

ORIGINAL RESEARCH

Open access

Choosing, tailoring, and integrating models for transfer learning in forecasting wind and solar power

Thomas Andersen^{1*}, Lars Nielsen¹, Mette Sørensen²

Abstract

Recent advancements have sparked growing interest in exploiting collections of pre-trained models, known as model hubs, particularly within computer vision applications. In practice, this involves first identifying a suitable pre-trained source model and then tailoring it to the new target domain to bridge any mismatches. Despite this trend, investigations into optimal model choice and tailoring methods for renewable energy generation forecasts remain scarce. Specifically, prior research has not delved into varied selection or adaptation techniques tailored to different neural network structures. Furthermore, no existing analyses have systematically evaluated how the volume of available training data affects outcomes, nor have they incorporated seasonal variations into performance assessments. We bridge these research gaps through an initial in-depth study of model selection and adaptation within transfer learning frameworks for renewable power prediction. Inspired by cutting-edge advances in computer vision, the analysis draws on operational data from 667 wind farms and solar (photovoltaic) installations across six datasets. To generate robust error metrics, we simulate scenarios with varying volumes of training observations across seasons. Model selection is then appraised using both marginal likelihood estimates and resulting forecast inaccuracies under these constrained data regimes. In addition, four separate adaptation methodologies are rigorously compared. Building on and surpassing conventional approaches, we introduce Bayesian linear regression to model the output variable from deep features derived by a neural network backbone. This technique delivers superior results compared to standard baselines, even with only 7 days of local training data. It also shows that full network fine-tuning provides negligible gains unless at least 3 months of data are available. Beyond that, we demonstrate that aggregating predictions from multiple models into ensembles markedly improves the effectiveness of both selection and adaptation. Consequently, ensemble-based strategies achieve mean error levels with only 30 days of training data that would otherwise require a full year of observations. In the most representative datasets, this yields average errors of 9.8% for photovoltaic power and 14% for wind power, utilizing only seven days of site-specific training data.

Keywords Transfer learning, Time series, Renewable energies, Temporal convolutional neural network, Ensembles, Wind and photovoltaic power

*Correspondence:

Thomas Andersen

thomas.andersen@ku.dk

¹ Department of Smart Power Networks, Faculty of Engineering, University of Copenhagen, Copenhagen, Denmark

² Department of Grid Optimization Systems, Faculty of Engineering, Technical University of Denmark, Lyngby, Denmark

Introduction

The growing integration of variable renewable energy sources, such as wind farms and photovoltaic (PV) installations, creates a key challenge when adding a new park to an operator's existing portfolio. New (target) parks often lack historical data, while accurate power forecasts are essential for maintaining grid stability because they depend heavily on weather conditions. Fortunately, operators often have access to numerous pre-trained models developed for other

existing parks, which can be leveraged to improve forecasting performance and significantly reduce the computational cost of building a new model from scratch [1–3]. The central question is therefore how to best exploit this collection (model hub) of pre-trained models. Research on inductive transfer learning (ITL) offers valuable approaches to address this issue [4].

Figure 1a provides an overview of the strategy we propose. The initial phase involves selecting a suitable source model from the hub. Recent studies [4, 5] have demonstrated that the choice of source model has a substantial impact on the final test error in computer vision tasks. To select an appropriate source model, suppose we have two source tasks (e.g., from different wind parks) with model parameters θ_1 and θ_2 , along with their respective input observations X_1 and X_2 and response values Y_1 and Y_2 . Using this information, the goal is to identify the most suitable model for transfer to a new target task T characterized by parameters θ_T and its corresponding input set X_T and output set Y_T .

Figure 1b illustrates this selection problem in the context of wind power forecasting. In renewable energy forecasting, inputs X to the machine learning models typically consist of weather predictions—such as wind speed or solar radiation—generated by numerical weather prediction (NWP) models. These features are used by models with parameters θ to predict the expected power output $E[p(Y | X, \theta)]$ for a day-ahead horizon, usually 24 to 48 hours ahead. The diagram highlights that the similarity between source and target depends on the relationship between input features (e.g., wind speed) and the resulting power generation, which varies across different wind speeds and turbine characteristics.

After selecting a source model, the second step is to adapt the transferred knowledge to the target park using the limited available data with an appropriate adaptation strategy. A common approach is fine-tuning only the final layer of a neural network. Such adaptation is crucial for generating reliable, site-specific forecasts for a new park despite scarce historical data. To the best of our knowledge, the combination of different model selection and adaptation strategies within inductive transfer learning has not yet been systematically explored for renewable power forecasting. This paper addresses that gap.

Selecting and adapting only a single source model has a clear limitation: it discards potentially useful knowledge from other source models. To overcome this, an optional third step can be introduced that selects multiple source models, adapts them to the target, and combines them using ensemble techniques. The forecasts from these adapted models are merged through a suitable weighting scheme. This ensemble approach enables the integration of insights from multiple parks into the target forecast.

Because inductive transfer learning remains underexplored in the domain of renewable power forecasting [6]—particularly for day-ahead horizons of 24 to 48 hours—we specifically investigate the following research questions:

Research Question 1: What constitutes a suitable similarity measure for selecting an appropriate source model for a new target park from a hub of pre-trained models?

Research Question 2: What is the most effective adaptation strategy once a source model has been selected?

Research Question 3: Do ensemble strategies provide advantages over the selection and adaptation of a single source model when combining knowledge from multiple sources?

Each research question corresponds directly to one of the core steps in applying transfer learning to renewable power forecasting. Our contributions consist of developing concrete methods for each step, with each building on the previous. In a preparatory step (step 0), we construct a model hub by training three model types—Bayesian extreme learning machine (BELM), multi-layer perceptron (MLP), and temporal convolutional network (TCN)—on six datasets spanning 667 distinct parks. For source model selection (step 1), we propose using either the marginal likelihood (also called the evidence) or the normalized root-mean-squared error (nRMSE). Once a model is selected, we adapt it to the target task (step 2) by evaluating four different adaptation strategies, including fine-tuning via weight decay. We demonstrate the

transferability of all three model types from source to target and benchmark them against a gradient boosting regression tree (GBRT) baseline. Notably, the GBRT outperforms conventional physical models [7], which are frequently used as a fallback when data is limited. In an optional third step, we investigate ensemble methods by adapting Bayesian model averaging (BMA) and the cooperative soft gating ensemble (CSGE) to the inductive transfer learning setting, and compare their performance with the best single-adapted model from the previous experiment.

Based on these datasets and models, the main contributions of this work are:

By addressing the first two research questions, we deliver practical methods and strategies for model selection and adaptation applicable to a broad range of renewable power forecasting problems where they have not previously been considered.

We demonstrate, contrary to widespread assumptions, that fine-tuning only the final layer of a neural network with weight decay can be among the least effective approaches for inductive transfer learning in renewable power forecasting when target data are scarce.

We further show that ensembles based on BMA and CSGE outperform single adapted models. These ensembles achieve a mean forecast error with only 30 days of target training data, whereas training from scratch would require a full year of data.

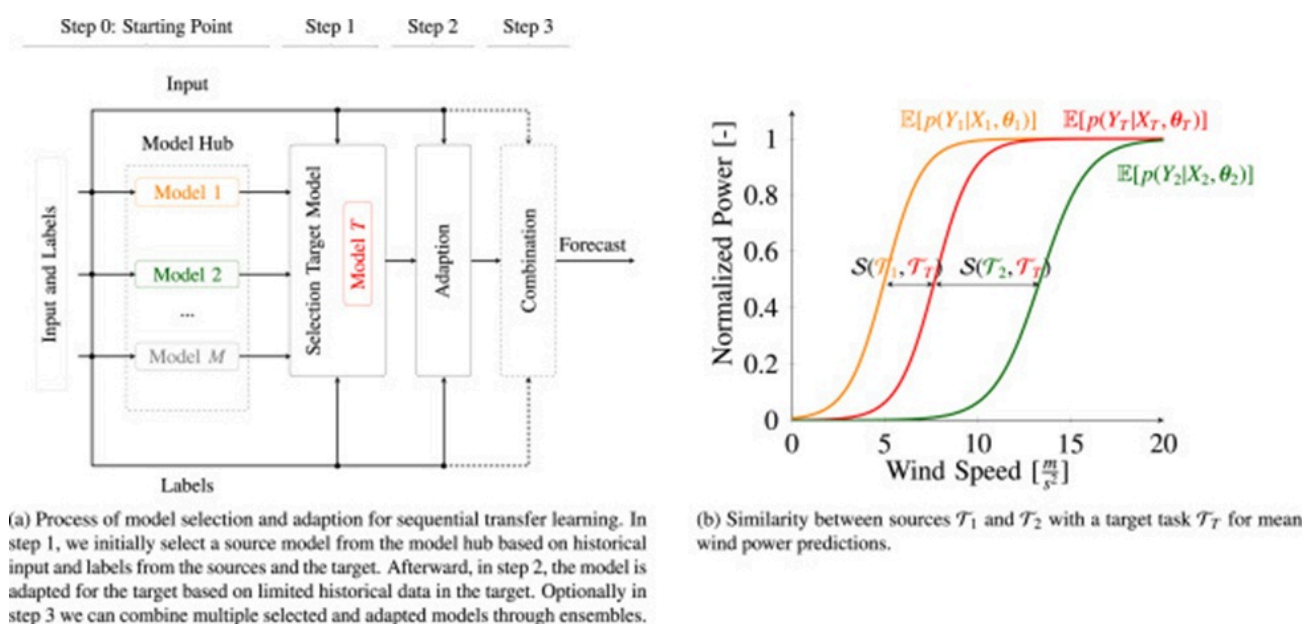


Figure 1. Overview diagrams of the knowledge transfer process using a collection of pre-trained models.

