





Article

# Air Temperature Forecasting Using Machine Learning Techniques: A Review

Jenny Cifuentes <sup>1</sup>, Geovanny Marulanda <sup>2,\*</sup>, Antonio Bello <sup>2</sup> and Javier Reneses <sup>2</sup><sup>1</sup> Santander Big Data Institute, Universidad Carlos III de Madrid, 28903 Getafe, Spain; jcifuent@inst.uc3m.es<sup>2</sup> Institute for Research in Technology (IIT), ICAI School of Engineering, Comillas Pontifical University, 28015 Madrid, Spain; antonio.bello@iit.comillas.edu (A.B.); javier.reneses@iit.comillas.edu (J.R.)

\* Correspondence: geovanny.marulanda@iit.comillas.edu

Received: 30 June 2020; Accepted: 10 August 2020; Published: 14 August 2020



**Abstract:** Efforts to understand the influence of historical climate change, at global and regional levels, have been increasing over the past decade. In particular, the estimates of air temperatures have been considered as a key factor in climate impact studies on agricultural, ecological, environmental, and industrial sectors. Accurate temperature prediction helps to safeguard life and property, playing an important role in planning activities for the government, industry, and the public. The primary aim of this study is to review the different machine learning strategies for temperature forecasting, available in the literature, presenting their advantages and disadvantages and identifying research gaps. This survey shows that Machine Learning techniques can help to accurately predict temperatures based on a set of input features, which can include the previous values of temperature, relative humidity, solar radiation, rain and wind speed measurements, among others. The review reveals that Deep Learning strategies report smaller errors (Mean Square Error = 0.0017 °K) compared with traditional Artificial Neural Networks architectures, for 1 step-ahead at regional scale. At the global scale, Support Vector Machines are preferred based on their good compromise between simplicity and accuracy. In addition, the accuracy of the methods described in this work is found to be dependent on inputs combination, architecture, and learning algorithms. Finally, further research areas in temperature forecasting are outlined.

**Keywords:** air temperature forecasting; artificial intelligence; machine learning; neural networks; support vector machines

## 1. Introduction

Mitigating climate change is one of the biggest challenges of humankind. Despite the complexity of predicting the effects of climate change on earth, there is a scientific consensus about its negative impacts. Among them, the affectation of ecosystems, decrease of biodiversity, soil erosion, extreme changes in temperature, sea level rise, and global warming have been identified. Likewise, impacts on economy, human healthy, food security and energy consumption are expected [1,2].

Specifically, air temperature forecasting has been a crucial climatic factor required for many different applications in areas such as agriculture, industry, energy, environment, tourism, etc. [3]. Some of these applications include short-term load forecasting for power utilities [4], air conditioning and solar energy systems development [5,6], adaptive temperature control in greenhouses [7], prediction and assessment of natural hazards [8], and prediction of cooling and energy consumption in residential buildings [9,10]. Therefore, there is a need to accurately predict temperature values because, in combination with the analysis of additional features in the subject of interest, they would help to establish a planning horizon for infrastructure upgrades, insurance, energy policy, and business development [11].

Along with other atmospheric parameters, air temperature values are measured near the surface of earth by trained observers and automatic weather stations. In particular, the World Meteorological Organization facilitates the creation of worldwide standards for instrumentation, observing practices and measurements timing in order to ascertain the homogeneity of data and statistics [12]. Empirical strategies have been developed for temperature forecasting, obtaining accurate results. Their high accuracy and reliability have been very dependent on the acquired data, where most of them follow data quality standards and quality measures [13–17].

In particular, this area has become a significant field of applications of Machine Learning (ML) techniques, due to the difficulties with achieving a high accuracy in the temperature prediction. In particular, it has been proved that the volatility of temperature time series obeys nontrivial long-range correlation, presenting a nonlinear behavior [18]. In addition, these time sequences have a considerable spatial, temporal, and seasonal variability [19].

