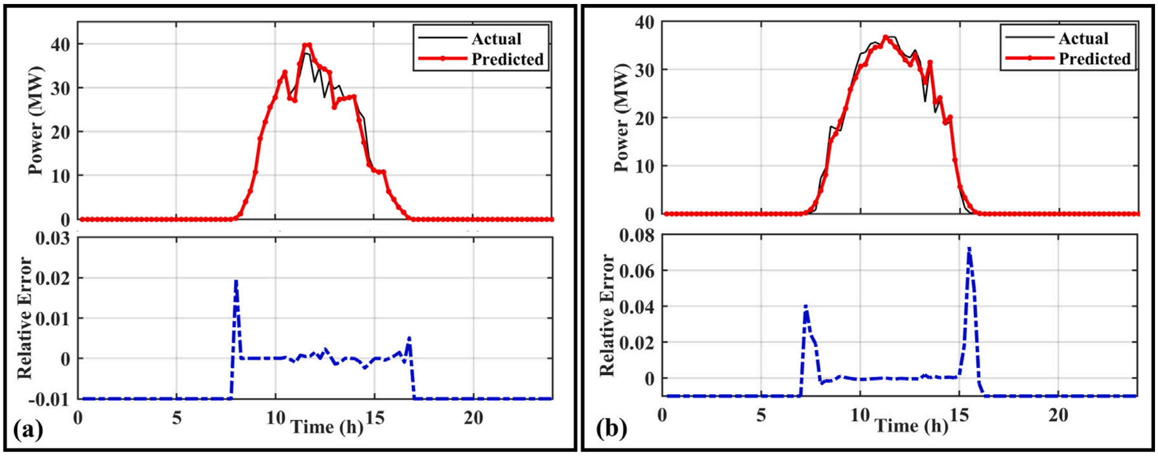

Fig. 11. Actual vs predicted power comparison of proposed technique in (a) Summer and (b) Winter season with relative error.

Table 5
Comparison of evaluation matrices for competing techniques on group 1 PV datasets in summer season.

Metrics	OA-FL-Conv-SGRU	FL-SLSTM	FL-SGRU	FL-MLP
NMSE	0.0113	0.0291	0.0502	0.1912
MAE	0.0251	0.0513	0.0970	0.2551
R2	0.9462	0.9384	0.9302	0.9051
RMSE	0.0301	0.0493	0.0734	0.1788

Table 6
Comparison of evaluation matrices for competing techniques on group 1 PV datasets in winter season.

Metrics	OA-FL-Conv-SGRU	FL-SLSTM	FL-SGRU	FL-MLP
NMSE	0.0122	0.0285	0.0498	0.1905
MAE	0.0248	0.0507	0.0965	0.2565
R2	0.9450	0.9378	0.9310	0.9035
RMSE	0.0304	0.0499	0.0728	0.1793

Table 7
Comparison of evaluation matrices for competing techniques on group 2 PV datasets in summer season.

Metrics	OA-FL-Conv-SGRU	FL-SLSTM	FL-SGRU	FL-MLP
NMSE	0.0115	0.0293	0.0505	0.1910
MAE	0.0249	0.0515	0.0972	0.2549
R2	0.9460	0.9382	0.9300	0.9050
RMSE	0.0303	0.0495	0.0733	0.1785

Table 8
Comparison of evaluation matrices for competing techniques on group 2 PV datasets in winter season.

Metrics	OA-FL-Conv-SGRU	FL-SLSTM	FL-SGRU	FL-MLP
NMSE	0.0120	0.0288	0.0500	0.1900
MAE	0.0250	0.0510	0.0968	0.2560
R2	0.9455	0.9375	0.9305	0.9030
RMSE	0.0305	0.0500	0.0725	0.1790

The remarkable efficiency of the OA-FL-Conv-SGRU model underscores the potential of integrating the OA with FL, convolutional layers, and GRU. The convolutional layers adeptly identify spatial patterns within photovoltaic data, while the GRU units skillfully capture time-based dependencies. This synergy proves vital in understanding the complex dynamics of solar power generation. The robust performance of OA-FL-Conv-SGRU remains consistent across various seasons, signifying its ability to handle seasonal shifts and weather-related changes. Although models like FL-SLSTM, FL-SGRU, and FL-MLP display commendable results, their NMSE, MAE, and RMSE metrics indicate a lesser predictive accuracy in comparison to OA-FL-Conv-SGRU. Conclusively, these findings highlight the vast potential of integrating the OA and FL with tailored neural architectures, pointing towards a promising future for enhanced photovoltaic energy predictions and sustainable energy management (see Fig. 11).