Related Work

The following part surveys recent progress in transfer learning (TL), emphasizing inductive transfer learning (ITL) methods originally developed for computer vision applications that have not yet been examined in the context of renewable power forecasting. This overview helps identify useful techniques worth adapting for renewable energy scenarios. We then provide a summary of existing studies on ITL applied to deterministic renewable power prediction tasks. Additional Bayesian-oriented research in other renewable energy fields can be found in Refs. [8-10].

Two key elements define successful ITL: choosing the right source model and applying an effective adaptation technique. Research presented in [4, 5] explores various selection approaches within computer vision. Their methodology involves

replacing the final layer of each source network with a Bayesian linear regression (BLR) component, which is then fitted to the target dataset via empirical Bayes, also known as evidence approximation. This procedure is applied to all models in the collection. Task similarity is subsequently assessed based on the evidence value each source model achieves on the target data. For adaptation, they propose a Bayesian tuning procedure that leverages predictions from multiple sources to constrain fine-tuning. Such selection and adaptation concepts require careful testing and extension when moving to renewable energy domains. One possibility is to produce forecasts directly from the BLR and benchmark them against the common practice of adjusting only the final layer. Fine-tuning usually applies weight decay to push parameters toward zero [11]. Yet this form of regularization overlooks the parameters originally learned on the source task. By comparison, the method introduced in Ref. [12] adds a penalty term that discourages large shifts away from the source parameters within the weight-decay framework.

Compared with domains such as computer vision, far fewer studies address ITL for renewable power forecasting [6]. A subset of earlier efforts concentrates on building transferable feature representations through autoencoders [13–16]. Although an autoencoder transfer step could be integrated into the framework discussed here, capturing the conditional distribution of power output is more important for assessing model suitability and for later combining outputs. The purely data-driven TL techniques covered in Refs. [17, 18] lie beyond the focus of the present study.

A large portion of TL literature in renewable power forecasting uses raw meteorological observations as inputs [14–25]. These contributions generally address short prediction windows ranging from 10 seconds to 2 hours. Day-ahead horizons, however, present greater difficulty, as they depend on outputs from numerical weather prediction (NWP) systems, and prediction inaccuracies typically increase as the forecast lead time lengthens [26].

Nearly all of the referenced TL studies frame power forecasting as a plain regression task. Nevertheless, recurring patterns such as daily cycles are widely recognized yet frequently ignored by standard regression formulations. To account for such temporal structures, Ref. [3] incorporates time-series models within a multi-task learning (MTL) framework. Likewise, the authors in Ref. [27] combine recurrent architectures with fine-tuning and report favorable outcomes for ultra-short-term photovoltaic (PV) predictions.

Reference [28] reports gains in day-ahead PV forecasting by training models that simultaneously target multiple outputs. Their model-combination concept resembles the ensemble idea explored later in this work, although it is not assessed under an inductive transfer learning lens. In Ref. [29], an MTL scheme based on Gaussian processes is proposed for PV targets. Another contribution [30] groups wind parks via clustering and applies a weighting mechanism to generate outputs for unseen parks; unfortunately, that analysis relies exclusively on simulated data instead of real historical power records.

A series of papers adopts MTL structures for day-ahead forecasting horizons [3, 31, 32]. The task-embedding strategy is detailed in Refs. [3] and [31] for multi-layer perceptrons (MLPs) and convolutional neural networks (CNNs) embed task-specific details to uncover hidden similarities across different tasks. The setup in Ref. [3] is particularly relevant because it mirrors key aspects of our experimental design. Still, assessing performance on a seasonal breakdown can produce misleading conclusions for ITL methods, which must guard against catastrophic forgetting throughout every season. MTL designs see limited uptake in practical industry settings, largely because of the extra effort needed for data preparation and model training. It is therefore vital to extract maximum benefit from already-trained single-task models when scaling renewable capacity through ITL.

In addition, prior literature has not systematically compared alternative model selection or adaptation procedures across different neural network families. Although a few studies have investigated ensemble-style merging of information from several locations for PV, those efforts fall short: they often disregard the actual quantity of available data, base expected power solely on static characteristic curves, or restrict analysis to either PV or wind exclusively. The present work fills these gaps through a comprehensive evaluation that addresses the noted shortcomings in day-ahead forecasting applications.

Proposed Methods

The sections ahead outline the suggested procedures for selecting models, adapting them, and merging their outputs. As preparation, we briefly describe the model types in the collection and introduce Bayesian linear regression (BLR), given its pivotal role in both selection and adaptation approaches.

Step 0: Source models for the model hub

To achieve dependable forecasts despite limited target data via transfer learning, the study employs three machine learning architectures: a Bayesian extreme learning machine (BELM), a multi-layer perceptron (MLP), and a temporal convolutional network (TCN).

BELM builds on linear regression by mapping inputs into a higher-dimensional feature space, where linear separation is often easier, thereby supporting improved predictions [33]. NWP-derived features are mapped using randomly initialized weights and a nonlinear transformation, such as the rectified linear unit (ReLU). The resulting expanded features then combine linearly to estimate the target value. Here, the linear step uses Bayesian linear regression (BLR). In contrast to ordinary linear regression, BLR supplies the marginal likelihood, offering a direct indicator of how well a source model aligns with target data—making it useful for ITL selection. BELM's quick training and convex nature explain its popularity in renewable power forecasting applications [19, 34].

MLPs, along with deeper neural networks in general, serve as established tools for regression and classification and are optimized via gradient-based methods. Inside an MLP, inputs pass through successive matrix multiplications followed by nonlinear activations (commonly ReLU). These operations form layers, with each layer's output feeding the next. Stacking layers in this manner allows the network to discover effective data representations. Renewable power estimation typically employs a straightforward linear summation in the output layer. MLPs excel at extracting representations from NWP inputs that transfer well, which explains their frequent use in renewable forecasting and TL contexts [35, 36].

TCNs extend the MLP concept to explicitly handle temporal relations in sequential forecasting problems [37-39]. They rely on one-dimensional convolutional layers that slide along the time axis for each input channel (where a channel typically represents a single feature). Although two-dimensional CNNs have dominated computer vision and associated TL work [40, 41], one-dimensional variants have recently attracted interest for time-series TL [42]. CNNs suit transfer learning particularly well because their layered hierarchy allows focused adjustment of only the later stages. In a TCN, standard convolutional layers are replaced by residual blocks—an idea popularized in vision tasks [43]. Each residual block includes a shortcut link that bypasses the internal transformations and adds the incoming activation to the block output, helping prevent vanishing gradients. Processing within a block proceeds in the following sequence: dilated convolution, weight normalization, ReLU, and dropout. Dilated convolutions enlarge the effective observation window without excessive computational or memory overhead. When the input and output sizes mismatch, an additional convolution aligns the dimensions within the shortcut path.

Bayesian linear regression

Bayesian linear regression (BLR) occupies a central position in this study, serving the needs for:

- Constructing the BELM,
- Quantifying similarity via marginal likelihood,
- Modifying neural networks by substituting their output layer with a BLR,
- Fusing predictions through Bayesian model averaging (BMA).

Because of this importance, a detailed definition follows, drawing on foundational presentations in Refs. [44, 45]. A Bayesian viewpoint on linear regression provides richer information through the posterior, proving especially helpful in data-scarce settings [44]—a typical situation in transfer learning. This perspective aids similarity evaluation for selection and supports uncertainty-aware assessments, in line with suggestions from [4]. We advocate applying BLR for adaptation in renewable forecasting by exchanging the last network layer, retraining on target samples, and then generating forecasts. The same formulation also enables model fusion via BMA and underpins BELM training.

Consider the posterior of the linear model expressed in the following general form (using the notation from the original derivations):

$$\underbrace{p(\boldsymbol{\theta}|X, Y)}_{\text{posterior}} = \frac{\overbrace{p(Y|X, \boldsymbol{\theta})p(\boldsymbol{\theta})}^{\text{likelihood prior}}}{\underbrace{p(Y|X)}_{\text{marginal likelihood}}}, \quad (1)$$

Where X and Y represent the collections of observed inputs and corresponding responses containing N ($N \geq 1$) training samples. Each feature vector x_n belongs to $\mathbb{R}^D$ with D ($D \geq 1$) dimensions, and each y_n is a scalar in $\mathbb{R}$. The likelihood term $p(Y|X, \boldsymbol{\theta})$ reflects how well the inputs X together with weights $\boldsymbol{\theta} \in \mathbb{R}^D$ account for the observed targets. The prior distribution captures initial assumptions regarding plausible weight values. The marginal likelihood serves as the normalizing constant, and the posterior—conditioned on the training data—encodes updated beliefs about the parameters of the target task.

Deriving the posterior starts with a Gaussian prior $p(\boldsymbol{\theta}|\alpha) = \mathcal{N}(\boldsymbol{\theta}|\mathbf{0}, \alpha^{-1}\mathbf{I})$, where $\alpha > 0$ controls the precision of this zero-mean isotropic Gaussian. The isotropic choice yields a closed-form solution, reducing the effort required to compute the mean vector and covariance matrix.

Let $\mathbf{y}$ denote the $1 \times N$ vector of targets and X the $N \times D$ design matrix whose rows hold the individual observations. For problems with multiple outputs, independent models can be learned per dimension. In the scenarios considered here, X contains either raw NWP variables (e.g., wind speed, radiation), randomly projected features from a BELM, or activations from the penultimate layer of a neural network. In the last two situations, the original weather inputs have already been transformed by the BELM or network layers.

