

Unsupervised domain adaptation methods for photovoltaic power forecasting

Loukas Ilias^{*,1}, Elissaios Sarmas¹, Vangelis Marinakis, Dimitris Askounis, Haris Doukas

Decision Support Systems Laboratory, School of Electrical and Computer Engineering, National Technical University of Athens, Zografou, Athens, 15780, Greece

ARTICLE INFO

Keywords:

PV power forecasting
Photovoltaics
Deep learning
Unsupervised domain adaptation
Adversarial training
Domain adversarial neural network
Margin disparity discrepancy

ABSTRACT

The accurate forecasting of photovoltaic (PV) power generation is of great significance in renewable energy systems, as it enables optimal energy management and grid stability. Despite the importance of this issue, substantial limitations still exist in the majority of existing research initiatives, which employ shallow machine learning algorithms. Recently, some studies have proposed employing convolutional and long short-term memory neural networks (LSTMs) in conjunction with transfer learning techniques; however, these approaches require that the production of PV systems is known during training. To overcome these limitations, we present the first study in the task of PV power forecasting utilizing unsupervised domain adaptation methods. Specifically, we employ two unsupervised methods, namely Domain Adversarial Neural Network and Margin Disparity Discrepancy. Both approaches use a source and a target domain during training, where the target labels of the target domain are unknown during training. We use production and weather data from seven PV systems with nominal capacities ranging from 23.52 kW to 271.53 kW, located in different areas. The findings demonstrate that our proposed architectures improve root mean squared error (RMSE), normalized RMSE, and R^2 scores over the smart persistence model across all the PV systems used for testing. Furthermore, our approaches improve the performance of the smart persistence model, with a forecast skill index reaching up to 45.35%. Our extensive experiments demonstrate that our introduced approaches offer valuable advantages over state-of-the-art ones, as the target variable of the target domain is unknown during training. We also demonstrate the robustness of our approaches by conducting a series of ablation experiments.

Code metadata

Permanent link to reproducible Capsule: <https://doi.org/10.24437/CO.8301917.v1>.

1. Introduction

The rapid deployment of photovoltaic (PV) systems has prompted a heightened need for accurate prediction of their energy production, especially in light of the stochastic formation and movement of clouds [1,2]. Developing forecasting models based on field measurements and weather data is critical for the efficient operation of PV systems. Accurate forecasts offer a host of benefits, particularly for the owners of residential and industrial PV systems. By enabling the maximization of self-consumed energy, precise forecasts can facilitate

the operation of these systems at or beyond grid-parity levels [3]. Furthermore, utility-scale PV systems depend on these models to plan plant downtime for maintenance purposes [4] and to optimize supply offers in day-ahead electricity markets, thus avoiding penalties and reduced revenues [5]. Additionally, large energy entities such as Distribution System Operators (DSOs) and Transmission System Operators (TSOs) rely on accurate predictions to manage the intermittent nature of grid-connected distributed PV systems [6]. Such forecasts improve reliability, reduce costs, and enable solar energy trading, leading to improved grid management, stability and security [7]. Accuracy predictions may also result in a decreased number of units in standby mode, which would lead to decreased operational costs for the entire power grid [8,9]. Thus, forecasting of energy production from PV systems has emerged as a crucial factor in optimizing the efficient operation of energy systems.

The code (and data) in this article has been certified as Reproducible by Code Ocean: (<https://codeocean.com/>). More information on the Reproducibility Badge Initiative is available at <https://www.elsevier.com/physical-sciences-and-engineering/computer-science/journals>.

* Corresponding author.

E-mail addresses: lilias@epu.ntua.gr (L. Ilias), esarmas@epu.ntua.gr (E. Sarmas), vmarinakis@epu.ntua.gr (V. Marinakis), askous@epu.ntua.gr (D. Askounis), h.doukas@epu.ntua.gr (H. Doukas).

¹ The first two authors contributed equally.

<https://doi.org/10.1016/j.asoc.2023.110979>

Received 20 March 2023; Received in revised form 10 October 2023; Accepted 24 October 2023

Available online 30 October 2023

1568-4946/© 2023 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

Deep learning (DL) has emerged as a powerful tool in many fields, including energy forecasting [10]. As a result, DL models have been increasingly utilized for forecasting PV power output due to their ability to capture complex, nonlinear relationships between input features and output targets [11–13]. However, the effectiveness of these models is highly dependent on the availability of sufficient training data, which is a critical challenge in the field of PV power output forecasting. DL models, while highly accurate in their predictions, are often data-hungry and require a substantial amount of data for effective training [14]. This data dependency sets DL models apart from traditional time series forecasting techniques and machine learning (ML) models. However, this dependency is offset by the improved predictive capabilities of DL models. Nevertheless, it is widely recognized that insufficient data results in under-fitting and leads to poor model performance, resulting in high variance estimation. This issue is commonly referred to as data scarcity, where the limited availability of training data poses a significant challenge in effectively training DL models to forecast PV power output [15]. Data scarcity in the context of PV production forecasting can occur in two forms. Firstly, it is a common occurrence for newly installed photovoltaic systems, as collecting sufficient power output data to train models requires a considerable amount of time. Secondly, missing data values or gaps in data may arise due to malfunctioning smart-meters, resulting in a lack of data for model training [16]. In both cases, the lack of data for DL model training presents a significant challenge. According to literature, at least one calendar year of training data is necessary to enable the model to learn seasonal patterns in PV production forecasting [17]. To mitigate the issue of data scarcity, a model trained for one location can be utilized for forecasting at another location where there is insufficient historical data. This approach leverages the transferability of DL models and has been shown to produce promising results in dealing with data scarcity.

To tackle the aforementioned limitations, we exploit two unsupervised domain adaptation methods, namely Domain-Adversarial Neural Network (DANN) [18] and Margin Disparity Discrepancy (MDD) [19]. The original DANN introduced by Ganin et al. [18] was developed for classification. Similarly, the authors in [19] designed their approach for multiclass classification tasks exploiting the cross-entropy loss function. In our study, we modify DANN and MDD, so as to ensure that these approaches are suitable for regression tasks. Regarding DANN, it consists of a feature extractor, domain classifier, and regression predictor. Similarly, MDD comprises a feature extractor, main, and auxiliary functions (neural networks). In both cases, we use a long short-term memory (LSTM) layer coupled with a transformer encoder as the feature extractor. We use data from seven PVs. Specifically, one PV is used for the source domain and the other six ones for the target domain. The predictions/regression outputs of the target domain are unknown during training. We test our proposed approaches on the target domain. Finally, we perform an ablation study and show the effectiveness of our introduced architectures. Findings indicate that domain adaption methods yield satisfactory results for the PV power forecasting in the absence of the target prediction data. There is no prior work employing domain adaptation methods in the task of PV power forecasting.

Our main contributions can be summarized as follows:

- To the best of our knowledge, this is the first study utilizing unsupervised domain adaptation methods in the task of PV power forecasting.
- We use data from seven PV systems with different nominal capacities and compare our introduced approaches with a smart persistence model.
- We perform a series of ablation experiments to prove the effectiveness and robustness of our proposed approaches.

The rest of the paper is organized as follows. Section 2 provides a literature review on short-term PV forecasting, while it also provides

a background in terms of the domain adaptation algorithms. Section 3 describes the task, provides details of the datasets used, presents the proposed methods for forecasting the PV power, and describes the evaluation metrics used for assessing the performance of the introduced approaches. Section 4 presents the experimental setup, the results of the proposed approaches, while a series of ablation experiments is also presented. Concluding remarks and plans for future work are provided in Section 5.

2. Related work

2.1. Short term PV forecasting

Short-term PV forecasting refers to the prediction of solar energy production from PV systems over a relatively short time horizon, typically ranging from a few minutes to a few hours ahead [20]. Short-term PV forecasting is essential for the efficient operation and management of PV systems, as it allows grid operators and energy managers to anticipate fluctuations in energy supply and demand and take appropriate actions to ensure grid stability and optimize energy use. Short-term PV forecasting can be accomplished using various methods, including statistical models, ML algorithms, and physical models based on meteorological data or numerical weather predictions (NWP). The accuracy of short-term PV forecasting is critical, as even small errors can lead to significant economic and environmental impacts.

While statistical models, ML algorithms [21], and physical models based on meteorological data have traditionally been used for short-term PV forecasting, recent research underscores the remarkable potential of DL techniques in outperforming these conventional methods in terms of accuracy and reliability. DL algorithms excel at handling vast datasets and capturing intricate nonlinear relationships, rendering them exceptionally well-suited for the inherently variable and dynamic nature of PV power production. Notably, the capability of DL models to harness historical data on solar irradiance and PV system performance empowers them to discern intricate patterns and relationships governing energy production. Consequently, DL models can make highly accurate predictions of energy output over short time horizons [22].

For instance, Aslam et al. [22] found that in various forecasting applications, hybrid DL algorithms consistently achieved superior prediction accuracy compared to conventional methods. Moreover, these hybrid DL schemes demonstrated enhanced resilience to data incompleteness, a challenge often encountered in real-world PV forecasting scenarios. Similarly, Korkmaz et al. [23] proposed a novel photovoltaic power forecasting system that integrates a deep Convolutional Neural Network (CNN) structure and an input signal decomposition algorithm. Experimental results from their study unequivocally affirmed that this DL-based approach outperforms traditional regression algorithms, delivering competitive and robust performance. Furthermore, Agga et al. [24] introduced a hybrid DL architecture that combines the strengths of the long short-term memory (LSTM) and CNN. In their comparative analysis against other ML models such as Linear Regression (LR), k-Nearest Neighbors (KNN), and Decision Tree Regression (DTR), the CNN-LSTM model consistently demonstrated higher prediction accuracy. These real-world examples emphasize that DL methods, such as CNN-LSTM, offer a substantial advantage over conventional models in capturing the complex dynamics of PV power production, because of their capacity to extract nuanced patterns from data.