5.2. Comparative analysis on Group 2: Without TL

Table 7 and 8 offer a comprehensive evaluation of competing techniques utilizing FL on Group 2 PV datasets during both the Summer and Winter seasons. The employed evaluation metrics—Normalized Mean Squared Error (NMSE), Mean Absolute Error (MAE), Coefficient of Determination (R2), and Root Mean Squared Error (RMSE)—provide valuable insights into the techniques' performance. In the Summer season, OA-FL-Conv-SGRU demonstrates remarkable superiority, exhibiting a substantially lower NMSE of 0.0115 compared to FL-SLSTM 0.0293, FL-SGRU 0.0505, and FL-MLP 0.1910. This performance advantage is similarly reflected across other metrics, with OA-FL-Conv-SGRU achieving the lowest MAE 0.0249 and RMSE 0.0303 while attaining the highest R2 94.60 all indicative of its superior predictive accuracy and fitting. Conversely, FL-MLP exhibits the weakest performance, likely attributed to its shallower architecture. Even in the Winter season, OA-FL-Conv-SGRU maintains its competitive edge with a lower NMSE of 0.0120 compared to other techniques. It continues to outperform its counterparts across all metrics, with the lowest MAE of 0.0250 RMSE 0.0305, and a high R2 of 94.55. This consistency in performance further underscores the robustness and effectiveness of the OA-FL-Conv-SGRU approach. Overall, these findings underscore the efficacy of the OA-FL-Conv-SGRU technique in capturing the intricate seasonal variations of the PV datasets, highlighting its potential for enhanced predictive modeling and energy yield estimation in solar photovoltaic systems (see Fig. 12).

5.3. Comparative analysis on Group 2: With TL

Table 9 and 10 continue the analysis of competing techniques using FL with TL on Group 2 PV datasets, focusing on their performance during the Summer and Winter seasons. The performance of the various methods is comprehensively assessed using a range of evaluation metrics including Normalized Mean Squared Error (NMSE), Mean Absolute Error (MAE), Coefficient of Determination (R2), and Root Mean Squared Error (RMSE). In both the Summer and Winter seasons, OA-FL-TL-CNN-SGRU maintains a consistent pattern of superior performance across all evaluated metrics. During the Summer season, OA-FL-TL-CNN-SGRU achieves the lowest NMSE 0.0081 compared to FL-TL-SLSTM 0.0122, FL-TL-SGRU 0.0319, and FL-TL-MLP 0.0807. This

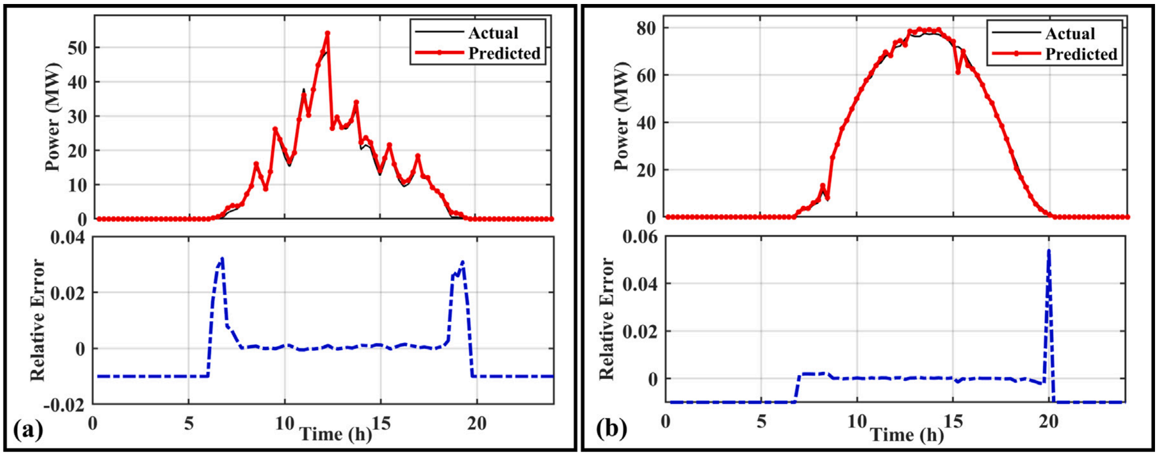

Fig. 12. Actual vs predicted power comparison of proposed technique in (a) Summer and (b) Winter season with relative error for Group 2 without TL.