Conditioned on X , $\mathbf{y}$, and the noise precision $\beta > 0$, the posterior takes the Gaussian form $p(\boldsymbol{\theta}|X, \mathbf{y}) = \mathcal{N}(\boldsymbol{\theta}|\mathbf{m}_N, \mathbf{S}_N)$, with parameters:

$$\mathbf{m}_N = \beta \mathbf{S}_N \mathbf{X}^T \mathbf{y} \text{ and } \mathbf{S}_N^{-1} = \alpha \mathbf{I} + \beta \mathbf{X}^T \mathbf{X}. \quad (2)$$

Where the explicit expressions for $\mathbf{m}_N$ and $\mathbf{S}_N$ follow the standard Bayesian update rules for linear models.

N here denotes the number of target samples used to refine the prior. When predicting for an unseen input X^* drawn from a test set, the predictive posterior distribution becomes:

$$\begin{aligned} p(\mathbf{X}_*|\mathbf{y}, \alpha, \beta) &= \int p(\mathbf{X}_*|\mathbf{y}, \boldsymbol{\theta}) p(\boldsymbol{\theta}|X, \mathbf{y}) d\boldsymbol{\theta} \\ &= \mathcal{N}(\mathbf{y}_*|\mathbf{X}_* \mathbf{m}_N, \sigma_N^2(\mathbf{X}_*)), \end{aligned} \quad (3)$$

With the associated predictive variance given by:

$$\sigma_N^2(\mathbf{X}_*) = \beta^{-1} + \mathbf{X}_*^T \mathbf{S}_N \mathbf{X}_*. \quad (4)$$

Step 1: Model selection for inductive transfer learning

Determining how closely a target task matches available source tasks constitutes a core difficulty in ITL [4]. A well-chosen source should deliver lower error than learning the target model independently, thereby preventing harmful negative transfer.

Before formally specifying selection within TL, it helps to recall the broader rationale for (Bayesian) model selection. Reference [44] notes that any sound selection procedure must weigh a model's expressive capacity against its agreement with the observed data. Figure 2 visualizes this balance from a Bayesian standpoint. The x-axis spans the universe of conceivable datasets, while the y-axis plots the evidence a model assigns to a particular dataset. Suppose model M_2 possesses more parameters and can therefore accommodate a wider variety of datasets than the simpler model M_1 . When evaluating evidence $p(D | M_m)$, the Bayesian criterion tends to prefer the less complex M_1 for the given dataset D . This built-in preference for parsimony in Bayesian model evaluation aligns with the principle known as Occam's razor.

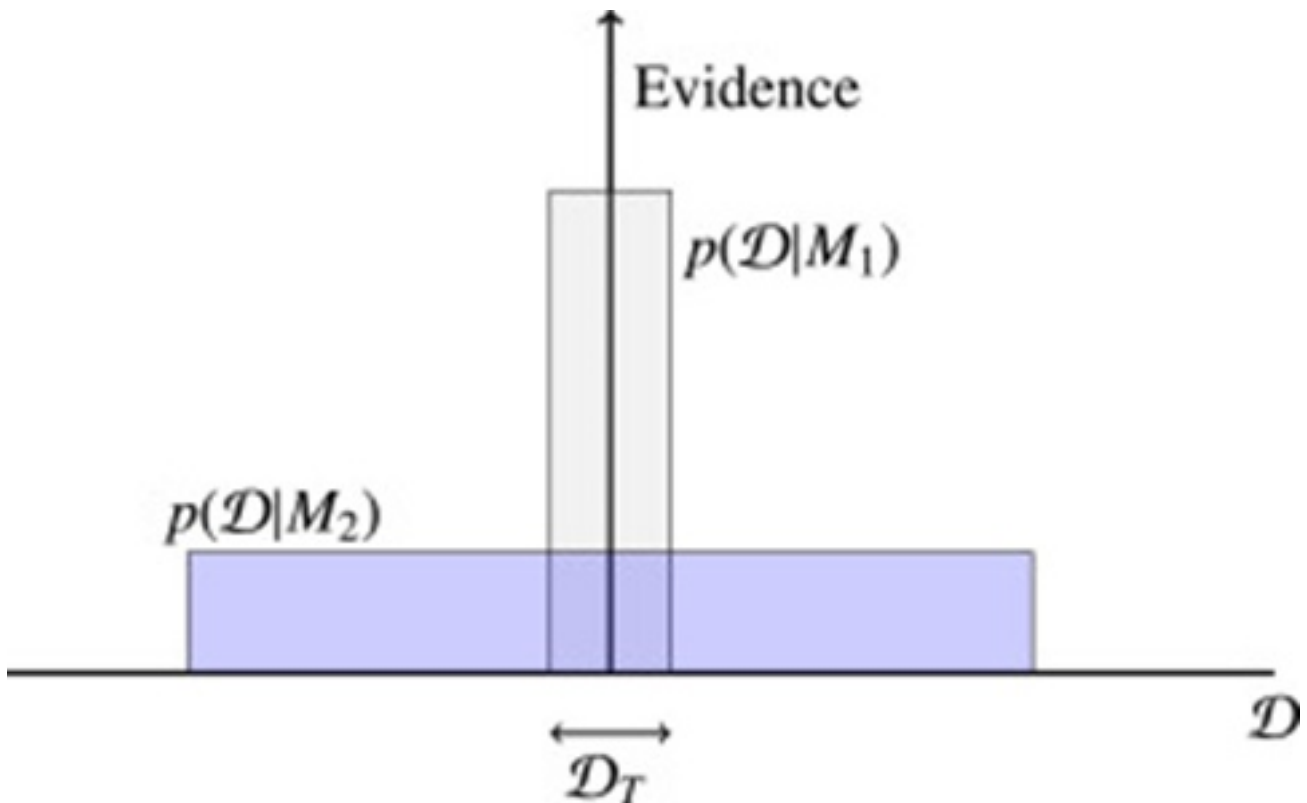


Figure 2. Bayesian model selection. Adapted from [45].

The same model-selection principle carries over directly to transfer learning settings. The objective is to identify, among M ($M > 1$) available source models, the one that best accounts for the scarce target data T . Consider two example tasks defined as $T_1 = \{Y, P_1(Y_1 | X_1)\}$ and $T_2 = \{Y, P_2(Y_2 | X_2)\}$, where both belong to the universal set of tasks T . Here, the sets Y_m and X_m represent observations drawn from the output space Y and input feature space X , respectively. A similarity function $S: T \times T \rightarrow \mathbb{R} \geq 0$ maps pairs of tasks to a non-negative scalar, enabling quantitative comparisons. For

instance, if $S(T1, TT) > S(T2, TT)$ for a target task TT, then task T1 shares greater similarity with TT than T2 does; higher values of S therefore indicate stronger alignment. Dissimilarity can be expressed simply as the reciprocal of such a similarity measure.

This raises a practical question: which similarity metric should be used, and which data are most relevant when selecting a source model from the collection for a given target? One straightforward option would be to evaluate similarity solely in the input feature space. However, input features alone provide incomplete insight into the expected power output (the response variable) in renewable energy systems. For example, identical solar radiation levels can yield substantially different energy production depending on the specific PV module characteristics. It is therefore necessary to incorporate information from the response variable as well.

Evidence or marginal likelihood

The work in Ref. [4] employs the marginal likelihood, also known as model evidence, as the similarity measure S. To compute it, the final layer of a source neural network T_m is substituted with a Bayesian linear regression (BLR) component. The prior parameters α and β of this BLR are then optimized using empirical Bayes [5, 44] on the limited target data. In this setup, the source network functions purely as a feature extractor. The marginal likelihood takes the following form:

$$\mathcal{S}(\mathcal{T}_T, \mathcal{T}_m) = \ln p(\mathbf{y} | \alpha, \beta) = \frac{D}{2} \ln \alpha + \frac{N}{2} \ln \beta - E(\mathbf{m}_N) - \frac{1}{2} \ln |\mathbf{S}_N^{-1}| - \frac{N}{2} \ln 2\pi, \quad (5)$$

Where D denotes the feature dimensionality (for example, the size of the penultimate layer of network m), N is the number of target samples, and the term $E(\mathbf{m}_N)$ is defined as $\beta/2 \cdot \|\mathbf{y} - \mathbf{X}\mathbf{m}_N\|^2 + \alpha/2 \cdot \mathbf{0}\mathbf{m}_N^T\mathbf{0}\mathbf{m}_N$ [5, 44]. This calculation jointly considers features produced by the source network for task T_m and the actual target responses from TT. Repeating the process for every source model m yields the marginal likelihood of each on the target data, which is then used to define $S(T_m, TT)$. The source model with the strongest evidence is chosen as the most suitable. For problems with multiple output dimensions, the procedure is performed separately for each dimension, and the resulting values are averaged [4].

Although this technique is elegant and broadly applicable, it has one notable limitation in inductive transfer learning for renewable power forecasting: it discards the weights of the original final layer that have already been trained. In many renewable energy applications, the output layer dimensions are compatible between source and target, so replacing it entirely is unnecessary. Moreover, a pre-trained output layer typically provides a better starting point than a randomly initialized one.

Normalized root mean-squared error

As an alternative, we suggest assessing similarity directly using the normalized root-mean-squared error (nRMSE) produced by the unaltered pre-trained source model on the target data, as expressed in Eqs. 6 and 7.

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N \left(y_i^{(T)} - \hat{y}_i^{(m)} \right)^2} \quad (6)$$

$$\mathcal{S}(\mathcal{T}_T, \mathcal{T}_m)^{-1} = \text{nRMSE} = \frac{\text{RMSE} - y_{\min}}{y_{\max} - y_{\min}} \quad (7)$$

In these expressions, $y_i(T)$ stands for the i -th observed response value from the target, $\hat{y}_i(m)$ represents the prediction generated by source model m on the target inputs, N ($N \geq 1$) is the number of samples, and $y_{\max}$ and $y_{\min}$ are the maximum and minimum values of the response variable, respectively. Because a smaller nRMSE indicates better alignment, the similarity measure is taken as the inverse, so that larger values correspond to better performance. For consistent normalization across all datasets, $y_{\max}$ is set to the park's nominal power capacity, and $y_{\min}$ is set to zero. This approach allows a direct evaluation of how effectively a source model performs on the available target samples. The source model with the lowest nRMSE on a validation subset of the target data is therefore selected.