DL models, such as recurrent neural networks (RNNs) and their special class, LSTM networks, have been widely used for short-term PV forecasting. LSTM models are capable of incorporating nonlinear, data-dependent controls into the RNN cell, ensuring that the gradient of the objective function with respect to the state signal neither vanishes nor explodes. A study by Wen et al. [25] used a deep LSTM model to predict hourly PV production at short-term horizons based on schedule, weather, and timescale variables, with the objective of optimally

dispatching the load of a community microgrid. Results showed that LSTM models were more accurate than other ML algorithms. Similarly, the authors in [26] exploited LSTMs to forecast the PV output power and compared LSTM with shallow ML algorithms, including Support Vector Regressor, and Adaptive Neuro-Fuzzy Inference System (ANFIS). Findings suggested that LSTM exhibited the lowest testing RMSE and MSE values. Also, the study [27] compared LSTMs with Multilayer Perceptron (MLP) algorithm and stated the superiority of LSTMs. Other studies, such as Lee et al. [28] and Wang et al. [29], proposed hybrid models that combined CNN and LSTM networks for day-ahead PV production forecasting. These studies concluded that the performance of CNNs and LSTMs varies for different forecasting horizons. Similarly, the authors in [30] introduced a hybrid model consisting of Dilated CNN, BiLSTM, and attention mechanism and proposed a transfer learning strategy. The authors stated that the introduced approach was superior to other models in both the accuracy and the stability.

Except for the aforementioned models, CNNs have also been popular in PV power forecasting, mainly due to their ability to process data with grid topology features and efficiently extract hidden structures and inherent features. Zang et al. [31] used popular CNN architectures, such as ResNet and DenseNet, for day-ahead PV production forecasting and compared their performance with statistical, natural, ML, and standard CNN models, reporting encouraging results. Ensemble learning, where multiple models are aggregated to make a prediction, has also been shown to be effective for short-term PV power forecasting [32,33]. The most typical studies in the literature are the ones by Zhu et al. [32] using a simple average ensemble, while Liu et al. [34] employed a weighted average ensemble model. Some studies have also considered meta-learning for PV power forecasting, such as Lateko et al. [35], who proposed a RNN-based meta-learner that combined the forecasts of five base models, and Eom et al. [36], who proposed a CNN-based feature-selective meta-learner that blended the forecasts of two statistical models and a LSTM.

While DL models have demonstrated impressive accuracy in short-term photovoltaic power forecasting, it is important to note that these models require significant amounts of training data [37]. This raises concerns regarding their reliability and robustness when applied to new and unseen scenarios, such as when limited or no training data is available. Thus, there is a pressing need to investigate the performance of DL models under such conditions and to explore techniques for improving their generalization capabilities.

2.2. Domain adaptation - Background

Over the years, there have been proposed several domain adaptation methods for minimizing the harmful effects of domain shift. These methods can be categorized into supervised and unsupervised ones. In terms of the supervised domain adaptation methods, both the source and target domains have labeled data. On the contrary, regarding unsupervised domain adaptation methods, there are labeled data only in the source domain, while the labels/predictions are unknown in the target domain during training. Also, the authors in [38] have categorized domain adaptation methods into instance-based strategy and feature-based strategy. With regards to the instance-based strategy, the source instances are reweighted in the loss during training. In terms of the feature-based strategy, this strategy aims at finding a new representation of the input features in which the source and target distributions match.

Adversarial training is widely used in the field of domain adaptation. Specifically, the authors in [18] have introduced Domain Adversarial Neural Network (DANN), which constitutes an unsupervised domain adaptation method. DANN consists of a label predictor, which predicts class labels and is used both during training and test time, and a domain classifier, which discriminates between the source and target domains and is used only during training. A gradient reversal layer is also exploited, where the input remains unchanged during forward

propagation and reverses the gradient by multiplying it by -1 during the backpropagation. An adversarial learning setting was also proposed by Tzeng et al. [39] and Shen et al. [40]. Ref. Tzeng et al. [39] chooses the adversarial loss type with respect to the domain classifier and the weight sharing strategy. The authors in [40] introduced WDGRL, a modification of DANN, which aims at minimizing the empirical Wasserstein distance in an adversarial manner. Research work Zhang et al. [19] introduced Margin Disparity Discrepancy (MDD) and provided margin-aware generalization bounds based on Rademacher complexity. Ref. Long et al. [41] introduced Conditional Domain Adversarial Networks (CDANs), which are based on a novel conditional domain discriminator conditioned on the cross-covariance of domain-specific feature representations and classifier predictions.

Maximum Mean Discrepancy (MMD) is a distance between embeddings of the probability distributions in a reproducing kernel Hilbert space [42]. MMD is a two-sample statistical test of the hypothesis that two distributions are equal based on observed samples from the two distributions. The authors in [43] modified MMD and introduced Feature Selection with MMD (f-MMD) for finding features, which contribute to the distance between the two domains the most. CORrelation ALignment (CORAL) [44] constitutes an unsupervised domain adaptation method, which minimizes domain shift by aligning the second-order statistics (covariance) of source and target distributions. This method does not require any target labels. DeepCORAL [45] constitutes an extension of CORAL, where the authors propose a method for learning a nonlinear transformation that aligns correlations of layers' activations in deep neural networks

2.3. Related work review findings

In reviewing the existing body of literature on short-term PV power forecasting, it becomes evident that prior research has predominantly centered around the utilization of ML techniques. Notably, these approaches have commonly operated within the context of supervised learning, relying on access to output variables, i.e., known power values, historical weather data etc. during training. Furthermore, a shared characteristic among these methodologies is their substantial data requirements, demanding ample training data for model development.

While these existing efforts have contributed valuable insights and techniques to the field of PV power forecasting, a distinct gap emerges. Specifically, the gap centers on the challenge of adapting forecasting models to scenarios where power output remains unknown during the training phase. This pivotal gap emphasizes the need for novel approaches capable of predicting PV power production in short term horizons without the luxury of output data for training. In the real world, this can be the case for newly installed PV systems, or in cases that sensors have not been installed during the PV installation.

Therefore, in this study we depart from the conventional paradigm and frame PV power forecasting as an unsupervised domain adaptation task. Through a series of rigorous experiments, our study aims to demonstrate the noteworthy advantages offered by our proposed domain adaptation approaches. By enabling accurate PV power predictions without requiring access to output variables during training, our research addresses a critical need within the field and pioneers an innovative path towards more robust and adaptable forecasting models. This distinct approach showcases the novelty and practical relevance of our study, contributing to the growing body of knowledge in PV power forecasting under data scarcity scenarios.

3. Materials and methods

3.1. Task

We define the PV power forecasting task as an unsupervised domain adaptation task. Specifically, in the unsupervised domain adaptation

Table 1

Overview of the dataset used.

	Datetime	Humidity	Temperature	cloudcover	windspeed kmph	Solar W/m ²	Diffuse solar W/m ²	Prodüzida	Year	Month	Day	timestamp
0	2018-08-01 02:00:00	82	17	0	17	0.0000	0.0000	0.00	2018	8	1	02:00:00
1	2018-08-01 03:00:00	84	17	0	17	0.0000	0.0000	0.00	2018	8	1	03:00:00
2	2018-08-01 04:00:00	86	17	1	16	0.0000	0.0000	0.00	2018	8	1	04:00:00
3	2018-08-01 05:00:00	87	17	1	15	0.0000	0.0000	0.00	2018	8	1	05:00:00
4	2018-08-01 06:00:00	88	17	2	14	0.0000	0.0000	0.00	2018	8	1	06:00:00
5	2018-08-01 07:00:00	79	19	1	15	3.4164	2.4406	0.00	2018	8	1	07:00:00
6	2018-08-01 08:00:00	70	21	1	15	107.1310	38.5490	0.75	2018	8	1	08:00:00
7	2018-08-01 09:00:00	61	23	0	15	292.9205	66.6320	4.75	2018	8	1	09:00:00
8	2018-08-01 10:00:00	55	25	0	14	487.4875	79.1451	9.50	2018	8	1	10:00:00
9	2018-08-01 11:00:00	49	27	0	13	660.1908	92.4183	13.50	2018	8	1	11:00:00
10	2018-08-01 12:00:00	42	30	0	12	793.2076	111.8966	15.75	2018	8	1	12:00:00

task, we have access to data from two domains-PV systems. The one domain constitutes the source domain with features $X_s \in \mathcal{R}^{n \times T}$, where n denotes the lag variable and T the number of features, and target labels $Y_s \in \mathcal{R}^I$. The second domain constitutes the target domain, where we do not have access to the output variable during training. Target domain consists of a feature set, namely $X_t \in \mathcal{R}^{n \times T}$. We use as source domain a PV system, which is denoted by PV1 and is located in Lisbon with a nominal capacity accounting for 23.52 kW. We use as target domains the rest of PV systems, i.e., PV2 – PV7, with nominal capacities ranging from 30 kW to 271.53 kW. The task is to predict the power of the target domains-PV systems.

3.2. Data

We exploit data from seven PV systems consisting of (hourly) PV production data and weather data. Specifically, production data are collected from the solar PV systems of a Portuguese energy community, while weather data are extracted via a local meteorological station² and the Copernicus Atmosphere Data Store.³ The PV systems are located in 4 cities in Portugal. Specifically, four PVs are located in Lisbon and the other PVs are located in Setubal, Faro, and Braga. In Table 1, we present an overview of the datasets used. Next, we apply some preprocessing steps. The selected features given as input to our proposed approaches are temperature, humidity, solar irradiance, PV production, one-hot encoding representation of the month of the year and sine/cosine transformation of the hour of day. The PVs have a nominal capacity ranging from 23.52 kW to 271.53 kW.