Table 9
Comparison of evaluation matrices for competing techniques on group 2 PV datasets in summer season.

Metrics	OA-FL-TL-CNN-SGRU	FL-TL-SLSTM	FL-TL-SGRU	FL-TL-MLP
NMSE	0.0081	0.0122	0.0319	0.0807
MAE	0.0101	0.0386	0.0744	0.1224
R2	0.9775	0.9591	0.9415	0.9163
RMSE	0.0133	0.0201	0.0522	0.1025

Table 10
Comparison of evaluation matrices for competing techniques on group 2 PV datasets in winter season.

Metrics	OA-FL-TL-CNN-SGRU	FL-TL-SLSTM	FL-TL-SGRU	FL-TL-MLP
NMSE	0.0083	0.0125	0.0315	0.0795
MAE	0.0103	0.0388	0.0738	0.1230
R2	0.9770	0.9585	0.9412	0.9168
RMSE	0.0136	0.0203	0.0520	0.1020

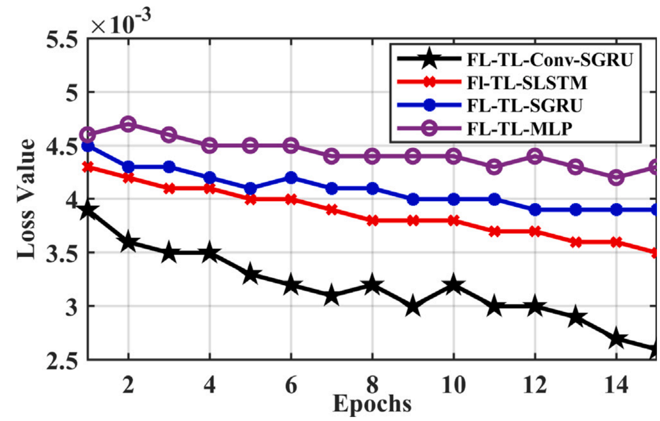

Fig. 13. Comparison of loss vs epochs on Group 2 dataset using federated TL.

trend persists across MAE, with OA-FL-TL-CNN-SGRU again exhibiting the smallest value 0.0101, underscoring its accurate predictions. Additionally, OA-FL-TL-CNN-SGRU achieves the highest R2 97.75, and the lowest RMSE 0.0133, highlighting its superior fitting capability and precision. In the Winter season, OA-FL-TL-CNN-SGRU's dominance remains evident, as it attains the lowest NMSE 0.0083, MAE 0.0103, and RMSE 0.0136 compared to its counterparts. Its consistently high R2 97.70 further underscores its ability to capture the underlying patterns in the data effectively. FL-TL-SLSTM, FL-TL-SGRU, and FL-TL-MLP show comparatively higher error rates, suggesting that OA-FL-TL-CNN-SGRU's architecture is particularly well-suited for handling seasonal variations in the PV datasets (see Fig. 13).

These findings underscore the robustness and reliability of OA-FL-TL-CNN-SGRU in accurately modeling and predicting PV system performance across diverse seasonal conditions. Its consistently superior performance across metrics and seasons suggests its potential for enhancing the accuracy and reliability of predictive modeling for solar energy applications (see Fig. 14).

5.4. Comparative analysis

An in-depth examination of the results from the three groups underscored the impact of integrating the OA with TL in enhancing the proposed model, OA-FL-TL-CNN-SGRU (see Fig. 15).

The OA-FL-CNN-SGRU model performed competitively in Group 1 when TL was not used, on par with the other models (FL-CNN-LSTM, FL-CNN-GRU, and FL-MLP). The suggested model produced an R2 score of around 0.9262, an MSE of 0.0049, and an MAE of 0.0251 in this configuration. However, OA-FL-TL-CNN-SGRU outperformed the other models on all criteria once TL was included in Group 2. The MSE and MAE decreased to 0.0030 and 0.0180, respectively, while the R2 score of the model increased to 0.9610, indicating improved prediction accuracy.