Step 2: Adaptation strategies for inductive transfer learning

Table 1 presents an overview of the 18 different combinations of models, selection methods, and adaptation techniques examined in this study. For the Bayesian extreme learning machine (BELM), a straightforward transfer learning baseline is obtained by updating its posterior online: the posterior learned on the source task serves as the new prior when incorporating target data. We also test the naïve strategy of deploying the selected source model to the target data with no parameter updates.

Two widely used fine-tuning approaches borrowed from computer vision are also included. The first applies conventional weight decay, which penalizes weights moving away from zero. The second variant, termed weight decay with respect to source parameters, penalizes deviations from the source weights instead. In addition, we evaluate the Bayesian tuning method proposed in Ref. [5].

The final three adaptation approaches serve as regularization techniques. Broadly speaking, this entails introducing an extra penalty component L_{pen} into the primary task loss L_{task} according to the following expression:

$$L = L_{\text{task}} + \lambda \cdot L_{\text{pen}}, \quad (8)$$

Where $\lambda \in \mathbb{R}$ acts as a tunable hyperparameter for the regularization strength, which is normally determined via hyperparameter tuning. The task-specific loss L_{task} is formulated as:

$$L_{\text{task}} = \frac{1}{N} \sum_{n=1}^N l(f(\mathbf{x}_n, \boldsymbol{\theta}), y_n), \quad (9)$$

With $\boldsymbol{\theta} \in \mathbb{R}^p$ ($p \in \mathbb{N}$, $p \geq 1$) being the parameter vector that undergoes optimization, $\mathbf{x}_n$ standing for the n th input sample ($n \in \mathbb{N}$ and total samples $N \in \mathbb{N}$, $N \geq 1$), and y_n denoting the associated target value. A single-output response is considered here for ease of explanation.

In the case of weight decay (WD) anchored at the origin [11], the penalty term L_{pen} is defined by:

$$L_{\text{WD}} = \frac{1}{2} \|\boldsymbol{\theta}\|_2^2 \quad (10)$$

When the goal is to discourage large shifts away from the source model, [12] suggests weight decay relative to the source weights (WDS), expressed as

$$L_{\text{WDS}} = \frac{1}{2} \|\boldsymbol{\theta} - \boldsymbol{\theta}^0\|_2^2, \quad (11)$$

Where $\boldsymbol{\theta}^0 \in \mathbb{R}^p$ corresponds to the parameter vector taken from the source model before the fine-tuning stage.

Finally, under Bayesian tuning, the penalty term L_{pen} takes the following form [5]:

$$L_{Bayesian} = \frac{1}{N} \sum_{n=1}^N \frac{1}{K} \sum_{k=1}^K \left(\frac{1}{M} \sum_{m=1}^M \mathbf{x}_{m,n}^T \boldsymbol{\theta}_{m,k} - \mathbf{x}_{t,n}^T \boldsymbol{\theta}_{t,k} \right)^2, \quad (12)$$

In which $n \in \mathbb{N}$ indexes the n th training sample, m refers to the m th source model that has been adapted using BLR, k indicates the k th output dimension (such as distinct forecast horizons), $\mathbf{x}_{m,k}$ represents features pulled from the m th source model, and $\mathbf{x}_{t,n}$ denotes the matching features extracted from the target model t . The vectors $\boldsymbol{\theta}_{k,c}$ and $\boldsymbol{\theta}_{t,c}$ are the mean parameter vectors computed via BLR.

Table 1. Summary of various strategies for model combination, selection, and adaptation. Here, RM denotes the RMSE-based selection approach, EV represents selection guided by evidence, DI indicates direct model application, WD refers to fine-tuning via weight decay relative to the original model, WDS corresponds to fine-tuning via weight decay based on source parameters, and BT signifies fine-tuning using Bayesian optimization.

Model	Selection	Adaptation	Abbreviation
Type	Strategy	Strategy	Empty Cell
MLP/TCN	RMSE [ours]	Direct [ours]	MLP-/TCN-RM-DI
MLP/TCN	RMSE [ours]	Weight decay [11]	MLP-/TCN-RM-WD
MLP/TCN	RMSE [ours]	Weight decay source [12]	MLP-/TCN-RM-WDS
MLP/TCN	EVIDENCE [5]	Direct [ours]	MLP-/TCN-EV-DI
MLP/TCN	EVIDENCE [5]	Direct linear [ours]	MLP-/TCN-EV-DILI
MLP/TCN	EVIDENCE [5]	Weight decay [11]	MLP-/TCN-EV-WD
MLP/TCN	EVIDENCE [5]	Weight decay source [12]	MLP-/TCN-EV-WDS
MLP/TCN	EVIDENCE [5]	Bayesian tuning [5]	MLP-/TCN-EV-BT
BELM	RMSE [ours]	Online [ours]	BELM-RM
BELM	EVIDENCE [5]	Online [ours]	BELM-EV

Step 3: Model combination for inductive transfer learning

We have examined model selection and adaptation methods when transferring from one source model to a target task. Nevertheless, relying on a single model can easily lead to overfitting, especially when the target data is scarce. Merging multiple source models into an ensemble helps mitigate this issue.

Bayesian model averaging

We build on the idea presented in Ref. [4] by moving beyond selecting a single model based on its evidence. Instead, we merge several models adapted via BLR using Bayesian model averaging (BMA). BMA offers a strong theoretical foundation because it accounts for the full predictive posterior [46], thereby incorporating model uncertainty through

$$p(\mathbf{y}_{T*} | \mathbf{y}_T) = \sum_{i=1}^{i=M} p(\mathbf{y}_{T*} | \mathbf{y}_T, \boldsymbol{\theta}_{M_i}) p(\boldsymbol{\theta}_{M_i} | \mathbf{y}_T). \quad (13)$$

The prior probability $p(\theta_{Mi}|T)$ reflects our initial belief regarding the similarity between model M_i and the target dataset. For simplicity, we assume a uniform prior across all source models. Note that the input variable has been left out here to keep the notation concise. The term $p(y_T * |y_T, \theta_{Mi})$ represents the predictive posterior distribution of model M_i , as defined in Eq. 3. This could, for instance, correspond to the outcome of the suggested direct linear adaptation approach.

Coopetitive soft gating ensemble

We further suggest applying the CSGE technique for combining models within the inductive transfer learning framework. Because the CSGE supports both ensemble selection and weighting modes, the term “coopetitive” combines cooperation and competition. The CSGE was originally developed for renewable power forecasting in Ref. [47]. Its core principle is to connect the weights assigned to ensemble members with their respective performance, meaning stronger-performing source models receive higher weights than weaker ones.

The CSGE determines the overall weight of each source model based on three key aspects:

- The global weight reflects a source model's overall performance on the available training data for the target task.
- The local weight captures how well a source model performs across different regions of the feature space on the target task. For example, in wind power, one model may excel at low wind speeds while another performs better at higher wind speeds.
- The forecast horizon-dependent weight indicates the performance of a source model for various lead times on the target task, such as predictions ranging from 24 to 48 hours ahead.

Figure 3 illustrates the structure of the CSGE. The CSGE consists of M ensemble members, where $m \in \{1, \dots, M\}$. Each ensemble member corresponds to a source model equipped with its predictive function f_m . Every source model produces a univariate forecast $\hat{y}_{t+k|t}^{(m)} \in \mathbb{R}$ for a given input $x_{t+k|t} \in \mathbb{R}^D$ of the target task T . The subscript T is omitted here for improved readability. Let D denote the dimensionality of the input feature vector x . The index k represents the forecast horizon, referenced by the subscript, relative to the forecast origin t . For each prediction generated by each source model, an aggregated weight $w_{t+k|t}^{(m)}$ is calculated.

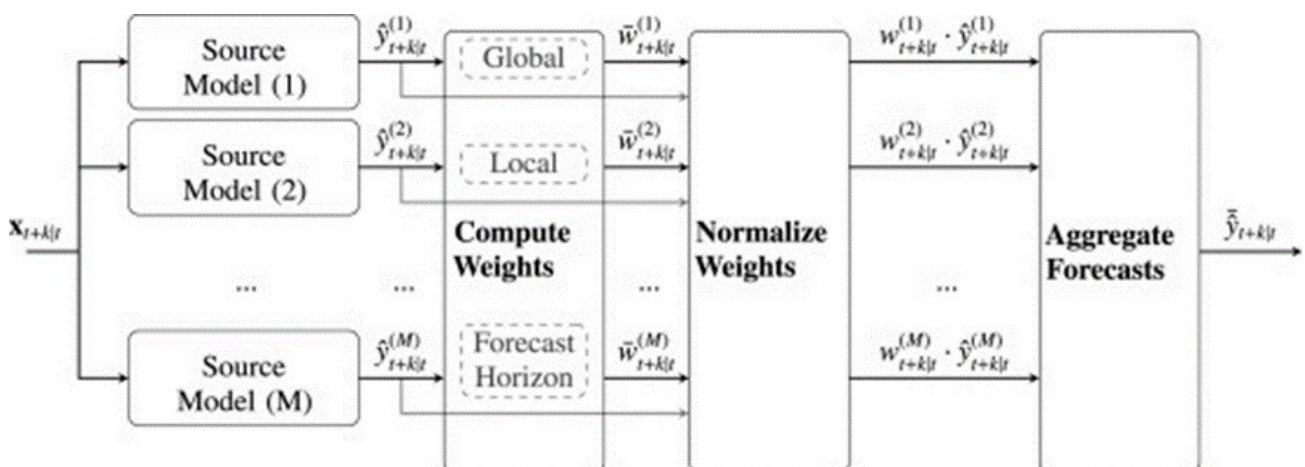


Figure 3. The architecture of the CSGE. Predictions $\hat{y}_{t+k|t}^{(m)}$ generated by the source models for a given input x are fed into the CSGE. The weights for each ensemble member are determined by combining its global, local, and forecast-horizon-dependent components. These weights are then normalized. In the final stage, the source models' predictions are multiplied by their respective weights and summed to produce the ensemble output.