3.3. Methodology

In this section, we describe our proposed approaches for forecasting the PV power.

3.3.1. DANN

The original DANN introduced by Ganin et al. [18] was developed for classification. Specifically, the authors in [18] added a label classifier after the feature extractor. In this study, we modify the DANN framework introduced by Ganin et al. [18] for adapting it to our regression task, i.e., forecasting the PV power. Our proposed method is illustrated in Fig. 1, while Table 2 reports the parameters of the DANN architecture. Specifically, our proposed DANN framework consists of three main parts, feature extractor, regressor, and domain classifier. Specifically, the feature extractor aims to extract features that are domain-invariant, which is an adversarial task for the domain classifier. The domain classifier determines whether the input belongs to the source or target domain. The regressor is trained to correctly estimate the PV power of the source data. These three main parts are described in detail below:

Feature Extractor: The first part is the feature extractor denoted by $G_f(\cdot; \theta_f)$.

First, we pass $x \in \mathcal{R}^{n \times T}$ through an LSTM layer [46] consisting of 512 units and ReLU activation function. n denotes the lag and is equal to five, while $T = 19$ is the number of features. We omit the dimension corresponding to the batch size for the sake of simplicity. Let the output of the LSTM layer denoted by $z \in \mathcal{R}^{n \times D}$, where $D = 512$. We pass z through the encoder layer of the transformer introduced by Vaswani et al. [47]. Specifically, we define z as the Query (Q), Key (K), and Value (V) matrices.

For capturing information from different spaces and strengthening the feature discrimination, we use the Multi-Head self-attention mechanism (MHA). Specifically, the MHA component first converts the original Q, K, V matrices into H sub-matrices of the same size.

$$Q^i = QW_i^Q, K^i = KW_i^K, V^i = VW_i^V, \quad (1)$$

where $W_i^Q \in \mathcal{R}^{D \times d_q}$, $W_i^K \in \mathcal{R}^{D \times d_k}$, and $W_i^V \in \mathcal{R}^{D \times d_v}$ are learnable parameters. As mentioned in [47,48], we set $d_q = d_k = d_v = \frac{D}{H}$.

Therefore, we calculate self-attention operations on H subspaces in parallel as follows:

$$\begin{cases} head_1 = Attention(Q^1, K^1, V^1), \\ \vdots \\ head_i = Attention(Q^i, K^i, V^i), \\ \vdots \\ head_H = Attention(Q^H, K^H, V^H), \end{cases} \quad (2)$$

where $Attention$ corresponds to the self-attention mechanism and is given by the equation below:

$$Attention(Q^i, K^i, V^i) = softmax\left(\frac{Q^i K^{iT}}{\sqrt{d_k}}\right) V^i. \quad (3)$$

Finally, the results learned by the multi-head attention are concatenated as output and projected to dimensionality d_O . Formally:

$$MHA(Q, K, V) = Concat(head_1, head_2, \dots, head_H)W^0, \quad (4)$$

where $W^0 \in \mathcal{R}^{D \times d_O}$. Specifically, $d_O = D$.

Next, as illustrated in Fig. 1, we add z and $MHA(Q, K, V)$ and pass the resulting matrix through a layer normalization [49]. Formally:

$$x = LayerNorm(z + MHA(Q, K, V)). \quad (5)$$

Next, we employ a fully connected feed-forward network consisting of two linear transformations with a ReLU activation function in between. Formally:

$$MLP(x) = W_2(reLU(W_1x + b_1)) + b_2, \quad (6)$$

where $W_1 \in \mathcal{R}^{D \times 4D}$, $W_2 \in \mathcal{R}^{4D \times D}$.

Next, we add the outputs of Eqs. (5) and (6) and pass the resulting matrix through a layer normalization.

Let the output of the transformer encoder be $z \in \mathcal{R}^{n \times D}$.

² <https://www.wunderground.com>.

³ <https://ads.atmosphere.copernicus.eu>.

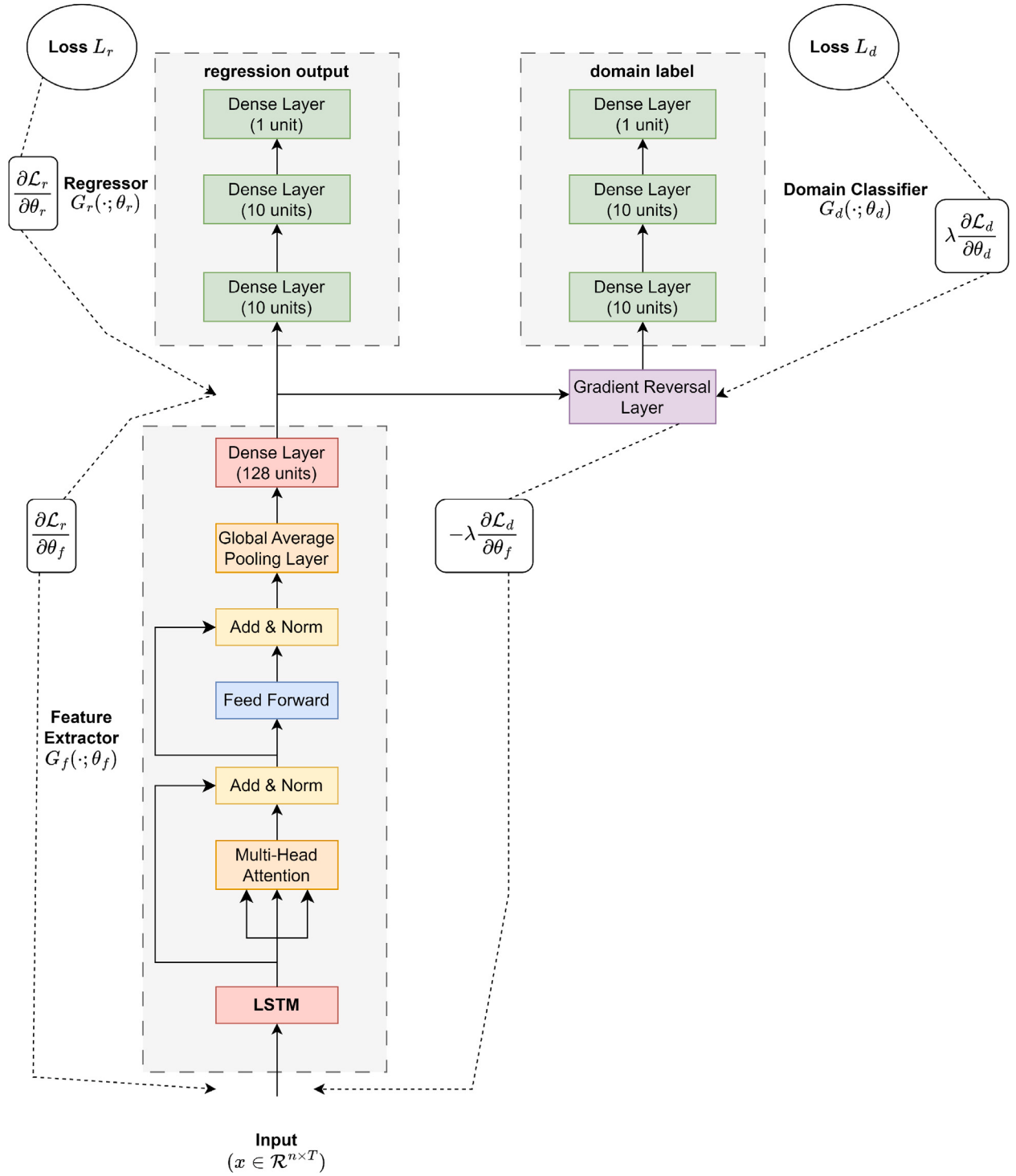


Fig. 1. Our proposed DANN approach. DANN comprises a feature extractor, regressor, and domain classifier.

Next, we pass z through a global average pooling layer and get $p \in \mathcal{R}^D$. Finally, p is passed through a ReLU activated dense layer consisting of 128 units.

Regressor - Predictor:

For obtaining the final prediction, we use a regressor denoted by $G_r(\cdot; \theta_r)$. Specifically, the regressor consists of two dense layers with 10

units and a ReLU activation function. The last layer consists of one unit with a linear activation function.

Domain Classifier: This is the most critical part of our architecture. This domain classifier is denoted by $G_d(\cdot; \theta_d)$ with parameters θ_d and consists of two ReLU activated dense layers with 10 layers each. The output layer consists of one unit with sigmoid activation function. The purpose of the domain classifier is to discriminate features extracted