In literature, many ML-based approaches have been explored in forecasting applications. Specifically, in the air temperature time series analysis, Artificial Neural Networks (ANN) and Support Vector Machines (SVM) are the most widely implemented strategies. In particular, most of the ANN models, developed to predict temperature values, are represented by the MultiLayer Perceptron Neural Networks (MLPNN) and Radial Basis Function Neural Networks (RBFNN) [20–32], with the Levenberg–Marquardt and Gradient Descent being the most used optimization algorithms. With regard to SVM models, most of the works developed in the field involve Radial Function Base Kernels [33–38]. In terms of performance, at a global scale, SVM has reported better performance metrics in comparison with classical ANNs [39] from 1 to 20 steps-ahead. In contrast, at a regional scale, recent Deep Learning (DL) approaches have been proposed, reporting high accuracy values. Specifically, Convolutional and Long Short Term Memory (LSTM) Recurrent Neural Networks (RNN) have been used to forecast hourly air temperature with significantly small errors for 1-step ahead [40]. In turn, a similar approach was proposed by Roesch and Günther [41] to overview annual, monthly, and daily patterns associated with air temperature time series.

The primary aim of this study is to review the ML techniques proposed in the literature for air temperature forecasting and to identify research gaps. To the best of our knowledge, this is the first review in the literature of ML-based techniques focused specifically on the problem of air temperature prediction, taking into account global and regional points of view. This paper is organized as follows. In Section 2, the most used ML-based strategies are described and the relevant associated concepts are introduced. In Sections 3 and 4, the comparison of the temperature forecasting ML-based strategies, at global and regional levels, is presented. Finally, conclusions and research gaps in temperature forecasting are discussed in Section 5.

## 2. Overview of Machine Learning Based Strategies and Forecast Performance Factors

ML is defined as a branch of the Artificial Intelligence field. The main objective of the algorithms developed in this area is to obtain a mathematical model that fits the data. Once this model represents accurately known data, it is used to perform the prediction using new data. In this way, the learning process involves two steps: the estimation of unknown parameters in the model, based on a given data-set, and the output prediction, based on new data and the parameters obtained previously.

In this way, ML strategies find models between inputs and outputs, even if the system dynamics and its relations are difficult to represent. For this reason, this approach has been widely implemented in a great variety of domains, such as pattern recognition, classification, and forecasting problems. There are three common methods implemented in ML:

- Supervised Learning, which has information of the predicted outputs to label the training set and is used for the model training.
- Unsupervised Learning, which does not have information about the desired output to label the training data. Consequently, the learning algorithm must find patterns to cluster the input data.
- Semi-supervised Learning, which uses labeled and unlabeled data in the training process.

- Reinforcement Learning, which uses the maximization of a scalar reward or reinforcement signal to perform the learning process, being positive or negative based on the system goal. Positive ones are known as “rewards” while negative ones are known as “punishments”.

Considering the large amount of ML-based approaches developed in forecasting applications, this work is focused on the most widely implemented ML strategies in temperature prediction: ANN and SVM. Although these methods are trained in a supervised way, neural network algorithms, capable of unsupervised training, could be included as well [42].

In particular, it is important to note that the most used input features in this field include the previous values of temperature as well as relative humidity, solar radiation, rain, and wind speed measurements. On the other hand, the prediction evaluation measures, more frequently used in these works, to assess the performance of these algorithms, have included the Mean Absolute Percentage Error (MAPE), the Mean Absolute Error (MAE), the Median Absolute Error (MdAE), the Root Mean Squared Error (RMSE), and the Mean Squared Error (MSE). Other indices, proposed in the literature, can be used as the correlation coefficient R (Pearson Coefficient), or the index of agreement d which are usually normalized in the (0–1) range [43]. These algorithms and their particularities will be discussed in the following subsections.

### 2.1. Artificial Neural Networks

Specifically, ANNs have been widely used for classification and forecasting applications in meteorology due to their accurate results solving pattern recognition, nonlinear function estimation, and optimization problems [44]. The accuracy of their results is based on the ANN’s capability to characterize nonlinear relationships and the availability of historical data of meteorological variables, making them an attractive analysis tool for researchers around the world.