The combined weight for a single source model and lead time, which integrates the global weight w_g , the local weight w_l , and the horizon-dependent weight w_h , is computed as

$$\bar{w}_{t+k|t}^{(m)} = w_g^{(m)} \cdot w_l^{(m,t)} \cdot w_h^{(m,k)}, \quad (14)$$

Where $\bar{w}_{t+k|t}^{(m)}$ is scaled so that the weights across all models sum to one, yielding the final weight $w_{t+k|t}^{(m)}$.

To determine the weights $w_{t+k|t}^{(m)}$, we apply the inverse similarity measure S^{-1} together with the cooperative soft gating rule defined in [Eq. 15](#).

$$\zeta'_\eta(\Phi, \phi) = \frac{\sum_{j=1}^J \Phi_j}{\phi^\eta + \epsilon} \quad (15)$$

By using the inverse S^{-1} (i.e., the nRMSE) to derive the weights, we quantify each source model's performance on the target task. Suppose $\Phi \in \mathbb{R}^J$ holds all $J \in \mathbb{N}$, $J \geq 1$, estimates computed from the nRMSE, with $\phi \in \Phi$. Here, $\eta \geq 0$ controls the strength of the exponential weighting, and the small positive constant $\epsilon > 0$ prevents division by zero. Larger values of η cause the CSGE to behave more like a gating ensemble, effectively focusing on only a few strong source models. Smaller values of η lead to a softer weighting ensemble. Once all weights have been calculated from the elements in Φ using [Eq. 15](#), they are normalized to sum to one, producing the final weights $w_{t+k|t}^{(m)}$. This procedure is carried out independently for each of the three weighting dimensions, following the methodology described in Ref. [\[47\]](#).

Experimental Evaluation

The next part of the paper outlines the overall experimental design. We performed experiments across six datasets, totaling 667 parks. Thanks to the cross-validation scheme employed, every park serves as a target park exactly once. This setup enables the most comprehensive evaluation to date of inductive transfer learning applied to renewable power forecasting.

Overall experimental setup

Data pre-processing steps follow the procedures used in Refs. [\[3, 7\]](#) to ensure direct comparability with existing state-of-the-art methods. We employed three types of source models: BELM, MLP, and TCN. To provide a strong, reliable baseline that performs well even with limited data and helps control overfitting, we trained a GBRT model for each target task as described in Ref. [\[7\]](#).

Datasets

All experiments focus on day-ahead forecasts, covering lead times of 24 to 48 hours. The datasets, summarized in [Table 2](#), use numerical weather prediction (NWP) variables as input features, including wind speed, wind direction, air pressure, and solar radiation. These weather forecasts are aligned with historical power measurements, which serve as the target variable for day-ahead prediction across all datasets. The weather data originates from either the European Center for Medium-Range Weather Forecasts (ECMWF) or the Icosahedral Non-Hydrostatic-European Union (ICON-EU) model.

The datasets differ in the number of input features, temporal resolution, and the quantity of samples available for training and testing. For example, the PVOPEN dataset contains 47 features, many of which are hand-crafted to capture seasonal solar patterns. In contrast, the other datasets do not include such manually engineered features.

Table 2. Overview of the evaluated datasets.

Dataset	#parks	#features	#train	#mean	resolution	NWP
Empty Cell	Empty Cell	Empty Cell	Samples	Samples	Empty Cell	Model
PVOPEN[3]	21	47	6336	8424	Hourly	ECMWF
PVSYN[7]	114	20	30 385	14 920	15-min	ICON-EU
PVREAL	42	25	58 052	19 344	15-min	ICON-EU
WINDOPEN[3]	45	13	27 724	26 636	15-min	ECMWF
WINDSYN[7]	260	29	33 714	16 678	15-min	ICON-EU
WINDREAL	185	33	36 129	12 092	15-min	ICON-EU

Note that four datasets—namely PVOPEN, WINDOPEN, WINDSYN, and PVSYN—were already analyzed in earlier research, such as Refs. [3, 7]. The situation differs for WINDREAL and PVREAL, which have not been explored previously. These two datasets remain unavailable to the public. Still, they offer the highest level of realism owing to their substantial variety. The WINDREAL dataset covers wind parks with 99 distinct nominal capacities, turbines from 13 manufacturers, and six hub heights. Every park is located in Germany. In the PVREAL dataset, the solar installations have 31 different nominal capacities, 10 different tilt angles, and 9 different azimuth directions, and they are likewise situated in Germany. It should also be emphasized that forecasting power output for wind parks generally poses greater challenges than for PV installations.

We applied a five-fold cross-validation procedure to every dataset. This guarantees that each park executes the target task exactly once and serves as a source park in the other four folds. Source models and their associated hyperparameters were fitted using only the training and validation splits. When training models for the target tasks, we divided the available data by season and limited historical records to 7, 14, 30, 60, or 90 days. All results shown are averaged across all target tasks and seasons. This arrangement avoids any distortion caused by seasonal patterns [26]. The input features from all datasets were normalized. Historical power values were divided by each plant's nominal capacity to allow error measures to be compared fairly. Every dataset was converted to a 15-minute resolution, except PVOPEN, which was adjusted to hourly resolution due to its originally coarser sampling frequency. The WINDSYN and PVSYN datasets include a predefined test set. For WINDOPEN and PVOPEN, we followed the same protocol as [3]: the initial year served as training data and the subsequent period as the test set. Because the PVREAL and WINDREAL datasets differ significantly in the length of available history, we randomly selected 25% of the days for testing. As every day has its own independent day-ahead weather forecast, no information from the future influences earlier periods [34]. From the leftover days, 25% were set aside for validation, and the remaining 75% were used for training.

Source models

As noted earlier, predictions of renewable power depend heavily on meteorological conditions. Consequently, the model inputs are themselves outputs from the numerical weather prediction system. These can be fed directly into models, such as an MLP, to estimate expected power generation for the coming day. Hyperparameter search for the models was performed using a tree-structured Parzen sampler, running 200 trials on the validation data.

Overall, four model types were trained. The BELM stands out as a useful source model because it enables straightforward similarity assessment based on the evidence and allows model updates with only linear computational scaling. We included an MLP because it remains a standard choice in industrial renewable power forecasting applications [3]. A TCN architecture was also trained to better handle periodic patterns in the data, in line with the approach in Ref. [3]. Finally, a GBRT model was fitted to provide a solid baseline capable of good generalization even when data is scarce [7].

Evaluation method

We computed the normalized root mean square error (nRMSE) on the test portion using Eq. 7 for every possible pairing of season and training data length. For each dataset, season, and training duration, we derived the mean performance rank based on nRMSE scores. To check for meaningful gains over the baseline, we applied the Wilcoxon signed-rank test with significance level $\alpha = 0.01$ across all parks in the dataset (Tables 3-8).

Table 3. Rank summary for all models, selections, and adaptation strategies on the PV datasets, cf. Table 1. Only entries that reach the highest ranks in a given dataset are displayed. GBRT functions as the baseline, and every model is checked to see whether its forecast error is significantly better ($\vee$), significantly worse ($\wedge$), or statistically equivalent ($\diamond$) at $\alpha = 0.01$. This statistical comparison is conducted over all parks inside each dataset for the indicated number of training days. The color scheme reflects rank quality: blue indicates lower (better) ranks, and red indicates higher (worse) ranks.

Data type	#Days	Baseline	BELM-EV	BELM-RM	MLP-EV-DILI	MLP-RM-DI	TCN-EV-BT	TCN-EV-DI	TCN-EV-DILI	TCN-EV-WD	TCN-RM-DI
PVOPEN	7	6.631	3.524	4.107	6.036	4.107	6.845	5.679	5.179	6.810	4.369
PVREAL	7	8.786	5.173	4.821	6.012	4.423	5.518	5.048	4.464	5.250	3.548
PVSYN	7	8.636	4.649	4.263	6.928	3.866	6.015	4.002	5.189	5.719	3.482
PVOPEN	14	6.143	3.738	3.357	5.298	4.155	7.524	5.857	5.298	7.417	4.726
PVREAL	14	8.619	5.125	4.744	5.101	4.137	6.268	5.351	4.119	6.190	3.345
PVSYN	14	8.252	4.691	4.408	6.171	3.750	6.904	3.936	4.329	6.785	3.217
PVOPEN	30	5.619	3.464	3.583	5.702	4.536	7.000	6.452	4.952	6.881	5.286
PVREAL	30	8.417	5.423	5.095	3.875	4.250	5.875	6.214	3.631	6.048	4.018
PVSYN	30	8.015	4.930	4.544	6.140	4.050	6.432	4.268	4.042	6.279	3.542
PVOPEN	60	5.440	3.726	3.845	5.321	4.952	6.762	6.940	4.679	6.655	5.179
PVREAL	60	8.095	5.726	5.244	4.286	4.565	5.786	5.732	3.452	5.613	3.792
PVSYN	60	7.509	4.886	4.733	6.401	4.220	6.312	4.130	3.978	6.267	3.608
PVOPEN	90	5.000	3.512	4.167	5.393	4.524	6.976	6.690	5.345	6.869	5.095
PVREAL	90	8.077	5.815	5.173	4.107	4.833	5.452	5.661	3.685	5.661	3.905
PVSYN	90	7.192	5.100	5.076	6.508	4.327	5.986	4.516	3.800	6.092	3.749

Table 4. Rank summary for all source models, selections, and adaptation strategies on the wind datasets. (Tables 1 and 3).