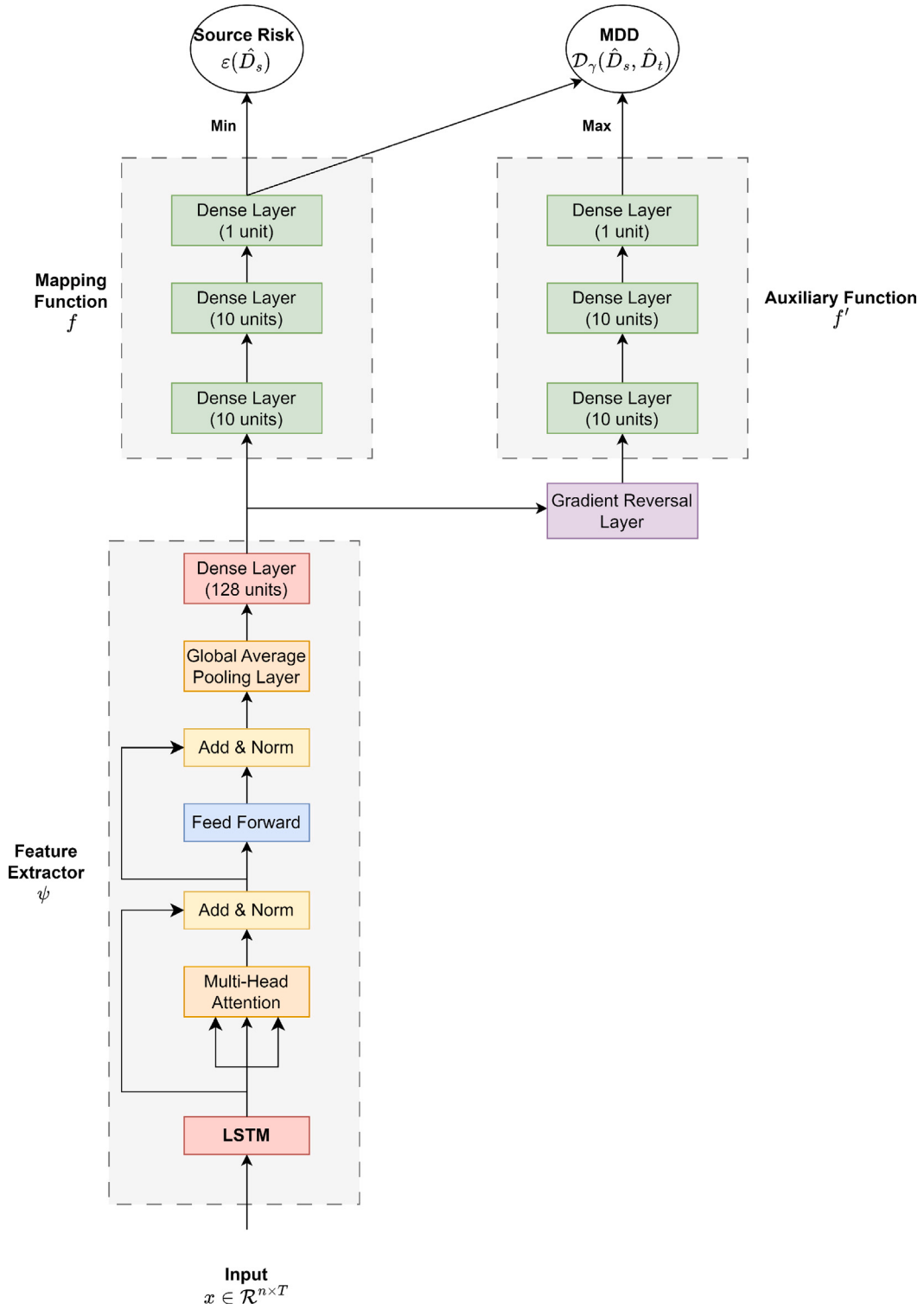


Fig. 2. Our proposed MDD approach. MDD consists of a feature extractor, mapping function, and auxiliary function.

by $G_f(\cdot; \theta_f)$ between source and target domains. The feature extractor aims to confuse the discriminator by providing features without domain characteristics. Therefore, domain adversarial training helps the model adapt to the target domain increasing its performance.

Formally, we note the prediction loss and the domain loss respectively as follows:

$$\mathcal{L}_r^i(\theta_f, \theta_r) = \mathcal{L}_r(G_r(G_f(x_i; \theta_f); \theta_r), y_i), \quad (7)$$

where the regression loss corresponds to the mean square loss.

$$\mathcal{L}_d^i(\theta_f, \theta_d) = \mathcal{L}_d(G_d(G_f(x_i; \theta_f); \theta_d), d_i), \quad (8)$$

where the domain loss refers to the binary cross entropy loss.

During training procedure, we optimize:

$$E(\theta_f, \theta_r, \theta_d) = \frac{1}{T} \sum_{i=1}^T (\mathcal{L}_r^i(\theta_f, \theta_r)) - \lambda \mathcal{L}_d^i(\theta_f, \theta_d), \quad (9)$$

Table 2
Parameter setting of our DANN.

	Layer	Parameters
Feature extractor	LSTM	512 units ReLU activation return_sequences = True
	Transformer encoder	1 layer num_heads = 4
	Global average pooling Dense layer	128 units
Regressor	Dense	10 units
	Dense	10 units
	Dense	1 unit
Domain classifier	Dense	10 units
	Dense	10 units
	Dense	1 unit

by finding the saddle point $\hat{\theta}_f, \hat{\theta}_r, \hat{\theta}_d$, such that:

$$(\hat{\theta}_f, \hat{\theta}_r) = \operatorname{argmin}_{\theta_f, \theta_r} E(\theta_f, \theta_r, \hat{\theta}_d), \quad (10)$$

$$\hat{\theta}_d = \operatorname{argmax}_{\theta_d} E(\hat{\theta}_f, \hat{\theta}_r, \theta_d). \quad (11)$$

The saddle point can be found as a stationary point of the following gradient updates:

$$\theta_f \leftarrow \theta_f - \mu \left(\frac{\partial \mathcal{L}_r^i}{\partial \theta_f} - \lambda \frac{\partial \mathcal{L}_d^i}{\partial \theta_f} \right), \quad (12)$$

$$\theta_r \leftarrow \theta_r - \mu \frac{\partial \mathcal{L}_r^i}{\partial \theta_r}, \quad (13)$$

$$\theta_d \leftarrow \theta_d - \mu \lambda \frac{\partial \mathcal{L}_d^i}{\partial \theta_d}, \quad (14)$$

where μ is the learning rate and is equal to 10^{-3} .

As we observe in Eq. (12), the gradients from the regression and domain predictors are subtracted, instead of being summed. As mentioned in [18], such a reduction is attained by introducing the gradient reversal layer (GRL). Specifically, during the forward propagation, the GRL acts as an identity transformation, whereas during the backward propagation, the GRL takes the gradient from the subsequent level, multiplies it by -1 , and passes it to the preceding layer.

Finally, the feature extractor will generate domain-invariant features that assist the regressor in predicting the output value of the input data.

3.3.2. MDD

For evaluating the discrepancy between source and target domains, we use MDD [19], which measures the difference of distributions. Our proposed method is illustrated in Fig. 2, while Table 3 reports the parameters of the MDD approach.

The minimization problem based on the margin loss and MDD can be described via the equation below:

$$\min_{f \in F} = \operatorname{err}_{\hat{D}_s}^{(\rho)}(f) + d_{f, F}^{(\rho)}(\hat{D}_s, \hat{D}_t), \quad (15)$$

where $\hat{D}_s$ denotes the source sample, $\hat{D}_t$ denotes the target sample, $\operatorname{err}_{\hat{D}_s}^{(\rho)}(f)$ indicates the source domain margin error, $d_{f, F}^{(\rho)}(\hat{D}_s, \hat{D}_t)$ denotes the empirical MDD, F denotes the hypothesis space, and f denotes the scoring function.

An adversarial learning algorithm is designed for solving this problem. This is accomplished by introducing an auxiliary classifier f' sharing the same hypothesis space with f . The optimization problem in the adversarial learning framework can be described via the equations below:

$$\begin{cases} \min_{f, \psi} \varepsilon(\hat{D}_s) + \eta D_\gamma(\hat{D}_s, \hat{D}_t), \\ \max_{f'} D_\gamma(\hat{D}_s, \hat{D}_t). \end{cases} \quad (16)$$

Table 3
Parameter setting of our MDD.

	Layer	Parameters
Feature extractor	LSTM	512 units ReLU activation return_sequences = True
	Transformer encoder	1 layer num_heads = 4
	Global average pooling Dense Layer	128 units
Mapping function	Dense	10 units
	Dense	10 units
	Dense	1 unit
Auxiliary function	Dense	10 units
	Dense	10 units
	Dense	1 unit

Concretely, we define:

$$\varepsilon(\hat{D}_s) = \mathbb{E}_{(x^s, y^s) \sim \hat{D}_s} L(f(\psi(x^s)), y^s) \quad (17)$$

$$\begin{aligned} D_\gamma(\hat{D}_s, \hat{D}_t) = & \mathbb{E}_{x^t \sim \hat{D}_t} L'(f'(\psi(x^t)), f(\psi(x^t))), \\ & - \gamma \mathbb{E}_{x^s \sim \hat{D}_s} L(f'(\psi(x^s)), f(\psi(x^s))), \end{aligned} \quad (18)$$

where ψ denotes the feature extractor. We use the same feature extractor with DANN, as described in Section 3.3.1. f and f' denote the mapping and auxiliary function. f and f' represent deep neural networks with two dense layers consisting of two units and one dense layer with one unit providing the output. In terms of losses denoted by L and L' , we use the mean squared error. The feature extractor ψ is trained for minimizing the discrepancy loss term through a gradient reversal layer as introduced in [50].

3.4. Evaluation metrics

To effectively evaluate the accuracy and precision of PV power forecasting models, appropriate error metrics must be employed. In this study, we employ several commonly used metrics to evaluate the performance of the proposed models on target data. These metrics include the Root Mean Squared Error (RMSE), the Normalized Root Mean Squared Error (nRMSE), the Mean Bias Error (MBE), the Coefficient of Determination (R^2), and the Forecast Skill Index (FSI). Each metric provides unique insights into the performance of the forecasting models and allows for a comprehensive evaluation of their accuracy. In this section, we provide detailed explanations of each metric, including their formulas and interpretation. By utilizing multiple metrics, we can gain a deeper understanding of the strengths and weaknesses of the forecasting models and identify areas for improvement.

The RMSE is one of the most widely used metrics for evaluating the accuracy of forecasting models. It measures the differences between the predicted and actual values, providing a single numerical value that quantifies the amount of error produced by the model. The RMSE is calculated by taking the square root of the mean of the squared differences between the predicted and actual values. A lower RMSE value indicates better performance of the forecasting model in terms of accuracy. The equation is given as:

$$\operatorname{RMSE} = \sqrt{\frac{1}{n} \sum_{t=1}^n (y_t - \hat{y}_t)^2}, \quad (19)$$

where n is the total number of data points, y_t is the actual value, and $\hat{y}_t$ is the predicted value at time t . A lower RMSE indicates a better performance of the model.