The perceptron is the basic structural element of an ANN. The inputs associated with this component are scaled by weights ( $W_i$ ), added over the  $n$  inputs  $x_i$ , translated by a bias ( $b$ ) and passed through an activation function  $f$ . The perceptron transfer function can be written as:

$$y = f\left(\sum_{i=1}^n W_{ji}x_i + b\right). \quad (1)$$

Perceptrons can be combined to form a MultiLayer Perceptron Neural Network (MLPNN). In general, the inner structure of these ANNs in prediction problems is composed by  $n$  inputs,  $m_k$  units for a single or  $k$  multiple hidden layers and a single output unit. The input layer receives the data-set for each class by means of its units that characterize the input features. The unit values of the hidden layers are defined by the sum of multiplications between the previous layer units and weights of the links connected to that node. Finally, the output layer is the final processing and its units represent the classes to be recognized or the variable to be predicted. An example of the mapping input  $x$ -output  $y$  function for an ANN with one hidden layer is defined by Equation (2):

$$y = g\left(\sum_{j=1}^{m_1} W_j f\left(\sum_{i=1}^n W_{ji}x_i + b_j\right) + c\right), \quad (2)$$

where  $g$  and  $c$  represent the activation function and bias for the output layer, respectively. With this rationale in mind, more complex architectures which include multiple hidden layers could be considered. Weights vector  $W$  characterizes the nonlinear mapping and is defined during the learning process to match the desired outputs, minimizing a defined error function; this stage is commonly called *training*. Each of the  $m$  hidden neurons are defined by an activation function that usually is represented by one of the following functions:

- Tangent Hyperbolic Function:  $f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$ ,

- Sigmoid Function:  $f(x) = \frac{1}{1+e^{-x}}$ ,
- Rectified Linear Unit (ReLU) Function:  $f(x) = \max(0, x)$ ,
- Gaussian Function:  $f(x) = e^{-x^2}$ ,
- Linear Function:  $f(x) = x$ ,

For prediction applications, in general, the output activation function is considered linear. During the generalization stage, called “recalling”, the validation on a different data-set is performed in order to evaluate the ANN performance with the weights calculated during the learning process. For temperature prediction, an example of the relationship input  $x$ -output  $y$ , considering the previous values of temperature as the input features, is presented in Equation (3):

$$T^*(t + \Delta t) = \sum_{j=1}^m W_j f\left(\sum_{i=0}^l W_{ji} T^*(t - i)\right) \quad (3)$$

As can be seen in Equation (3), the ANN representation is equivalent to the classic nonlinear auto-regressive (AR) model for prediction purposes. In this way,  $l$  can be calculated using the auto-mutual information factor like in the AR models case. Careful attention must be given to the size of training data in order to obtain the best performance during the neural network analysis. The use of few training samples could not be sufficient to compute weights that allow the generalization, while a too large number could cause the data over-fitting and require much more time for learning. A more detailed description of this method may be found in [45,46].

A broad variety of ANN architectures have been proposed for forecasting tasks. Alternative to MLPNN, a RBNN has been widely explored in air temperature forecasting. This architecture differs from the MLPNN in that the input layer is not weighted, and, based on this representation, the first hidden layer nodes receive each full input value with no modifications. Additionally, just the activation function is generally adjusted, which in most cases is set by a Gaussian activation function. In general, RBNNs involve a simpler training process because they contain fewer weights than classical MLPNNs, which leads to a good generalization and high noise tolerance.

In the ANN research line, an ML area called Deep learning (DL) has been widely implemented in many fields and applications. DL-based ANN is an approach which includes at least two nonlinear transformations (hidden layers). Their advantages lie in the ability to handle big data, and to automatically extract relevant features [47]. Different DL-based architectures have been implemented in forecasting applications; however, they have not been widely explored for the analysis and prediction of air temperature time series. Some examples of the architectures used in prediction tasks are the Recurrent Neural Networks (RNN) and the Convolutional Neural Networks (CNN). RNN, on one hand, uses the internal state of the network at the previous output as input to the model, following a chained module structure, so that the information is recurrently analyzed. Figure 1 shows this dependence structure, where  $A$  is a repeating module and  $x_t, h_t$  the input and output at time  $t$ , respectively. In traditional RNNs, this module will consist only of a single ANN.