Data type	#Days	Baseline	BELM-EV	BELM-RM	MLP-EV-DILI	MLP-RM-DI	TCN-EV-BT	TCN-EV-DI	TCN-EV-DILI	TCN-EV-WD	TCN-RM-DI
WINDOPEN	7	7.387	4.012	5.526	6.734	4.861	5.382	6.399	4.532	5.133	3.832
WINDREAL	7	8.476	4.478	4.305	6.807	4.252	5.469	6.152	5.103	5.378	3.407
WINDSYN	7	7.863	5.165	4.096	6.644	4.264	5.382	6.305	5.183	5.544	3.688
WINDOPEN	14	6.671	4.422	5.850	6.306	4.919	5.566	6.387	3.965	5.468	3.780

WINDREAL	14	7.892	4.661	4.872	6.223	4.368	5.933	6.248	4.048	5.905	3.386
WINDSYN	14	7.601	4.847	4.516	5.892	4.290	5.997	6.472	4.365	6.062	3.684
WINDOPEN	30	5.156	5.527	6.365	6.892	5.048	5.407	6.246	3.856	5.186	3.862
WINDREAL	30	6.848	5.352	5.779	5.918	4.806	5.525	6.514	3.566	5.574	3.589
WINDSYN	30	6.669	4.812	4.926	5.447	4.833	5.864	6.827	4.081	6.108	4.049
WINDOPEN	60	4.250	5.974	6.467	6.711	5.414	5.421	6.789	3.289	5.309	4.000
WINDREAL	60	5.665	5.804	6.052	5.819	4.958	5.646	6.758	3.580	5.628	3.801
WINDSYN	60	5.947	5.116	5.304	5.614	4.973	5.590	7.203	3.891	5.724	4.351
WINDOPEN	90	3.992	6.289	6.969	7.070	5.719	4.648	6.406	3.516	4.805	3.969
WINDREAL	90	5.017	5.876	6.212	6.071	5.143	5.569	6.944	3.463	5.604	3.836
WINDSYN	90	5.008	5.212	5.394	5.478	5.281	5.701	7.359	3.880	5.879	4.518

Table 5. Rank summary of ensembles on the PV datasets. The best-performing model, namely TCN-EV-DILI, serves as the baseline here. (Tables 1 and 3).

Data type	#Days	Baseline	BMA-BELM	BMA-MLP	BMA-TCN	CSGE-MLP-DI	CSGE-MLP-DILI	CSGE-MLP-DILI-GBRT	CSGE-TCN-DI	CSGE-TCN-DILI	CSGE-TCN-DILI-GBRT
PVOPEN	7	7.381	3.143	3.643	4.167	4.429	5.702	7.571	4.952	5.667	7.476
PVREAL	7	6.524	5.084	4.187	3.807	4.506	5.910	8.367	4.066	4.602	7.904
PVSYN	7	5.053	6.007	6.031	5.921	3.739	5.254	7.805	3.596	4.447	7.031
PVOPEN	14	7.512	3.643	3.512	4.333	4.857	5.262	7.298	5.071	5.500	7.571
PVREAL	14	6.405	5.869	4.149	3.905	5.155	5.161	7.827	4.714	4.536	7.268
PVSYN	14	4.353	6.943	6.417	6.566	4.566	4.542	6.969	4.246	3.998	6.346
PVOPEN	30	7.607	3.964	3.476	4.310	5.655	4.940	6.238	6.274	5.690	6.393
PVREAL	30	6.476	7.071	3.929	3.899	6.208	4.369	6.643	5.673	4.214	6.518
PVSYN	30	3.789	7.625	7.138	7.478	5.548	4.075	5.226	5.213	3.958	4.943
PVOPEN	60	7.548	4.214	3.607	4.679	5.929	4.881	5.833	6.464	5.488	6.262
PVREAL	60	6.464	7.565	4.131	4.542	6.440	4.077	5.988	6.125	4.065	5.601
PVSYN	60	3.623	7.401	7.554	7.879	5.776	4.426	4.455	5.404	4.229	4.253
PVOPEN	90	8.131	3.964	3.571	4.833	6.119	4.833	4.940	6.631	6.048	5.857
PVREAL	90	6.631	7.768	4.310	4.810	6.482	4.024	5.113	6.244	4.458	5.161
PVSYN	90	3.714	7.451	7.800	8.281	5.719	4.116	4.116	5.495	4.181	4.127

Table 6. Rank summary of ensembles on the wind datasets. (Table 5)

Data type	#Days	Baseline	BMA-BELM	BMA-MLP	BMA-TCN	CSGE-MLP-DI	CSGE-MLP-DILI	CSGE-MLP-DILI-GBRT	CSGE-TCN-DI	CSGE-TCN-DILI	CSGE-TCN-DILI-GBRT
WINDOPEN	7	5.817	4.787	4.065	3.598	5.657	6.254	7.751	5.296	4.586	7.006
WINDREAL	7	6.363	5.756	4.975	3.710	4.724	5.360	7.926	5.068	4.079	6.932
WINDSYN	7	5.810	5.540	5.046	3.848	5.193	5.880	7.595	5.222	4.313	6.401
WINDOPEN	14	5.633	5.521	3.917	3.740	6.639	5.373	6.976	6.734	4.148	6.195
WINDREAL	14	6.184	7.077	4.705	3.919	6.066	4.504	6.582	6.440	3.807	5.686
WINDSYN	14	5.218	6.681	4.700	4.232	6.637	4.957	6.200	6.881	4.085	5.302
WINDOPEN	30	5.221	6.209	4.074	4.368	7.669	5.184	5.301	7.650	4.595	4.669
WINDREAL	30	5.699	7.646	4.927	4.512	6.753	4.275	5.186	7.170	4.113	4.710
WINDSYN	30	5.061	7.478	4.683	4.596	7.477	4.365	4.877	7.683	4.219	4.530
WINDOPEN	60	4.966	6.527	4.791	5.014	7.912	4.831	4.493	8.182	4.446	3.838
WINDREAL	60	5.757	7.894	5.225	4.882	7.295	4.182	4.052	7.613	4.234	3.865
WINDSYN	60	4.468	7.216	5.569	5.212	7.791	4.453	4.177	7.766	4.401	3.945
WINDOPEN	90	4.984	6.685	4.694	5.161	8.282	4.919	4.089	8.185	4.484	3.516
WINDREAL	90	5.656	7.979	5.307	5.097	7.376	4.181	3.633	7.697	4.403	3.660
WINDSYN	90	4.627	7.405	5.602	5.599	8.087	4.131	3.514	8.009	4.437	3.589

Table 7. Mean nRMSE of ensembles on the PV datasets. The best-performing model , namely TCN-EV-DILI, serves as the baseline here. (Tables 1 and 3)

Data type	#Days	Base-line	BMA-BELM	BMA-MLP	BMA-TCN	CSGE-MLP-DI	CSGE-MLP-DILI	CSGE-MLP-DILI-GBRT	CSGE-TCN-DI	CSGE-TCN-DILI	CSGE-TCN-DILI-GBRT
PVOPEN	7	0.087	0.070	0.075	0.074	0.073	0.079	0.083	0.073	0.079	0.084
PVREAL	7	0.109	0.100	0.101	0.100	0.099	0.106	0.119	0.098	0.102	0.117
PVSYN	7	0.100	0.096	0.101	0.098	0.090	0.101	0.109	0.090	0.097	0.106
PVOPEN	14	0.084	0.070	0.073	0.073	0.072	0.075	0.081	0.072	0.075	0.081
PVREAL	14	0.103	0.100	0.099	0.098	0.097	0.100	0.107	0.097	0.099	0.106
PVSYN	14	0.089	0.095	0.094	0.094	0.088	0.089	0.097	0.087	0.088	0.096
PVOPEN	30	0.077	0.069	0.071	0.072	0.072	0.072	0.075	0.072	0.073	0.075
PVREAL	30	0.097	0.100	0.094	0.094	0.097	0.094	0.099	0.096	0.094	0.099
PVSYN	30	0.084	0.094	0.091	0.092	0.086	0.084	0.087	0.085	0.084	0.087
PVOPEN	60	0.076	0.068	0.070	0.070	0.071	0.071	0.072	0.072	0.071	0.073

PVREAL	60	0.095	0.099	0.093	0.094	0.096	0.093	0.096	0.096	0.093	0.096
PVSYN	60	0.082	0.092	0.091	0.092	0.085	0.084	0.084	0.085	0.084	0.084
PVOPEN	90	0.077	0.068	0.070	0.070	0.071	0.071	0.071	0.072	0.072	0.072
PVREAL	90	0.096	0.098	0.093	0.094	0.096	0.093	0.094	0.096	0.094	0.095
PVSYN	90	0.081	0.091	0.091	0.091	0.084	0.082	0.083	0.084	0.082	0.083

Table 8. Mean nRMSE of ensembles on the wind datasets. (Table 7).