The nRMSE is a normalized version of RMSE that is often used in PV power forecasting studies. By dividing the RMSE value by the mean value of the actual data, nRMSE provides a more meaningful

comparison of model performance across different scales of data. This is particularly useful when evaluating models that predict a wide range of values, as a model with a lower RMSE value may not necessarily be the best model if it is predicting a smaller range of values. The equation is given as:

$$nRMSE = \frac{RMSE}{\bar{y}}, \quad (20)$$

where $\bar{y}$ is the mean value of the actual data.

Complementary to the above-mentioned metrics, MBE is a metric that measures the average difference between the predicted and actual values in a forecasting model. In the context of PV power forecasting, MBE provides insights into any systematic biases in the model's predictions. A positive MBE value indicates that the model is overestimating the actual values, while a negative MBE value indicates that the model is underestimating the actual values. The equation is given as:

$$MBE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i). \quad (21)$$

Finally, the Coefficient of Determination (R^2) measures the proportion of variance in the actual data that is explained by the forecasting model, providing insights into the ability of the model to accurately capture the underlying patterns in the data. R^2 ranges from 0 to 1, where a value of 1 indicates that the model perfectly predicts the actual data. The equation is given as:

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2}. \quad (22)$$

While metrics such as RMSE, nRMSE, MBE, and R^2 are widely used to evaluate the accuracy and bias of forecasting models, they may not be sufficient to compare models that operate on different datasets, locations, or horizons [51]. In this regard, the use of a forecast skill score, such as the Forecast Skill Index (FSI), can provide a more comprehensive and objective evaluation of the proposed models' performance [52]. FSI is a metric that compares the performance of forecasting models with a smart persistence model, and it is calculated as follows:

$$FSI = 1 - \frac{RMSE_{proposed}}{RMSE_{reference}}. \quad (23)$$

By using RMSE as the metric, FSI provides a normalized and location-independent measure of the forecasting model's performance, enabling the comparison of models operating in different contexts [15]. The reference model is typically a persistence model that assumes that the PV power output will remain constant from the previous hour. However, using a smart persistence model is highly encouraged [53], allowing for the comparison of the proposed models with a better reference model. In this study we use the smart persistence model proposed by Pedro and Coimbra [54], which estimates the expected power output under clear-sky conditions based on the following equation:

$$\hat{y}(t + \Delta t) = \begin{cases} y_{c-s}(t + \Delta t), & \text{if } y_{c-s}(t) = 0, \\ y_{c-s}(t + \Delta t) \frac{y(t)}{y_{c-s}(t)}, & \text{otherwise.} \end{cases} \quad (24)$$

4. Experiments and results

4.1. Experimental setup

Before training the proposed architectures, we scale both predictor and target variables to a [0,1] scale. After training, we apply a post-processing step, where predicted scaled outputs are transformed to the actual range. Regarding data of the target domain, we use one year of data (8760 h) for training, while the rest of the data is used for testing. We use λ of Eq. (14) equal to 0.1. We set the number of heads equal to 4 in the multi-head self-attention. We use η of Eq. (16) equal to 0.1. We use margin (γ) of Eq. (18) equal to 3. We use the Adam optimizer

with a learning rate of $1e-3$. We use 20% of the training set as the validation one. We use *EarlyStopping* and stop training if the validation loss has stopped decreasing for 10 consecutive epochs. We set the number of epochs and batch size equal to 200 and 128 respectively. We use the data from PV1 as source data. All the other PV data are used as target data. We train and evaluate our proposed approaches 20 times and report the mean results. We use the Awesome Domain Adaptation Python Toolbox (ADAPT) [55] and Tensorflow [56] library. All experiments are conducted on a single Tesla P100-PCIE-16 GB GPU.

4.2. Results

This section reports the results of an empirical evaluation of the proposed DL models for forecasting PV power production, as compared to a baseline smart persistence model. Fig. 3 provides a visual comparison of the performance of the proposed DANN and MDD models in forecasting PV power production, as compared to the actual values. The horizontal axis represents the hourly time-step, covering an indicative evaluation period, while the vertical axis displays the PV power production. By comparing the predicted values with the actual data, we can visually assess the accuracy of the proposed models. The figure allows us to observe the trends in the predicted values, as well as the differences between the two models and their performance relative to the actual data.

Table 4 shows the performance of the proposed DL models (DANN and MDD) and the Smart Persistence model for six different PV systems. The results demonstrate that the DL models outperform the Smart Persistence model in most cases, achieving lower values of RMSE, MBE, and nRMSE and higher values of R^2 . For instance, for the PV2 system, both DANN and MDD models outperform the SP model, achieving a lower RMSE of 2.84 and 2.70 kWh compared to 3.27 kWh for the SP model. Moreover, the DANN model achieves the highest R^2 of 93.44%, while the MDD model has the highest FSI of 17.43%. Similar results can be observed for the other PV systems, where the DL models outperform the SP model in all evaluation metrics.

In relation to the comparison of the two proposed DL models, it is observed that the results exhibit variability for each PV system. Fig. 4 presents a comparative boxplot summarizing the performance of the proposed approaches for the six target PV systems based on the RMSE metric. More specifically, for PV2, the MDD model presents a marginally superior RMSE than the DANN model, with difference of 0.14 kWh. In contrast, the DANN model yields a higher R^2 and lower nRMSE values, indicating a better overall match with the data. Regarding PV3, the DANN model performs better than the MDD model, showing a lower RMSE and a higher R^2 , implying a better fit to the data, while the MDD model presents a lower MBE, indicating a lesser tendency towards underestimation of actual values. For PV4, the MDD model slightly outperforms the DANN model in terms of RMSE, with a difference of 0.08 kWh, whereas the DANN model has a lower MBE, denoting an improved overall fit to the data, and both models exhibit similar R^2 values. Concerning PV5, the MDD model produces better RMSE and nRMSE values than the DANN model, with differences of 0.29 kWh and 3.44%, respectively, whereas the DANN model displays a higher R^2 , indicating a better fit to the data. For PV6, both models demonstrate comparable RMSE, MBE, and R^2 values, yet the MDD model yields a marginally lower nRMSE, implying a slightly superior fit to the data. Finally, for PV7, the MDD model presents better RMSE and nRMSE values than the DANN model, with differences of 0.35 kWh and 4.00%, respectively, while the DANN model exhibits a higher R^2 .

In conclusion, the difference in performance between DANN and MDD is not consistent across all PV systems. For example, DANN performs better than MDD in PV systems 2, 4, and 7, while MDD performs better in PV systems 3, 5, and 6. This suggests that the choice of approach may depend on the specific PV system being forecasted. Moreover, the nRMSE values range from 31.52% (PV 7, MDD) to 62.15% (PV 3, Smart Persistence), indicating that the accuracy of the

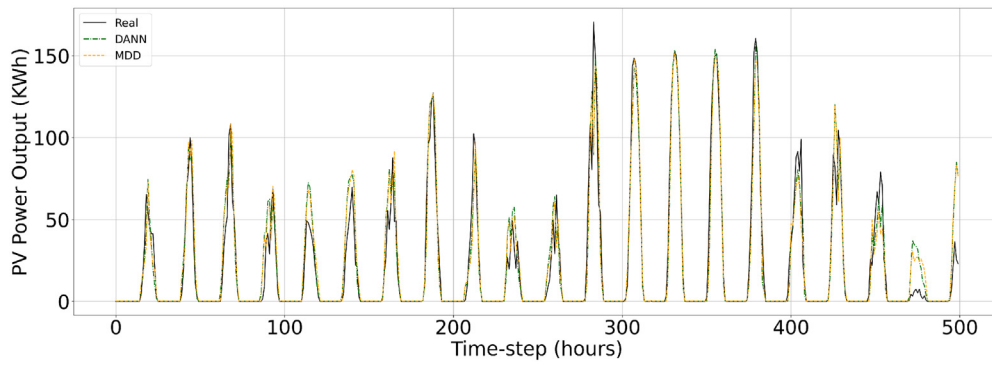


Fig. 3. Example illustrating how our proposed PV power forecasting models perform in comparison with the actual values. The horizontal axis indicates the hourly time-step of the evaluation period, while the vertical axis shows the PV power production.

Table 4

Average forecasting performance (accuracy) of the smart persistence model and our proposed approaches. $RMSE$ and MBE are measured in kWh, while for $nRMSE$, R^2 , and FSI the percentage is given for each model.

PV	Metric	DANN	MDD	Smart persistence
PV_2	$RMSE$ (kWh)	2.84	2.70	3.27
	MBE (kWh)	-0.03	0.04	0.17
	$nRMSE$ (%)	40.02	38.02	46.18
	R^2 (%)	93.44	94.13	91.37
	FSI (%)	13.14	17.43	
PV_3	$RMSE$ (kWh)	12.82	13.78	17.32
	MBE (kWh)	2.55	1.88	0.02
	$nRMSE$ (%)	46.41	49.89	62.15
	R^2 (%)	91.89	90.59	85.27
	FSI (%)	25.98	20.43	
PV_4	$RMSE$ (kWh)	2.07	2.15	2.69
	MBE (kWh)	0.64	0.76	-0.82
	$nRMSE$ (%)	44.56	46.30	57.84
	R^2 (%)	91.26	90.64	85.40
	FSI (%)	23.04	20.07	
PV_5	$RMSE$ (kWh)	3.17	2.88	5.27
	MBE (kWh)	-0.37	0.09	0.35
	$nRMSE$ (%)	37.72	34.31	62.75
	R^2 (%)	93.11	94.45	81.51
	FSI (%)	39.84	45.35	
PV_6	$RMSE$ (kWh)	7.29	7.12	8.59
	MBE (kWh)	0.54	-0.71	-0.04
	$nRMSE$ (%)	44.20	43.16	52.11
	R^2 (%)	91.61	92.11	88.58
	FSI (%)	15.13	17.11	
PV_7	$RMSE$ (kWh)	3.16	2.81	4.59
	MBE (kWh)	0.64	0.42	0.15
	$nRMSE$ (%)	35.52	31.52	51.49
	R^2 (%)	94.20	95.54	88.19
	FSI (%)	31.15	38.77	

forecasts varies considerably across PVs and approaches. PV_3 appears to be the most challenging to forecast accurately, while PV_7 appears to be the easiest. In general, the proposed DL approaches appear to be significantly more accurate than Smart Persistence in terms of all error metrics, suggesting that they may be more useful in real-world applications where accurate PV generation forecasting is critical.