The performance of the OA-FL-TL-CNN-SGRU model declined to levels seen in Group 1 when Group 3 switched to not using TL. The MSE and MAE climbed to 0.0047 and 0.0244, respectively, while the model's R2 score dropped to 0.9282. These adjustments confirmed TL's importance in enhancing model performance.

When TL was used in Group 2, the other models (FL-CNN-LSTM, FL-CNN-GRU, and FL-MLP) also showed some improvement in their performance measures, however, the improvements were not as noticeable as those found for the proposed model. This comparative research shows how TL may improve model performance overall, especially in FL contexts where data can be varied and dispersed across numerous clients. The suggested OA-FL-TL-CNN-SGRU model successfully used data from one set of datasets to improve predictions on another, demonstrating the value of TL and supporting the advantages of FL frameworks.

5.5. Comparative analysis with previous studies

The presented Table 11 offers a comprehensive comparative analysis of various FL-based techniques for PV power forecasting, juxtaposed with the results obtained from the newly proposed OA-FL-TL-CNN-SGRU model. This analysis contributes to a deeper understanding of the performance landscape and highlights the advantages of the novel

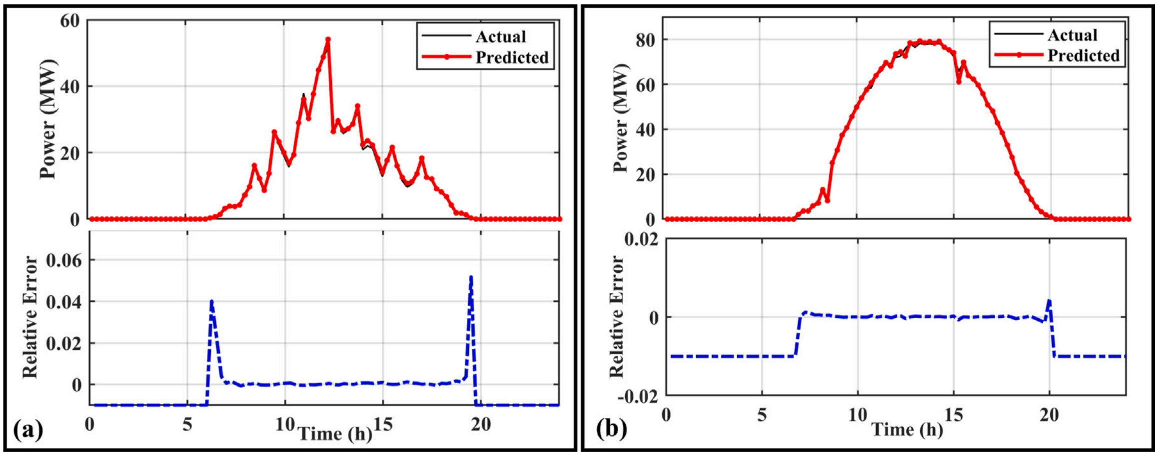

Fig. 14. Actual vs Predicted power comparison of proposed technique in (a) Summer and (b) Winter season with relative error for Group 2 with TL.