Data type	#Days	Base-line	BMA-BELM	BMA-MLP	BMA-TCN	CSGE-MLP-DI	CSGE-MLP-DILI	CSGE-MLP-DILI-GBRT	CSGE-TCN-DI	CSGE-TCN-DILI	CSGE-TCN-DILI-GBRT
WINDOPEN	7	0.176	0.170	0.166	0.163	0.167	0.172	0.187	0.167	0.166	0.184
WINDREAL	7	0.152	0.151	0.147	0.140	0.142	0.148	0.166	0.142	0.141	0.161
WINDSYN	7	0.184	0.191	0.174	0.167	0.174	0.178	0.193	0.174	0.169	0.186
WINDOPEN	14	0.165	0.167	0.160	0.157	0.166	0.162	0.173	0.166	0.158	0.169
WINDREAL	14	0.140	0.149	0.136	0.133	0.140	0.137	0.145	0.140	0.133	0.141
WINDSYN	14	0.163	0.187	0.161	0.158	0.170	0.162	0.167	0.171	0.157	0.163
WINDOPEN	30	0.158	0.164	0.153	0.154	0.165	0.154	0.156	0.165	0.153	0.155
WINDREAL	30	0.134	0.146	0.133	0.131	0.139	0.132	0.135	0.139	0.131	0.133
WINDSYN	30	0.153	0.182	0.153	0.152	0.166	0.151	0.154	0.166	0.150	0.152
WINDOPEN	60	0.146	0.156	0.146	0.146	0.158	0.145	0.145	0.158	0.145	0.144
WINDREAL	60	0.133	0.143	0.132	0.131	0.139	0.131	0.131	0.139	0.130	0.130
WINDSYN	60	0.150	0.176	0.153	0.153	0.165	0.150	0.150	0.165	0.150	0.149
WINDOPEN	90	0.141	0.150	0.140	0.141	0.152	0.140	0.139	0.152	0.140	0.138
WINDREAL	90	0.132	0.140	0.130	0.130	0.138	0.129	0.129	0.138	0.129	0.129
WINDSYN	90	0.147	0.173	0.150	0.151	0.162	0.147	0.146	0.161	0.147	0.146

Experiment on model selection and model adaptation

This part of the study conducts an experiment that jointly addresses research questions 1 and 2. Evaluating any selection method only makes sense after the chosen model has been adapted to the target:

Research Question 1: What constitutes a suitable similarity measure for selecting an appropriate source model for a new target park from a hub of pre-trained models?

Research Question 2: What is the most effective adaptation strategy once a source model has been selected?

Key findings for questions 1 and 2

The choice of model selection technique and the adaptation approach are tightly interconnected. When target training data is scarce (7–30 days), simply picking the model with the lowest forecast error and applying it without any modification often yields top-tier results. As more data becomes available, swapping the final layer for a Bayesian linear regression model proves more effective. None of the conventional fine-tuning strategies rank among the leading options.

Experimental setup

We used the source models described earlier. Fine-tuning was limited to 1 epoch, with hyperparameters tuned via grid search on 30% of the available target samples.

For weight-decay-based adaptation, seven logarithmically spaced learning rates from 10^{-1} to 10^{-4} were evaluated, mirroring the setup in Ref. [4]. Similarly, seven grid values for the penalty coefficient λ were tested in logarithmic space between 10^{-6} and 10^{-3} . The same learning rate was used for both Bayesian tuning and source-based weight decay. The penalty λ for the Bayesian loss took values from the set {0.1, 0.25, 0.5, 1, 2, 4, 8}, while for source weight decay, λ was chosen from {1, 0.1}. The training data was randomly shuffled at each epoch. The remaining adaptation methods required no hyperparameter tuning.

Detailed findings

The strongest methods are listed in **Tables 3 and 4**. Only approaches that reached the top four ranks for at least one dataset are included. The BELM frequently ranks among the best performers and consistently outperforms the baseline across training lengths up to 30 days. With 14 days or less of data, applying a selected model directly without further adaptation tends to be advantageous. Starting at 30 days, training a Bayesian linear regression on features extracted from the source model, together with the target's historical power measurements, becomes clearly beneficial — especially on the more diverse WINDREAL and PVREAL datasets. This pattern arises because features derived from a model trained on a single park do not transfer well to other locations. Additional target data is therefore necessary for the BLR to learn and compensate for site-specific differences.

These observations lead to two important practical insights. First, gradient-based optimization carries a notable risk of catastrophic forgetting, which is particularly undesirable for model hubs used in safety-critical domains such as renewable power forecasting. Second, the convex optimization underlying Bayesian linear regression provides a more stable training process, greatly reducing this risk. Surprisingly, neither weight decay nor Bayesian tuning ranked among the best performers in our tests, even though fine-tuning is a standard practice in many other fields. The reason appears to be that source models trained on individual parks are often too narrowly specialized. Small differences in weather regimes or plant characteristics (e.g., turbine type) can then trigger catastrophic forgetting during fine-tuning.

Further analysis indicates that evidence-based selection works best for the TCN across all fine-tuning variants. For the MLP, selection based on nRMSE is more effective. The probabilistic character of the evidence likely captures similarities between source and target more thoroughly, which benefits the convolutional layers in the TCN. Updating the MLP's final layer, however, only needs nRMSE-based selection.

Experiment on model combination

This section describes the experiment conducted to answer research question three.

Research Question 3: Do ensemble strategies provide advantages over the selection and adaptation of a single source model when combining knowledge from multiple sources?

Key findings for research question 3

Ensemble methods yield statistically meaningful improvements over the results of the previous experiment. Bayesian model averaging performs best on simpler tasks, whereas the cooperative soft gating ensemble excels in more

challenging settings.

Experimental setup

The TCN-EV-DILI configuration serves as the baseline. For Bayesian model averaging, every source model is first adapted to the target data as described earlier. For MLP and TCN source models, the final layer is replaced by a Bayesian linear regression following the direct linear adaptation procedure. After adaptation, each model produces a predictive posterior according to Eq. 3. These posteriors are then combined using BMA via Eq. 13. Three BMA variants are tested, one per source model type.

For the CSGE, global and horizon-dependent errors are calculated with nRMSE. Local error estimation uses a k-nearest neighbor method after projecting the feature space to two dimensions via principal component analysis (PCA). Three neighbors are considered in the reduced space. The hyperparameter η is selected from $\{1, 2\}$ via grid search. Learning rates are optimized over $\{0.5, 0.1, 1 \times 10^{-3}, 1 \times 10^{-5}\}$. Six CSGE variants are evaluated in total: two without target adaptation (CSGE-MLP-DI and CSGE-TCN-DI), two with final-layer BLR updates (CSGE-MLP-DILI and CSGE-TCN-DILI), and two extended versions that additionally incorporate the GBRT as a source model (CSGE-MLP-DILI-GBRT and CSGE-TCN-DILI-GBRT).

Detailed findings

Main outcomes are presented in Tables 5 and 6. On the PV datasets, the top CSGE variants surpass the baseline in nearly every setting. BMA also delivers excellent performance on PVOPEN and on PVREAL when training data is limited to 30 days or less. For the wind datasets, BMA performs well with very little data (under 30 days). With larger training sets, the CSGE based on TCN models with replaced final layers achieves the best results. Adding the GBRT as an extra source model further boosts performance on these datasets.

This highlights the CSGE's flexibility: because weights are derived directly from target error, it can readily combine models of any type. The proposed BMA approach lacks this flexibility. However, BMA has the advantage of delivering full probabilistic forecasts, although probabilistic evaluation is not the focus of this work.

Importantly, ensemble strategies outperform the best single model from the prior experiment in almost all cases. This demonstrates that selecting and adapting one individual source model is inherently uncertain, as any single model may be overly specific to its original training conditions. Combining several models mitigates the impact of site-specific weather patterns and technical differences, leading to significantly better accuracy.

In addition to the ranking-based statistical analysis, the actual forecast error values are examined for practical relevance. The mean nRMSE results for this experiment are shown in Tables 7 and 8.

Across all six datasets, error levels drop steadily as the volume of training data increases. Among the PV datasets, PVREAL shows the highest errors; even so, the best model reaches 9.8% nRMSE with only seven days of data. On PVOPEN with seven days, the error falls to 7%, which improves upon previously published results in Ref. [3].

For WINDOPEN, the best error rates range from 16.3% (7 days) to 13.8% (90 days), in line with earlier findings [3]. The WINDSYN dataset shows the largest errors among the wind cases, ranging from 16.7% to 14.6%.

The nRMSE analysis confirms that strong prediction quality can be achieved even with very limited training data. Moreover, once more than 30 days are available, the resulting error levels often match those obtained when training with a full year of data, as reported in Refs. [3, 7].

Conclusion

We conducted an extensive evaluation of combinations of base models, selection criteria, adaptation techniques, and two ensemble strategies across six real-world datasets. This represents the broadest study so far on model-hub-based transfer learning for renewable power forecasting.

Our findings indicate that the popular practice of fine-tuning only the final layer of a neural network does not perform well in this domain. Replacing the final layer with a Bayesian linear regression fitted on extracted source features and limited target measurements delivers considerably better results, especially when starting from a temporal convolutional neural network. The difference likely stems from the fact that computer vision models are usually pre-trained on many diverse tasks, encouraging broad generalization. In contrast, renewable forecasting models are typically trained on a single, narrow task per park, limiting their transferability during fine-tuning.

For model selection, we advise using raw forecast error when the training data is less than 30 days, and switching to the evidence criterion when more data is available. We also demonstrated that ensemble combination provides substantial gains beyond relying on a single adapted model. The proposed cooperative soft-gating ensemble assigns influence based on observed target error, whereas Bayesian model averaging is particularly useful when historical data is limited.

Future work will focus on overcoming the limitations of fine-tuning with scarce data by augmenting target sets with synthetic samples. We also intend to extend the framework to multi-task learning scenarios.

List of symbols

D	Size of input features.
θ	The parameters of a linear model.
X	All input features.
y	All response features.
x	A single input vector.
y	A single response to the respective target.
$\hat{y}$	A single response prediction.
X	The set of input features.
Y	The set of response features. Input feature space. Output feature space.
N	The number of samples.
n	Index of samples.
$\mathbb{R}$	The set of all real numbers.
$\mathbb{R}^+$	Set of all non-negative real numbers.
$\mathbb{R}_{\geq 1} \mathbb{N}_{\geq 1}$	Set of all positive real numbers. The set of all positive natural numbers. Normal distribution. A task.
T	The set of all tasks. Similarity measure between two tasks.