4.3. Ablation study

In this section, we perform a series of ablation experiments to show the effectiveness and robustness of our introduced architectures. Results of the ablation studies are illustrated in Figs. 5 and 6. Specifically, we use the PV_7 as the target domain.

In terms of DANN architecture, first we vary the number of heads of the encoder part of the transformer. Results show that the larger the number of heads is, the better performance is obtained. Specifically,

we observe that setting the number of heads equal to 1 results in an RMSE of 3.59, while a number of heads equal to 2 results in an RMSE of 3.19. Secondly, we vary the number of layers of the encoder part of the transformer. Findings suggest that as the number of layers increases, the RMSE presents a surge also. This can be justified by the fact that the architecture is complex and overfits. Finally, we vary the λ parameter and show that a value of 0.10 yields the best RMSE. On the contrary, λ values of 0.001 and 0.01 yield an RMSE of 3.70 and 3.45 respectively.

Regarding the MDD architecture, first we vary the number of heads. Findings suggest that the larger the number of heads gets, the better RMSE is attained. Secondly, we vary the number of layers of the Transformer encoder. Results show that as the number of layers increases, the RMSE increases. To be more precise, an RMSE of 3.31 is obtained by setting the number of layers to two, while an RMSE of 3.79 is yielded by using four encoder layers. Finally, we vary the value of γ parameter from 1 to 4. Findings show that RMSE values of 3.96, 3.43, and 2.97 are attained by setting γ equal to 1, 2, and 4 respectively.

5. Conclusions and future work

5.1. Conclusions

This study marks a significant leap forward in the application of unsupervised domain adaptation methods in PV power forecasting. We presented the first study applying unsupervised domain adaptation methods to enhance the precision of PV power production forecasting especially in cases with data scarcity. The significance of our contribution becomes evident when we consider the novel aspects we have introduced to this field. Our research brings forth two pioneering unsupervised domain adaptation methods, the DANN and the MDD, meticulously fine-tuned for regression tasks within PV power forecasting. These methods employ a feature extractor architecture that integrates an LSTM layer with a transformer encoder. While DANN incorporates a feature extractor, domain classifier, and regression predictor, MDD integrates primary and auxiliary functions within neural networks. Our work extends these adaptation techniques into an entirely uncharted territory—the realm of accurate PV power production forecasting.

Our methodology leveraged weather and production data from a single PV system as the source domain, seamlessly integrating it with data from six diverse PV systems that served as our target domain. We established a baseline using the smart persistence model and subsequently introduced innovative architectures custom-tailored to this specific task. Our findings unequivocally underscore the superiority of our proposed approaches over the smart persistence model. To be precise, our models consistently exhibited a substantial reduction in RMSE across all six PV systems, with reductions ranging from 0.43 kWh to as much as 4.5 kWh. Moreover, across the various performance metrics we evaluated, including RMSE, MBE, $nRMSE$, and R^2 , our

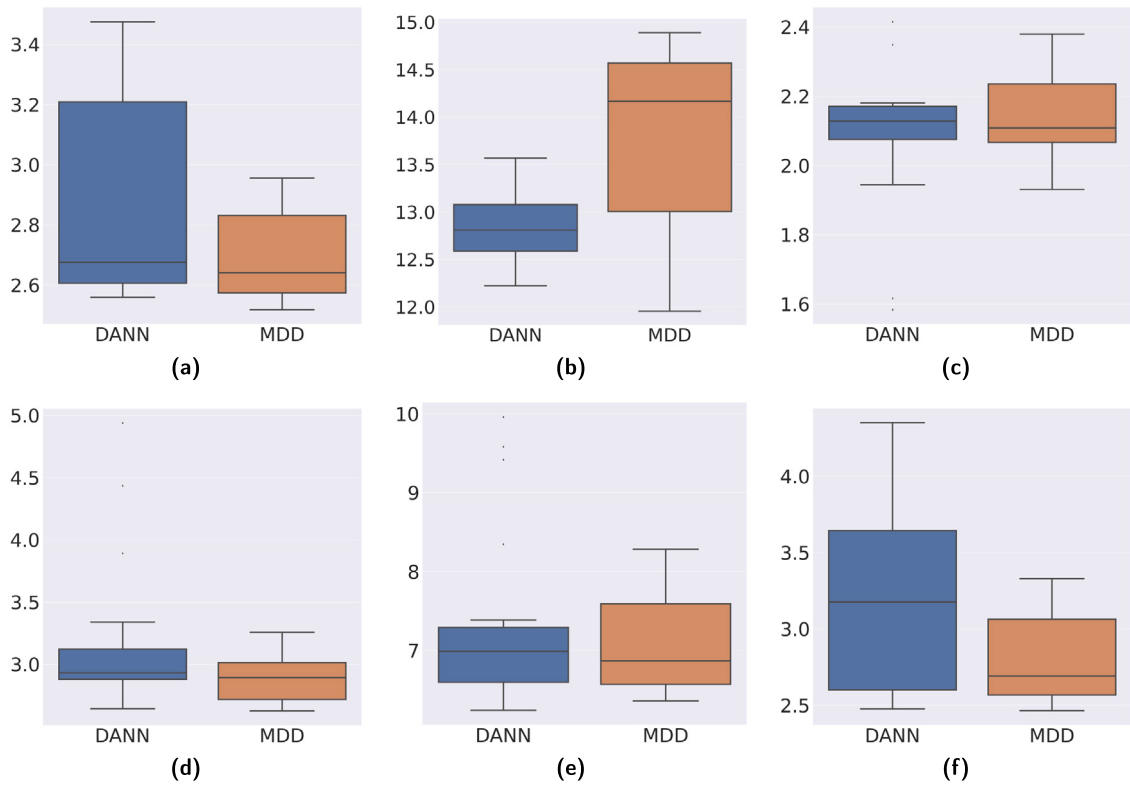


Fig. 4. Boxplot that summarizes the performance of the proposed approaches for the six target PV systems based on the $RMSE$: (a) PV2 (b) PV3 (c) PV4 (d) PV5 (e) PV6 (f) PV7.

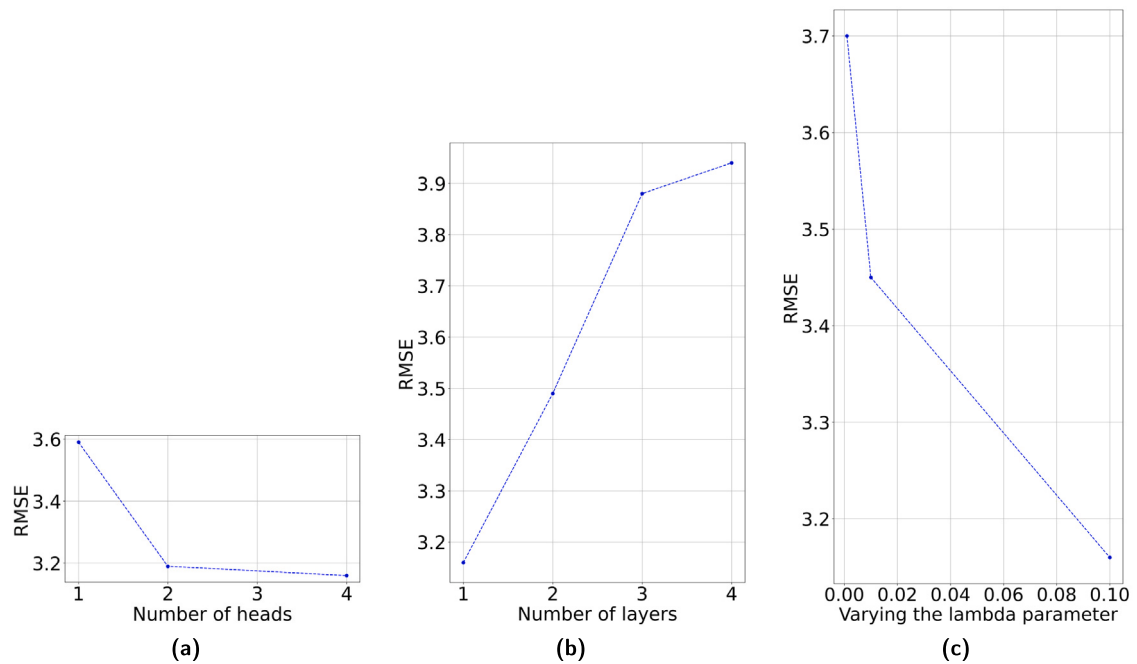


Fig. 5. Ablation study. DANN. (a) Varying the number of heads. (b) Varying the number of Transformer-Encoder Layers (c) Varying the lambda parameter.

proposed DL models consistently outperformed the Smart Persistence model, achieving lower RMSE values by an average of 0.25 kWh, higher R^2 values by an average of 8.21%, and reducing nRMSE by an average of 15.83% across the six PV systems.