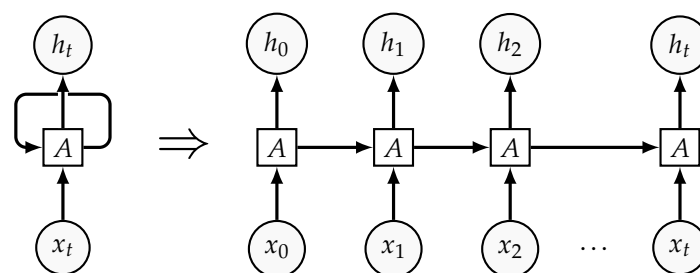


Figure 1. RNN structure.

CNNs, on the other hand, are a kind of ANN developed for feature extraction. Originally developed for two-dimensional data and image recognition, these networks perform a series of operations on the data matrix to reduce its size. One-dimensional CNN are widely used in time series forecasting problems to identify patterns on time-sequenced data.

## 2.2. Support Vector Machines

SVM algorithm, on the other hand, has been considered one of the most robust and accurate strategies among the ML-based approaches. It is a kernel-based technique developed by [48] and has been used in forecasting, classification, and regression applications. The main objective of SVM is to map the input data  $x$  into a high-dimensional feature space by means of a nonlinear mapping and generate an optimal hyper-plane  $w \cdot x + b = 0$  in this new space. In contrast to the ANN strategy which uses the training error in the optimization process, SVM seeks to minimize an upper bound of the generalization error. In order to obtain the optimal hyper-plane  $\{x \in S(w, x) + b = 0\}$ , the norm of the vector  $w$  must be minimized while the margin defined between the 2 classes  $\frac{1}{\|w\|}$  is maximized:

$$\min_{i=1, \dots, n} |(w, x) + b| = 1. \quad (4)$$

The SVM regression estimating function to get the predicted output  $y^*$  from the input data-set  $x$  is given by:

$$y^* = f(x, \alpha, \alpha^*) = \sum_{i=1}^N (\alpha_i - \alpha_i^*) K(x_i, x_j) + b, \quad (5)$$

where  $K(x_i, x_j)$  is the kernel function commonly defined as:

- A Linear kernel:  $K(x_i, x_j) = x_i^T x_j$ ,
- A Polynomial kernel:  $K(x_i, x_j) = (\gamma x_i^T x_j + r)^d$ ,
- A Radial kernel:  $K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2)$ ,
- A Sigmoid kernel:  $K(x_i, x_j) = \tanh(\gamma x_i^T x_j + r)$ ,

where  $d, r \in N$  and  $\gamma \in R^+$  are constant.  $\alpha_i$  and  $\alpha_i^*$  are Lagrange multipliers, which are solutions of a quadratic programming problem and satisfy the Karush–Kuhn–Tucker conditions. These coefficients are calculated by maximizing the following form:

$$\begin{aligned} & -\epsilon \sum_{i=1}^N (\alpha_i^* + \alpha_i) + \sum_{i=1}^N y_i (\alpha_i^* - \alpha_i) \\ & - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N (\alpha_i^* + \alpha_i) \times (\alpha_j^* - \alpha_j) K(x_i, x_j), \end{aligned} \quad (6)$$

subject to  $\sum_{i=1}^N (\alpha_i^* - \alpha_i) = 0$  with  $0 \leq \alpha_i^*, \alpha_i \leq C$ . The parameter  $C$  defines the smoothness of the approximating function and  $\epsilon$  determines the error margin to be tolerated. Lagrange multipliers  $\alpha_i$  and  $\alpha_i^*$  act as forces pushing the estimating values towards the desired output value  $y$ . The parameter  $b$  (or bias parameter) in Equation (5) requires the direct derivation of Karush–Kuhn–Tucker conditions that lead with the quadratic programming problem described. More details of this approach can be found in [49,50].