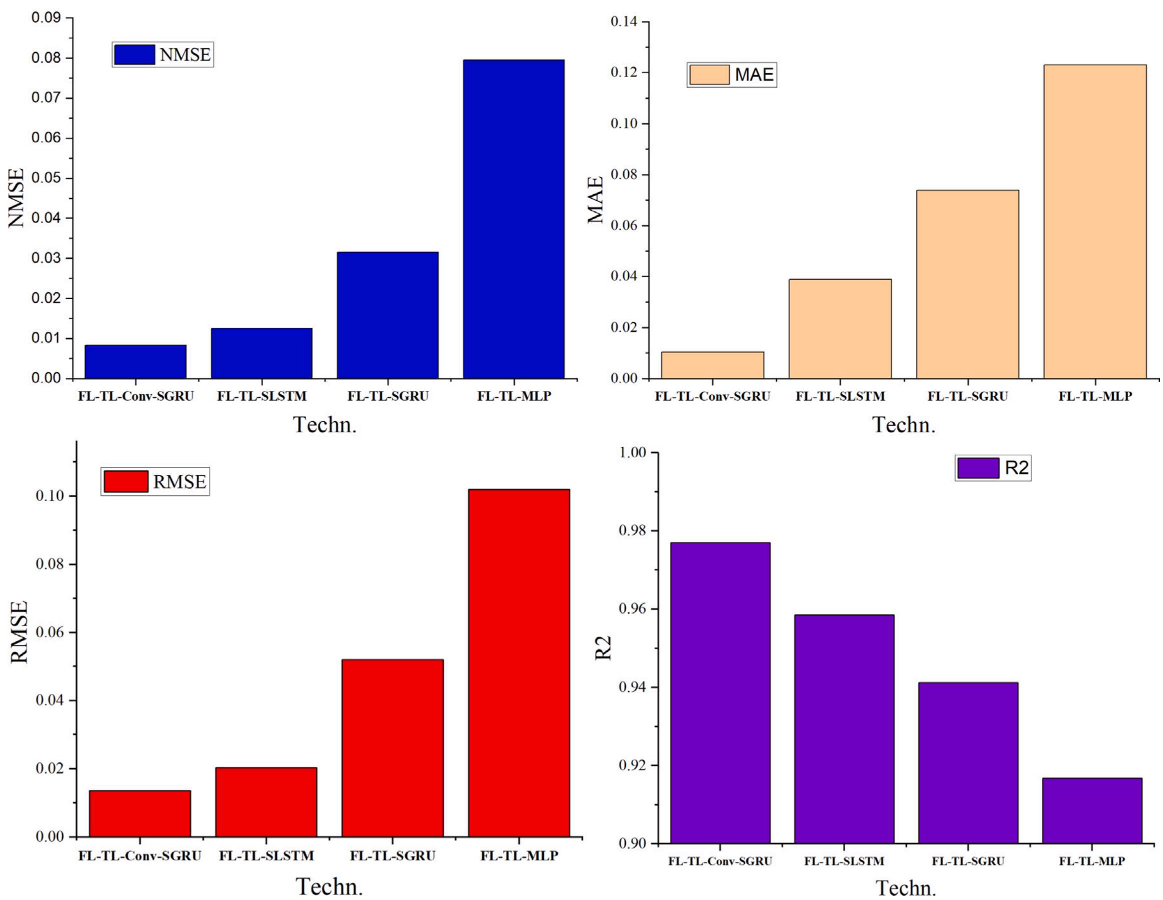

Fig. 15. Bar chart comparison of evaluation matrices of competing techniques.

model in terms of predictive accuracy and precision. One of the references, [13], employs an MLP-FL-based BTM for PV forecasting. Their model yields an RMSE of 9.72 and a MAPE of 13%, indicating that the predictions have an average deviation of 9.72 units from the actual values. Another study [26] introduces FedDRL using DDPG, which attains a notable level of accuracy with an NMAE of 0.020 and NRMSE of 0.037. This demonstrates the model's capability to closely predict PV power values. In contrast, the FL-CNN-LSTM model proposed in [27] achieves an RMSE of 0.07, exhibiting a notably higher precision in forecasting PV power. The subsequent investigation [14] focuses on FL-LSTM, revealing a relatively strong R2-Score of 0.84, an RMSE of 0.089, a low MSE of 0.017, and an MAE of 0.045. These metrics collectively suggest that the FL-LSTM model is adept at capturing the underlying patterns and fluctuations in PV power production. In the context of this comparative analysis, the newly introduced OA-FL-TL-CNN-SGRU (Our Model) emerges as a standout performer. It showcases exceptional predictive accuracy, as evidenced by its low RMSE of 0.013, NMSE of 0.0083, and a remarkable R2 of 0.9775. Moreover, its MAE of 0.0103 underscores its capability to minimize the absolute prediction error. These outcomes collectively position the OA-FL-TL-CNN-SGRU model as a highly promising and accurate tool for PV power forecasting.

In summary, the presented table underscores the superiority of the OA-FL-TL-CNN-SGRU model over existing FL-based techniques for PV

Table 11
Comparison of FL-based power forecasting.