S	Precision matrix of a linear model.
<i>m</i>	Index of a source model.
<i>M</i>	The number of source models.
<i>T</i>	The index of the target task.
$\bar{w}$	CSGE weight not normalized.
<i>w</i>	CSGE weight normalized.

Acknowledgements

We thank enercast GmbH for providing the PVREAL and WINDREAL datasets. We also thank Marek Herde, Mohammad Wazed Ali, and David Meier for their valuable input.

Conflict of interest

The authors declare the following financial interests/personal relationships, which may be considered potential competing interests: Jens Schreiber reports that financial support was provided by the German Federal Ministry of Education and Research.

Financial support

This work is the result of the project TRANSFER (01IS20020B), funded by the BMBF (German Federal Ministry of Education and Research).

Ethics statement

None

Received: 11 Sep 2024

Revised: 18 Oct 2024

Accepted: 26 Nov 2024

Published online: 20 March 2025

Rights and permissions

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

References

Schreiber J. Transfer learning in the field of renewable energies - a transfer learning framework providing power forecasts throughout the lifecycle of wind farms after initial connection to the electrical grid. In: Organic computing doctoral dissertation colloquium. Kassel University Press GmbH; 2019. p. 75-87.

Schwartz R, Dodge J, Smith NA, Etzioni O. Green AI. arXiv:1907.10597. 2019. p. 1-12.

Schreiber J, Vogt S, Sick B. Task embedding temporal convolution networks for transfer learning problems in renewable power time-series forecast. In: ECML PKDD. 2021. p. 1-16.

You K, Liu Y, Wang J, Long M, Jordan MI, Yang J. Logme: Practical assessment of pre-trained models for transfer learning. In: Proceedings of the 38th International Conference on Machine Learning (ICML). 2021. p. 12133-43.

You K, Liu Y, Wang J, Long M, Jordan MI, Yang J. Ranking and tuning pre-trained models: A new paradigm of exploiting model hubs. arXiv:2110.10545. 2021. p. 1-45.

Alkhayat G, Mehmood R. A review and taxonomy of wind and solar energy forecasting methods based on deep learning. *Energy AI*. 2021;4:100060.

Vogt S, Schreiber J. Synthetic photovoltaic and wind power forecasting data. arXiv:2204.00411. 2022.

Zheng X-W, Li H-N, Gardoni P. Hybrid Bayesian-copula-based risk assessment for tall buildings subject to wind loads considering various uncertainties. *Reliab Eng Syst Saf*. 2023;233:109100.

Alruqi M, Sharma P, Deepanraj B, Shaik F. Renewable energy approach towards powering the CI engine with ternary blends of algal biodiesel-diesel-diethyl ether: Bayesian optimized Gaussian process regression for modeling-optimization. *Fuel*. 2023;334:126827.

Said Z, Sharma P, Syam Sundar L, Nguyen VG, Tran VD, Le VV, et al. Using Bayesian optimization and ensemble boosted regression trees for optimizing thermal performance of solar flat plate collector under thermosyphon condition employing MWCNT-Fe3O4/water hybrid nanofluids. *Sustain Energy Technol Assess*. 2022;53:102708.

Li H, Chaudhari P, Yang H, Lam M, Ravichandran A, Bhotika R. Rethinking the hyperparameters for fine-tuning. In: International Conference on Learning Representations (ICLR). 2020. p. 1-20.

Li X, Grandvalet Y, Davoine F. Explicit inductive bias for transfer learning with convolutional networks. In: Proceedings of the 35th International Conference on Machine Learning (ICML). Vol. 6. 2018. p. 4408-19.

Qureshi AS, Khan A. Adaptive transfer learning in deep neural networks: Wind power prediction using knowledge transfer from region to region and between different task domains. *Comput Intell*. 2019;35(4):1088-112.

Liu X, Cao Z, Zhang Z. Short-term predictions of multiple wind turbine power outputs based on deep neural networks with transfer learning. *Energy*. 2021;217:119356.

Ju Y, Li J, Sun G. Ultra-short-term photovoltaic power prediction based on self-attention mechanism and multi-task learning. *IEEE Access*. 2020;8:44821-9.

Henze J, Schreiber J, Sick B. Representation learning in power time series forecasting. In: Deep learning: algorithms and applications. Cham: Springer; 2020. p. 67-101.

Cao L, Wang L, Huang C, Luo X, Wang J-H. A transfer learning strategy for short-term wind power forecasting. In: Chinese Automation Congress (CAC). 2018. p. 3070-5.

Cai L, Gu J, Ma J. Probabilistic wind power forecasting approach via instance-based transfer learning embedded gradient boosting decision trees. *Energies*. 2019;12(1):159.

Liu Y, Wang J. Transfer learning based multi-layer extreme learning machine for probabilistic wind power forecasting. *Appl Energy*. 2022;312:118729.

Chen J, Zhu Q, Li H, Zhu L, Shi D, Li Y, et al. Learning heterogeneous features jointly: A deep end-to-end framework for multi-step short-term wind power prediction. *IEEE Trans Sustain Energy*. 2020;11(3):1761-72.

Sheng H, Ray B, Shao J, Lasantha D, Das N. Generalization of solar power yield modelling using knowledge transfer. *Expert Syst Appl*. 2022;116992.

Yin H, Ou Z, Fu J, Cai Y, Chen S, Meng A, et al. A novel transfer learning approach for wind power prediction based on a serio-parallel deep learning architecture. *Energy*. 2021;234:121271.

Khan M, Naeem MR, Al-Ammar EA, Ko W, Vettikalladi H, Ahmad I, et al. Power forecasting of regional wind farms via variational auto-encoder and deep hybrid transfer learning. *Electronics*. 2022;11(2):206.

Almonacid-Olleros G, Almonacid G, Gil D, Medina-Quero J. Evaluation of transfer learning and fine-tuning to nowcast energy generation of photovoltaic systems in different climates. *Sustainability*. 2022;14(5):3092.

Yan C, Pan Y, Archer CL. A general method to estimate wind farm power using artificial neural networks. *Wind Energy*. 2019;22(11):1421-32.

Schreiber J, Buschin A, Sick B. Influences in forecast errors for wind and photovoltaic power: A study on machine learning models. In: *INFORMATIK 2019*. GI e.V.; 2019. p. 585-98.

Zhou S, Zhou L, Mao M. Transfer learning for photovoltaic power forecasting with long short-term memory neural network. In: *International Conference on Big Data and Smart Computing (BigComp)*. 2020. p. 125-32.

Ceci M, Corizzo R, Fumarola F, Malerba D. Predictive modeling of PV energy production: How to set up the learning task for a better prediction? *IEEE Trans Ind Inform*. 2017;13(3):956-66.

Shireen T, Shao C, Wang H, Li J, Zhang X, Li M. Iterative multi-task learning for time-series modeling of solar panel PV outputs. *Appl Energy*. 2018;212:654-62.

Tasnim S, Rahman A, Oo A, Islam S. Wind power prediction in new stations based on knowledge of existing stations: A cluster based multi source domain adaptation approach. *Knowl-Based Syst*. 2018;145:15-24.

Vogt S, Braun A, Dobschinski J, Dierolf C. Wind power forecasting based on deep neural networks and transfer learning. In: *Wind Integration Workshop*. Vol. 18. 2019. p. 8.

Schreiber J, Sick B. Emerging relation network and task embedding for multi-task regression problems. In: *International Conference on Pattern Recognition (ICPR)*. 2020. p. 2663-70.

Vapnik VN. *The nature of statistical learning theory*. New York: Springer-Verlag; 2000.

Gensler A. *Wind power ensemble forecasting [PhD thesis]*. University of Kassel; 2018. p. 204.

Zhang C, Bin J, Liu Z. Wind turbine ice assessment through inductive transfer learning. In: *IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*. IEEE; 2018. p. 1-6.

Guariso G, Nunnari G, Sangiorgio M. Multi-step solar irradiance forecasting and domain adaptation of deep neural networks. *Energies*. 2020;13(15):1-18.

Thill M, Konen W, Bäck T. Time series encodings with temporal convolutional networks. In: *Lecture Notes in Computer Science*. LNCS. Vol. 12438. 2020. p. 161-73.

Zhu R, Liao W, Wang Y. Short-term prediction for wind power based on temporal convolutional network. *Energy Rep*. 2020;6:424-9.

Yan J, Mu L, Wang L, Ranjan R, Zomaya AY. Temporal convolutional networks for the advance prediction of ENSO. *Sci Rep*. 2020;10(1).

Goodfellow I, Bengio Y, Courville A. Deep learning. Cambridge, MA: MIT Press; 2016. p. 775.

Tan C, Sun F, Kong T, Zhang W, Yang C, Liu C. A survey on deep transfer learning. In: *Lecture Notes in Computer Science*. 2018;11141:270-9.

Fawaz HI, Forestier G, Weber J, Idoumghar L, Muller PA. Transfer learning for time series classification. In: *2018 IEEE International Conference on Big Data (Big Data)*. 2019. p. 1367-76.

He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016. p. 770-8.

Bishop CM. Pattern recognition and machine learning. New York: Springer; 2006. p. 738.

Borthwick M. Math for machine learning. Cambridge University Press; 2019. p. 411.

Hoeting JA, Madigan D, Raftery AE, Volinsky CT. Bayesian model averaging: a tutorial. *Stat Sci*. 1999;14(4):382-417.

Gensler A, Sick B. A multi-scheme ensemble using cooperative soft gating with application to power forecasting for renewable energy generation. *arXiv:1803.06344*. 2018.