These ML-based solutions have become an alternative approach to conventional techniques and have been used in a number of meteorological forecasting applications [51–53]. It should be noted that the impact of coupling these strategies with other tools, such as principle component analysis, Kalman Filter, fuzzy logic, among others, has been studied as an interesting improvement in the estimation process performance [54–56].

### 2.3. Evaluation Measures

Based on the fact that a standard evaluation measure has not been defined for prediction, the comparison among the different forecasting strategies has become difficult. This is mainly due to the different time horizons and scales of the estimated data and the variability of the meteorological time series for diverse locations. However, some measures have been proposed to compare the predicted output  $\hat{y}$  with the observed data  $y$ . The most widely used measures to evaluate the ML strategies, implemented for forecasting tasks, are:

- Mean Absolute Error (MAE): This measure is an error statistic that averages the distances between the estimated and the observed data for  $N$  samples:

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i| \quad (7)$$

- Median Absolute Error (MdAE): This measure is defined as the median of the absolute differences  $|y - \hat{y}|$  for any  $N$  pairs of forecasts and measurements:

$$MdAE = \text{Median}(|y - \hat{y}|) \quad (8)$$

- Mean Square Error (MSE): This measure is defined as the average squared difference between the predicted and the observed temperature data, for  $N$  samples:

$$MSE = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i|^2 \quad (9)$$

- Root Mean Square Error (RMSE): This measure is the standard deviation of the difference between the estimation and the true observed data (See Equation (10)). This measure is more sensitive to big prediction errors:

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i|^2} \quad (10)$$

Although these measures have been widely used in forecasting tasks due to their simplicity, main drawbacks reported are focused on the scale dependency [57], the high influence of outliers in the prediction evaluation [58], and their low reliability [59], evidenced by the variability of the results when a different fraction of data are evaluated.

In addition to these error measures, percentage errors have been calculated as well during the evaluation in the forecasting domain. This group of measures includes:

- Mean Absolute Percentage Error (MAPE): This measure offers a proportionate nature of error with respect to the input data. It is defined as:

$$MAPE = \frac{1}{N} \sum_{i=1}^N \frac{|y_i - \hat{y}_i|}{y_i} \times 100 \quad (11)$$

- Root Mean Square Percentage Error (RMSPE) RMSPE is calculated according to:

$$RMSPE = \sqrt{\frac{1}{N} \sum_{i=1}^N \frac{|y_i - \hat{y}_i|^2}{y_i}} \times 100 \quad (12)$$

These kinds of measures are unit-free, have good sensitivity when small changes are present in data, and do not show data asymmetry [60]. In addition, these measures involve divisions by a number equal or close to zero, with errors that could be indeterminate or excessively large, and have very low outlier protection compared to other measures which have bounds for errors [59,61].

An additional group based on relative measures contains functions calculated as a ratio of mentioned above error measures by means of the estimated forecasting and reference models data. In this group, it is possible to find:

- Relative Mean Absolute Error (RMAE): This measure is computed as:

$$RMAE = \frac{MAE}{MAE^*} \quad (13)$$

where  $MAE$  and  $MAE^*$  are calculated by using Equation (7) for the forecasting model and the reference model, respectively.

- Relative Root Mean Square Error (RRMSE): This measure is calculated in a similar way to the RMAE, but in this case using the error defined in Equation (10):

$$RMAE = \frac{RMSE}{RMSE^*} \quad (14)$$

This approach, in general, establishes the number of cases when the evaluated forecasting model is superior to the reference but does not give an assessment of the difference value [62].