5.2. Future work

This work not only breaks new ground but also lays out promising paths for future exploration in the RES forecasting domain. Firstly,

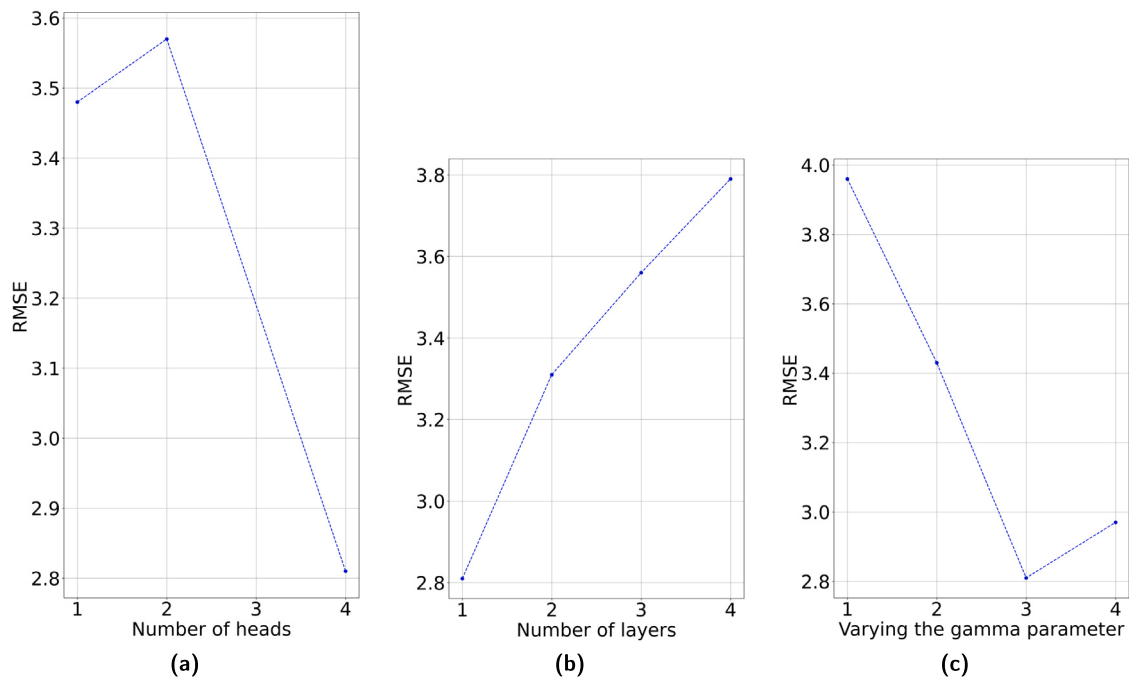


Fig. 6. Ablation study. MDD. (a) Varying the number of heads. (b) Varying the number of Transformer-Encoder Layers (c) Varying the gamma parameter.

our intention is to diversify our domain adaptation methodologies by integrating semi-supervised approaches, potentially harnessing additional labeled data to further elevate the precision of our forecasting. Additionally, we are planning to extend our techniques to encompass short-term solar irradiance forecasting, incorporating the use of sky images. This expansion into a related, albeit distinct, domain holds the potential to capitalize on our expertise in unsupervised domain adaptation and offer invaluable insights into the realm of renewable energy forecasting.

Moreover, the discernible variance in model performance across different PV systems warrants deeper investigation into the underlying factors contributing to this variability. Our future work may entail exploring system-specific adaptations to fine-tune forecast accuracy. Furthermore, we advocate for robustness testing of our models, which would account for real-world uncertainties, such as sudden weather fluctuations or equipment malfunctions. Finally, strategic collaborations with industry stakeholders and grid operators present the opportunity to facilitate the practical integration of our models into real-world scenarios, making meaningful contributions to the ongoing efforts to optimize the integration of renewable energy sources into existing power grids.

CRediT authorship contribution statement

Loukas Ilias: Conceptualization, Methodology, Software, Validation, Formal analysis, Visualization, Investigation, Data curation, Writing – original draft, Writing – review & editing. **Elissaios Sarmas:** Conceptualization, Methodology, Software, Validation, Formal analysis, Visualization, Investigation, Data curation, Writing – original draft, Writing – review & editing. **Vangelis Marinakis:** Funding acquisition, Supervision, Project administration, Writing – review & editing. **Dimi-iris Askounis:** Supervision, Project administration, Writing – review & editing. **Haris Doukas:** Funding acquisition, Supervision, Project administration, Writing – review & editing.

Declaration of competing interest

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

We confirm that we have given due consideration to the protection of intellectual property associated with this work and that there are no impediments to publication, including the timing of publication, with respect to intellectual property. In so doing we confirm that we have followed the regulations of our institutions concerning intellectual property.

Data availability

Data will be made available on request.

Acknowledgments

The work presented is based on research conducted within the framework of the project “Modular Big Data Applications for Holistic Energy Services in Buildings (MATRYCS)”, of the European Union’s Horizon 2020 research and innovation programme under grant agreement no. 101000158 (<https://matrycs.eu/>), of the Horizon 2020 European Commission project BD4NRG under grant agreement no. 872613 (<https://www.bd4nrg.eu/>) and of the Horizon Europe European Commission project DigiBUILD under grant agreement no. 101069658 (<https://digibuild-project.eu/>). The authors wish to thank the Coopérnico team, whose contribution, helpful remarks and fruitful observations were invaluable for the development of this work. The content of the paper is the sole responsibility of its authors and does not necessary reflect the views of the EC.

References

- [1] M. Braun, T. Stetz, R. Bründlinger, C. Mayr, K. Ogimoto, H. Hatta, H. Kobayashi, B. Kroposki, B. Mather, M. Coddington, et al., Is the distribution grid ready to accept large-scale photovoltaic deployment? State of the art, progress, and future prospects, *Prog. Photovolt.: Res. Appl.* 20 (6) (2012) 681–697.
- [2] D.W. Van der Meer, J. Widén, J. Munkhammar, Review on probabilistic forecasting of photovoltaic power production and electricity consumption, *Renew. Sustain. Energy Rev.* 81 (2018) 1484–1512.
- [3] R. Wang, S. Hasanefendic, E. Von Hauff, B. Bossink, The cost of photovoltaics: Re-evaluating grid parity for PV systems in China, *Renew. Energy* 194 (2022) 469–481.