Ref.	Year	Technique	Results
[13]	2023	MLP-FL-based BTM PV forecasting model	RMSE: 9.72 MAPE: 13%
[26]	2023	FedDRL using DDPG	NMAE:0.020 NRMSE:0.037
[27]	2023	FL-CNN-LSTM	RMSE: 0.07
[14]	2023	FL-LSTM	R2-Score: 0.84 RMSE: 0.089 MSE:0.017 MAE: 0.045
Our Model	2023	OA-FL-TL-CNN-SGRU	RMSE: 0.013 NMSE: 0.0083 R2: 0.9775 MAE: 0.0103

power forecasting. Its consistently exceptional performance across multiple evaluation metrics suggests its potential to enhance the precision and reliability of PV power predictions, contributing to more efficient and effective energy management strategies.

6. Conclusion

In an era marked by increasing reliance on renewable energy sources and heightened concerns about data privacy, this paper presents a pioneering approach to photovoltaic (PV) power forecasting that is both accurate and privacy-aware. At the heart of our solution is the Orchard Optimized Federated Transfer Learning-based COnvolutional Neural Network with Stacked Gated Recurrent Unit (OA-FL-TL-CNN-SGRU) model, which seamlessly integrates FL and TL within a Convolutional Neural Network (CNN) and Stacked Gated Recurrent Unit (SGRU) architecture. This innovative combination excels in capturing the intricate spatiotemporal dynamics of PV power generation, consistently outperforming existing forecasting techniques across a range of evaluation metrics. Key to the success of our model is its ability to engage in collaborative training over a multitude of heterogeneous datasets, thanks to FL. This decentralized approach ensures the confidentiality of sensitive data across diverse PV installations while still leveraging their collective intelligence. Furthermore, the inclusion of TL allows our model to generalize across varying system setups and geographical conditions, thus enhancing its robustness and versatility. As we navigate a changing global energy landscape that increasingly emphasizes renewable resources and data privacy, our OA-FL-TL-CNN-SGRU framework stands out as an invaluable asset. It addresses the urgent demands for efficient energy management, safeguards data privacy, and sets a new benchmark for PV power forecasting, making it a timely and impactful contribution to the field.

Declaration of competing interest

All authors claim that there is not any conflict of interest regarding the above submission. The work of this submission has not been published previously. It is not under consideration for publication elsewhere. Its publication is approved by all authors and that, if accepted, it will not be published elsewhere in the same form, in English or in any other language, including electronically without the written consent of the copyright-holder.

Data availability

The links to the data are provided.

Acknowledgments

This research is supported by the Biomechatronics and Collaborative Robotics research group at the Top Research Center Mechatronics (TRCM), University of Agder (UiA), Norway.