Likewise, additional indices have been used in the evaluation of the forecasting systems. Among them, the correlation coefficient  $R$  (Pearson Coefficient) has been defined as the co-variance between the estimated  $\hat{y}$  and the observed  $y$  data over the product of their standard deviations ( $S_{\hat{y}}, S_y$ ):

$$P = \frac{1}{N-1} \sum_{i=1}^N \left( \frac{y_i - \text{mean}(y)}{S_y} \right) \left( \frac{\hat{y}_i - \text{mean}(\hat{y})}{S_{\hat{y}}} \right) \quad (15)$$

The Index of Agreement  $d$ , on the other hand, is calculated by the expression:

$$d = 1 - \left[ \frac{\sum_{i=1}^N (y_i - \hat{y}_i)^2}{\sum_{i=1}^N (|y_i - \text{mean}(\hat{y}_i)| + |\hat{y}_i - \text{mean}(\hat{y}_i)|)^2} \right] \quad (16)$$

Based on the d-statistic, the closer the index value is to one, the better the agreement between the observed and the predicted data.

Based on the fact that each evaluation measure has the disadvantage that could guide an inaccurate evaluation of the prediction process, it has been proved that is not possible to choose only one measure. Shcherbakov et al. [62] recommend using the error measures and correlation coefficients when the analyzed time series have the same scale and when a data pre-processing has been performed. In addition, although percentage measures have been widely used in forecasting tasks, they do not recommend them due to their non-symmetry.

An additional topic that has been widely considered during the result evaluation of ML models is the Statistical Significance Analysis (SSA). While, in most applications, they have been used to select the best ML model, this tool also supports the interpretation of prediction results. In general, ML-based strategies are commonly validated using re-sampling approaches like k-fold cross-validation from which mean skill scores are directly computed and compared. Although this approach is very simple, it could be misleading as it is difficult to know if the difference between mean skill scores is real or the result of a statistical fluke. In this context, SSA is proposed to overcome this limitation and quantify the likelihood of the samples of skill scores being measured, given the assumption that they were drawn from the same distribution. If this hypothesis (often called null hypothesis) is rejected, it indicates that the difference in skill scores is statistically significant. As such, SSA is considered very

useful to improve both model reliability and results interpretation and presentation during the model selection process. Forecasting applications have included, for instance, normality tests to confirm that data-sets are (or are not) normally distributed, parametric statistical significance tests for normally distributed results, or non-parametric statistical significance tests for more complex distributions of results.

#### 2.4. Input Features, Time Horizon, and Spatial Scale

ML strategies have become an alternative approach to conventional techniques and are used in a number of different applications for modeling, prediction, and forecasting of temperature values. For the particular case of forecasting applications, three considerations on the input features can be envisioned to characterize the model:

- The model is based on other meteorological or geographical variables (e.g., solar radiation, rain, relative humidity measurements, among others).
- The model only takes into account the historically observed data of temperature as system input.
- The model takes a combination of both temperature values and other parameters, to perform the prediction.

Likewise, it is important to underline that one of the most intuitive criteria that impacts the prediction performance is the forecast time horizon (known as the look-ahead or lead time). The forecast time horizon in temperature prediction, defined as the length of time into the future for which prediction is performed, is characterized in terms of a long-term and a short-term estimation. The n-months ahead forecasts are designated as long-term forecasts. Alternatively, the short time horizon is defined as a n-hours or n-days ahead prediction.

An additional factor that has a significant impact on forecast performance is the spatial scale. Due to the well-known aggregation effect, forecasts for geographically diverse stations, which aggregate in a global scale prediction, usually have smaller errors than the forecasts for individual meteorological stations in a regional scale. Local effects, which are more random, are often more difficult to predict in the temperature trends. In this way, this paper reports the air temperature values forecasting, performed at a global and regional scale. Specifically, at a regional level, hourly, daily and monthly predictions have been envisioned, based on the particular applications of the different forecasting systems.

### 3. Long-Term Global Temperature Forecasting

Different research papers have reported that the climate will warm over the coming century, as a reaction to the changes in the anthropogenic emissions of CO<sub>2</sub> [63]. Then, there is an increasing involvement of science and scientists to characterize the impacts of global climate change on decadal [64] or longer time scales [65], in order to structure prospects for global policy actions. This variability has been studied in response to the Global mean Temperature rise that the earth has been experienced since pre-industrial times. Therefore, this section details the application of ML-based strategies in global temperature forecasting using a variety of meteorological variables.