- [4] J. Ramirez-Vergara, L.B. Bosman, E. Wollega, W.D. Leon-Salas, Review of forecasting methods to support photovoltaic predictive maintenance, *Clean. Eng. Technol.* (2022) 100460.
- [5] M. David, J. Boland, L. Cirocco, P. Lauret, C. Voyant, Value of deterministic day-ahead forecasts of PV generation in pv+ storage operation for the Australian electricity market, *Sol. Energy* 224 (2021) 672–684.
- [6] M. Pierro, M. De Felice, E. Maggioni, D. Moser, A. Perotto, F. Spada, C. Cornaro, Photovoltaic generation forecast for power transmission scheduling: A real case study, *Sol. Energy* 174 (2018) 976–990.
- [7] E. Lorenz, T. Scheidsteiger, J. Hurka, D. Heinemann, C. Kurz, Regional PV power prediction for improved grid integration, *Prog. Photovolt., Res. Appl.* 19 (7) (2011) 757–771.
- [8] J. Antonanzas, N. Osorio, R. Escobar, R. Urraca, F.J. Martinez-de Pison, F. Antonanzas-Torres, Review of photovoltaic power forecasting, *Solar Energy* 136 (2016) 78–111.
- [9] M.J. Mayer, Influence of design data availability on the accuracy of physical photovoltaic power forecasts, *Sol. Energy* 227 (2021) 532–540.
- [10] R.A. Rajagukguk, R.A. Ramadhan, H.-J. Lee, A review on deep learning models for forecasting time series data of solar irradiance and photovoltaic power, *Energies* 13 (24) (2020) 6623.
- [11] M. Abdel-Basset, H. Hawash, R.K. Chakraborty, M. Ryan, PV-Net: An innovative deep learning approach for efficient forecasting of short-term photovoltaic energy production, *J. Clean. Prod.* 303 (2021) 127037.
- [12] P. Li, K. Zhou, X. Lu, S. Yang, A hybrid deep learning model for short-term PV power forecasting, *Appl. Energy* 259 (2020) 114216.
- [13] Y. Tang, K. Yang, S. Zhang, Z. Zhang, Photovoltaic power forecasting: A hybrid deep learning model incorporating transfer learning strategy, *Renew. Sustain. Energy Rev.* 162 (2022) 112473.
- [14] M. Massaoudi, I. Chih, H. Abu-Rub, S.S. Refaat, F.S. Oueslati, Convergence of photovoltaic power forecasting and deep learning: State-of-art review, *IEEE Access* 9 (2021) 136593–136615.
- [15] E. Sarma, N. Dimitropoulos, V. Marinakis, Z. Mylonas, H. Doukas, Transfer learning strategies for solar power forecasting under data scarcity, *Sci. Rep.* 12 (1) (2022) 14643.
- [16] M. Abubakar, B. Akoush, A. Khalil, M.A. Hassan, Unleashing deep neural network full potential for solar radiation forecasting in a new geographic location with historical data scarcity: a transfer learning approach, *Eur. Phys. J. Plus* 137 (4) (2022) 474.
- [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] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. March, V. Lempitsky, Domain-adversarial training of neural networks, *J. Mach. Learn. Res.* 17 (59) (2016) 1–35, URL <http://jmlr.org/papers/v17/15-239.html>.
- [19] Y. Zhang, T. Liu, M. Long, M. Jordan, Bridging theory and algorithm for domain adaptation, in: K. Chaudhuri, R. Salakhutdinov (Eds.), *Proceedings of the 36th International Conference on Machine Learning*, in: *Proceedings of Machine Learning Research*, vol. 97, PMLR, 2019, pp. 7404–7413, URL <https://proceedings.mlr.press/v97/zhang19i.html>.
- [20] E. Sarma, E. Spiliotis, E. Stamatopoulos, V. Marinakis, H. Doukas, Short-term photovoltaic power forecasting using meta-learning and numerical weather prediction independent long short-term memory models, *Renew. Energy* 216 (2023) 118997.
- [21] D. Markovics, M.J. Mayer, Comparison of machine learning methods for photovoltaic power forecasting based on numerical weather prediction, *Renew. Sustain. Energy Rev.* 161 (2022) 112364, <http://dx.doi.org/10.1016/j.rser.2022.112364>, URL <https://www.sciencedirect.com/science/article/pii/S136403212200274X>.
- [22] S. Aslam, H. Herodotou, S.M. Mohsin, N. Javaid, N. Ashraf, S. Aslam, A survey on deep learning methods for power load and renewable energy forecasting in smart microgrids, *Renew. Sustain. Energy Rev.* 144 (2021) 110992.
- [23] D. Korkmaz, H. Acikgoz, C. Yildiz, A novel short-term photovoltaic power forecasting approach based on deep convolutional neural network, *Int. J. Green Energy* 18 (5) (2021) 525–539.
- [24] A. Agga, A. Abbou, M. Labbadi, Y. El Houm, I.H.O. Ali, CNN-LSTM: An efficient hybrid deep learning architecture for predicting short-term photovoltaic power production, *Electr. Power Syst. Res.* 208 (2022) 107908.
- [25] L. Wen, K. Zhou, S. Yang, X. Lu, Optimal load dispatch of community microgrid with deep learning based solar power and load forecasting, *Energy* 171 (2019) 1053–1065.
- [26] M.N. Akhter, S. Mekhilef, H. Mokhlis, Z.M. Almohaimeed, M.A. Muhammad, A.S.M. Khairuddin, R. Akram, M.M. Hussain, An hour-ahead PV power forecasting method based on an RNN-LSTM model for three different PV plants, *Energies* 15 (6) (2022) <http://dx.doi.org/10.3390/en15062243>, URL <https://www.mdpi.com/1996-1073/15/6/2243>.
- [27] M. Elsaraiti, A. Merabet, Solar power forecasting using deep learning techniques, *IEEE Access* 10 (2022) 31692–31698, <http://dx.doi.org/10.1109/ACCESS.2022.3160484>.
- [28] W. Lee, K. Kim, J. Park, J. Kim, Y. Kim, Forecasting solar power using long-short term memory and convolutional neural networks, *IEEE Access* 6 (2018) 73068–73080.
- [29] K. Wang, X. Qi, H. Liu, A comparison of day-ahead photovoltaic power forecasting models based on deep learning neural network, *Appl. Energy* 251 (2019) 113315.
- [30] Y. Tang, K. Yang, S. Zhang, Z. Zhang, Photovoltaic power forecasting: A hybrid deep learning model incorporating transfer learning strategy, *Renew. Sustain. Energy Rev.* 162 (2022) 112473, <http://dx.doi.org/10.1016/j.rser.2022.112473>, URL <https://www.sciencedirect.com/science/article/pii/S1364032122003781>.
- [31] H. Zang, L. Cheng, T. Ding, K.W. Cheung, Z. Wei, G. Sun, Day-ahead photovoltaic power forecasting approach based on deep convolutional neural networks and meta learning, *Int. J. Electr. Power Energy Syst.* 118 (2020) 105790.
- [32] R. Zhu, W. Guo, X. Gong, Short-term photovoltaic power output prediction based on k-fold cross-validation and an ensemble model, *Energies* 12 (7) (2019) 1220.
- [33] A. Nespoli, S. Leva, M. Mussetta, E.G.C. Ogliari, A selective ensemble approach for accuracy improvement and computational load reduction in ANN-based PV power forecasting, *IEEE Access* 10 (2022) 32900–32911, <http://dx.doi.org/10.1109/ACCESS.2022.3158364>.
- [34] L. Liu, M. Zhan, Y. Bai, A recursive ensemble model for forecasting the power output of photovoltaic systems, *Sol. Energy* 189 (2019) 291–298.
- [35] A.A. Lateko, H.-T. Yang, C.-M. Huang, H. Aprillia, C.-Y. Hsu, J.-L. Zhong, N.H. Phuong, Stacking ensemble method with the RNN meta-learner for short-term pv power forecasting, *Energies* 14 (16) (2021) 4733.
- [36] H. Eom, Y. Son, S. Choi, Feature-selective ensemble learning-based long-term regional pv generation forecasting, *IEEE Access* 8 (2020) 54620–54630.
- [37] E. Sarma, E. Spiliotis, V. Marinakis, G. Tzanes, J.K. Kaldellis, H. Doukas, ML-based energy management of water pumping systems for the application of peak shaving in small-scale islands, *Sustainable Cities Soc.* 82 (2022) 103873.
- [38] A. de Mathelin, F. Deheeger, M. Mougeot, N. Vayatis, From theoretical to practical transfer learning: The ADAPT library, in: *Federated and Transfer Learning*, Springer, 2022, pp. 283–306.
- [39] E. Tzeng, J. Hoffman, K. Saenko, T. Darrell, Adversarial discriminative domain adaptation, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 2962–2971, <http://dx.doi.org/10.1109/CVPR.2017.316>.
- [40] J. Shen, Y. Qu, W. Zhang, Y. Yu, Wasserstein distance guided representation learning for domain adaptation, in: *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, in: AAAI'18/IAAI'18/EAAI'18, AAAI Press, 2018.
- [41] M. Long, Z. Cao, J. Wang, M.I. Jordan, Conditional adversarial domain adaptation, in: S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, R. Garnett (Eds.), *Advances in Neural Information Processing Systems*, vol. 31, Curran Associates, Inc., 2018, URL <https://proceedings.neurips.cc/paper/2018/file/ab88b15733f543179858600245108dd8-Paper.pdf>.
- [42] A. Gretton, D. Sejdinovic, H. Strathmann, S. Balakrishnan, M. Pontil, K. Fukumizu, B.K. Sriperumbudur, Optimal kernel choice for large-scale two-sample tests, in: F. Pereira, C. Burges, L. Bottou, K. Weinberger (Eds.), *Advances in Neural Information Processing Systems*, vol. 25, Curran Associates, Inc., 2012, URL <https://proceedings.neurips.cc/paper/2012/file/dbe272bab69f8e13f14b405e038deb64-Paper.pdf>.
- [43] S. Uguroglu, J. Carbonell, Feature selection for transfer learning, in: D. Gunopulos, T. Hofmann, D. Malerba, M. Vazirgiannis (Eds.), *Machine Learning and Knowledge Discovery in Databases*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2011, pp. 430–442.
- [44] B. Sun, J. Feng, K. Saenko, Return of frustratingly easy domain adaptation, in: *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, AAAI '16, AAAI Press, 2016, pp. 2058–2065.
- [45] B. Sun, K. Saenko, Deep CORAL: Correlation alignment for deep domain adaptation, in: G. Hua, H. Jégou (Eds.), *Computer Vision – ECCV 2016 Workshops*, Springer International Publishing, Cham, 2016, pp. 443–450.
- [46] S. Hochreiter, J. Schmidhuber, Long Short-Term Memory, *Neural Comput.* 9 (8) (1997) 1735–1780, <http://dx.doi.org/10.1162/neco.1997.9.8.1735>, arXiv:https://direct.mit.edu/neco/article-pdf/9/8/1735/813796/neco.1997.9.8.1735.pdf.
- [47] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, in: *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS '17, Curran Associates Inc., Red Hook, NY, USA, 2017, pp. 6000–6010.
- [48] C. Chen, D. Han, C.-C. Chang, CAAN: Context-aware attention network for visual question answering, *Pattern Recognit.* 132 (2022) 108980, <http://dx.doi.org/10.1016/j.patcog.2022.108980>, URL <https://www.sciencedirect.com/science/article/pii/S0031320322004605>.
- [49] J.L. Ba, J.R. Kiros, G.E. Hinton, Layer normalization, 2016, arXiv preprint arXiv:1607.06450.
- [50] Y. Ganin, V. Lempitsky, Unsupervised domain adaptation by backpropagation, in: F. Bach, D. Blei (Eds.), *Proceedings of the 32nd International Conference on Machine Learning*, in: *Proceedings of Machine Learning Research*, vol. 37, PMLR, Lille, France, 2015, pp. 1180–1189, URL <https://proceedings.mlr.press/v37/ganin15.html>.
- [51] D. Yang, A guideline to solar forecasting research practice: Reproducible, operational, probabilistic or physically-based, ensemble, and skill (ROPES), *J. Renew. Sustain. Energy* 11 (2) (2019) 022701.

- [52] D. Yang, Making reference solar forecasts with climatology, persistence, and their optimal convex combination, *Sol. Energy* 193 (2019) 981–985.
- [53] D. Yang, S. Alessandrini, J. Antonanzas, F. Antonanzas-Torres, V. Badescu, H.G. Beyer, R. Blaga, J. Boland, J.M. Bright, C.F. Coimbra, et al., Verification of deterministic solar forecasts, *Sol. Energy* 210 (2020) 20–37.
- [54] H.T. Pedro, C.F. Coimbra, Assessment of forecasting techniques for solar power production with no exogenous inputs, *Sol. Energy* 86 (7) (2012) 2017–2028.
- [55] A. de Mathelin, F. Deheeger, G. Richard, M. Mougeot, N. Vayatis, Adapt: Awesome domain adaptation python toolbox, 2021, arXiv preprint [arXiv:2107.03049](https://arxiv.org/abs/2107.03049).
- [56] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G.S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, X. Zheng, TensorFlow: Large-scale machine learning on heterogeneous systems, 2015, URL <https://www.tensorflow.org/>. Software available from tensorflow.org.