References

[1] Y. Li, J. Wang, D. Zhao, G. Li, C. Chen, A two-stage approach for combined heat and power economic emission dispatch: Combining multi-objective optimization with integrated decision making, Energy 162 (2018) 237–254, http://dx.doi.org/10.1016/j.energy.2018.07.200.
[2] Renewables information 2019 - Analysis, IEA, 2021, URL https://www.iea.org/reports/renewables-information-overview. (Accessed 10 August 2023).
[3] M. Rana, I. Koprinska, V.G. Agelidis, 2D-interval forecasts for solar power production, Sol. Energy 122 (2015) 191–203.
[4] M.H. Alsharif, M.K. Younes, J. Kim, Time series ARIMA model for prediction of daily and monthly average global solar radiation: The case study of Seoul, South Korea, Symmetry 11 (2) (2019) 240.
[5] H. Zang, L. Cheng, T. Ding, K.W. Cheung, Z. Liang, Z. Wei, G. Sun, Hybrid method for short-term photovoltaic power forecasting based on deep convolutional neural network, IET Gener. Transm. Distrib. 12 (20) (2018) 4557–4567.
[6] G. Li, H. Wang, S. Zhang, J. Xin, H. Liu, Recurrent neural networks based photovoltaic power forecasting approach, Energies 12 (13) (2019) 2538.
[7] C.J. Hoofnagle, B. Van Der Sloot, F.Z. Borgesius, The European union general data protection regulation: what it is and what it means, Inf. Commun. Technol. Law 28 (1) (2019) 65–98.
[8] The Health Insurance Portability and Accountability Act of 1996, HIPAA, Centers for Medicare & Medicaid Services, 2018, URL http://www.cms.hhs.gov/hipaa/. (Accessed 10 August 2023).
[9] B. McMahan, E. Moore, D. Ramage, S. Hampson, B.A. y Arcas, Communication-efficient learning of deep networks from decentralized data, in: Artificial Intelligence and Statistics, PMLR, 2017, pp. 1273–1282.
[10] X. Zhang, F. Fang, J. Wang, Probabilistic solar irradiation forecasting based on variational Bayesian inference with secure federated learning, IEEE Trans. Ind. Inform. 17 (11) (2020) 7849–7859.
[11] S. Jiang, L. Wang, D. Zhang, J. Liu, Y. Zhang, M. Li, Federated learning-based forecasting model for PV power generations considering privacy protection and data sharing, in: The 10th Renewable Power Generation Conference, Vol. 2021, RPG 2021, 2021, pp. 641–645, http://dx.doi.org/10.1049/icp.2021.2326.
[12] Y. Shi, X. Xu, Deep federated adaptation: An adaptative residential load forecasting approach with federated learning, Sensors 22 (9) (2022) 3264.
[13] P. Hosseini, S. Taheri, J. Akhavan, A. Razban, Privacy-preserving federated learning: Application to behind-the-meter solar photovoltaic generation forecasting, Energy Convers. Manage. 283 (2023) 116900.
[14] A. Ahmadi, M. Talaei, M. Sadipour, A.M. Amani, M. Jalili, Deep federated learning-based privacy-preserving wind power forecasting, IEEE Access 11 (2023) 39521–39530, http://dx.doi.org/10.1109/ACCESS.2022.3232475.
[15] A. El Hendouzi, A. Bourouhou, Solar photovoltaic power forecasting, J. Electr. Comput. Eng. 2020 (2020) 1–21.
[16] N. Amral, C. Ozveren, D. King, Short term load forecasting using multiple linear regression, in: 2007 42nd International Universities Power Engineering Conference, IEEE, 2007, pp. 1192–1198.
[17] F. Barbieri, S. Rajakaruna, A. Ghosh, Very short-term photovoltaic power forecasting with cloud modeling: A review, Renew. Sustain. Energy Rev. 75 (2017) 242–263.
[18] M. Karamirad, M. Omid, R. Alimardani, H. Mousazadeh, S.N. Heidari, ANN based simulation and experimental verification of analytical four-and five-parameters models of PV modules, Simul. Model. Pract. Theory 34 (2013) 86–98.
[19] Z. Yang, J. Wang, A hybrid forecasting approach applied in wind speed forecasting based on a data processing strategy and an optimized artificial intelligence algorithm, Energy 160 (2018) 87–100.
[20] M. Hassanzadeh, M. Etezadi-Amoli, M. Fadali, Practical approach for sub-hourly and hourly prediction of PV power output, in: North American Power Symposium 2010, 2010, pp. 1–5.
[21] M. Alanazi, M. Mahoor, A. Khodaei, Two-stage hybrid day-ahead solar forecasting, in: 2017 North American Power Symposium, NAPS, 2017, pp. 1–6.
[22] L. Ferbar Tratar, E. Strmcnik, The comparison of Holt–Winters method and multiple regression method: a case study, Energy 109 (2016) 266–276.
[23] A. Yona, T. Senjyu, A.Y. Saber, T. Funabashi, H. Sekine, C.H. Kim, Application of neural network to 24-hour-ahead generating power forecasting for PV system, in: 2008 IEEE Power and Energy Society General Meeting - Conversion and Delivery of Electrical Energy in the 21st Century, 2008, pp. 1–6.
[24] H. Lu, G. Chang, A hybrid approach for day-ahead forecast of PV power generation, IFAC-PapersOnLine 51 (28) (2018) 634–638.
[25] M. Abdel-Nasser, K. Mahmoud, Accurate photovoltaic power forecasting models using deep LSTM-RNN, Neural Comput. Appl. 31 (7) (2019) 2727–2740.

[26] Y. Li, R. Wang, Y. Li, M. Zhang, C. Long, Wind power forecasting considering data privacy protection: A federated deep reinforcement learning approach, Appl. Energy 329 (2023) 120291.
[27] W. Zhang, X. Chen, K. He, L. Chen, L. Xu, X. Wang, S. Yang, Semi-asynchronous personalized federated learning for short-term photovoltaic power forecasting, Digit. Commun. Netw. (2022).
[28] T. Li, A.K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, V. Smith, Federated optimization in heterogeneous networks, Proc. Mach. Learn. Syst. 2 (2020) 429–450.
[29] C. Xie, S. Koyejo, I. Gupta, Asynchronous federated optimization, 2019, arXiv preprint arXiv:1903.03934.
[30] K. Wang, R. Mathews, C. Kiddon, H. Eichner, F. Beaufays, D. Ramage, Federated evaluation of on-device personalization, 2019, arXiv preprint arXiv:1910.10252.
[31] V. Smith, C. Chiang, M. Sanjabi, A. Talwalkar, Federated multi-task learning, in: Advances in Neural Information Processing Systems 30, NIPS 2017, 2017, pp. 4427–4437.
[32] A. Fallah, A. Mokhtari, A. Ozdaglar, Personalized federated learning: A meta-learning approach, 2020, arXiv preprint arXiv:2002.07948.
[33] K.C. Sim, P. Zadrazil, F. Beaufays, An investigation into on-device personalization of end-to-end automatic speech recognition models, 2019, arXiv preprint arXiv:1909.06678.
[34] A.K. Ozcanli, M. Baysal, Islanding detection in microgrid using deep learning based on 1D CNN and CNN-LSTM networks, Sustain. Energy Grids Netw. 32 (2022) 100839.
[35] R. Ahmed, et al., A review and evaluation of the state-of-the-art in PV solar power forecasting: Techniques and optimization, Renew. Sustain. Energy Rev. 124 (2020) 109792.
[36] M. Ding, et al., A gated recurrent unit neural networks based wind speed error correction model for short-term wind power forecasting, Neurocomputing 365 (2019).
[37] A. Kisvari, Z. Lin, X. Liu, Wind power forecasting – a data-driven method along with gated recurrent neural network, Renew. Energy 163 (2021) 1895–1909.
[38] L. Liao, H. Li, W. Shang, L. Ma, An empirical study of the impact of hyperparameter tuning and model optimization on the performance properties of deep neural networks, ACM Trans. Softw. Eng. Methodol. (TOSEM) 31 (3) (2022) 1–40.
[39] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT Press, 2016.
[40] F. Hutter, H. Hoos, K. Leyton-Brown, An evaluation of sequential model-based optimization for expensive blackbox functions, in: Proceedings of the 15th Annual Conference Companion on Genetic and Evolutionary Computation, 2013, pp. 1209–1216.
[41] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, 2014, arXiv preprint arXiv:1412.6980.
[42] T. Tieleman, G. Hinton, Rmsprop: Divide the gradient by a running average of its recent magnitude. coursera: Neural networks for machine learning, COURSERA Neural Netw. Mach. Learn. 17 (2012).
[43] K.O. Stanley, R. Miikkulainen, Evolving neural networks through augmenting topologies, Evol. Comput. 10 (2) (2002) 99–127.
[44] B. Shahriari, K. Swersky, Z. Wang, R.P. Adams, N. De Freitas, Taking the human out of the loop: A review of Bayesian optimization, Proc. IEEE 104 (1) (2015) 148–175.
[45] M. Kaveh, M.S. Mesgari, B. Saeidian, Orchard algorithm (OA): A new meta-heuristic algorithm for solving discrete and continuous optimization problems, Math. Comput. Simulation 208 (2023) 95–135.
[46] Y. Liu, Y. Kang, C. Xing, T. Chen, Q. Yang, A secure federated transfer learning framework, IEEE Intell. Syst. (2020).
[47] S. Saha, Y.T. Solano-Correa, F. Bovolo, L. Bruzzone, Unsupervised deep transfer learning-based change detection for HR multispectral images, IEEE Geosci. Remote Sens. Lett. (2020).
[48] S. Saha, F. Bovolo, L. Bruzzone, Unsupervised deep change vector analysis for multiple-change detection in VHR images, IEEE Trans. Geosci. Remote Sens. 57 (6) (2019) 3677–3693.