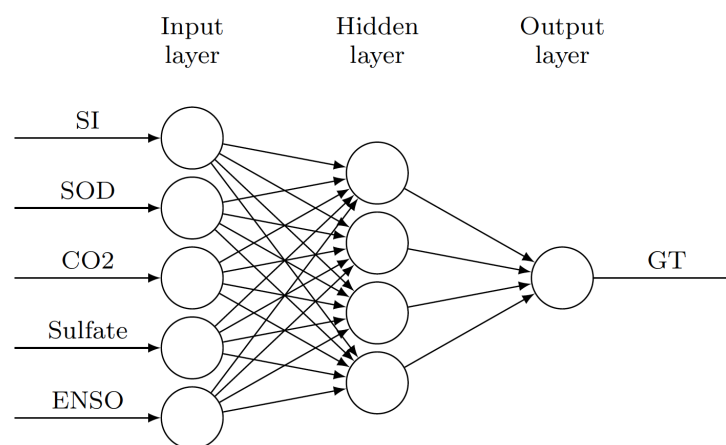
Miyano and Girosi [20] applied an MLPNN using back-propagation and the generalized delta rule, a stochastic gradient descent algorithm, to predict Global Temperature (GT) variations. They used 45 data points (1861–1909) for training and tested the approach by means of three data-sets: 1910–1944, 1910–1964, and 1910–1984 obtaining a RMSE of 0.12 °C, 0.13 °C and 0.15 °C, respectively.

Knutti et al. [66] propose a neural network based climate model to predict ranges for climate sensitivity. Data used for the estimation process include the observed surface warming over the industrial period and estimates of global ocean heat uptake. The neural network structure implemented includes 10 neurons, Sigmoid and Linear activation functions for the input and the hidden layer, respectively, and the Levenberg–Marquardt algorithm as optimizer strategy. Although the surface warming calculated from the climate model fits well with observations, some features like



the almost constant temperatures found in 1940–1970 and the strong warming after 1980 are not well simulated.

Pasini et al. [67] used an ANN for GT anomalies estimation based on global physical-chemical forcing and circulation patterns. The GT is estimated as a function of parameters combination obtained from natural/anthropogenic forcings and an inter-connected ocean-atmosphere circulation pattern (called El Niño Southern Oscillation—ENSO) since 1866 (See Figure 2). Solar Irradiance (SI) and Stratospheric Optical Depth (SOD) are considered as indices of natural forcings on the climate system, while CO<sub>2</sub> concentration and sulfate emissions are characterized as anthropogenic forcings. The MLPNN includes a single layer with few (four or five) hidden neurons. It is trained by means of the generalized Widrow–Hoff rule based on gradient descent and momentum terms and the activation function is a normalized sigmoid proposed in [68]. The authors explain the physical relationship between inputs and targets by excluding some inputs–target pair from the training set. Once the network is trained, they use the excluded pairs as a validation/test set in order to assess the modeling performance on new cases that are unknown to the network. The model performance of the GT estimation strategy is presented in terms of the linear correlation coefficient  $R$  of 0.877; the highest value obtained among four scenarios proposed of input variations: Natural, Anthropogenic, Natural + anthropogenic, Natural + anthropogenic + ENSO.



**Figure 2.** ANN model to estimate Global Temperature.

Fildes and Kourentzes [69] presented an empirical evaluation of univariate and multivariate forecasting methods used to predict GT. In particular, they assessed the CO<sub>2</sub> emissions inclusion in a nonlinear multivariate neural network, by means of data obtained from the annualised HadCrut3v (a data-set of land and ocean temperatures), and total carbon emissions from fossil fuels, between 1850 and the forecast origin. The authors developed an ANN model with a single hidden layer and carried out an evaluation of the suitable amount of hidden nodes (between 1–30). They identified 11 and 8 nodes to be convenient for the univariate and multivariate ANN, respectively. The nonlinearities were modeled using the hyperbolic tangent activation function and the optimizer implemented was the Levenberg–Marquardt algorithm. Multivariate ANN (GT and CO<sub>2</sub>) for 1–4-step-ahead forecasts gets MAEs (MdAEs) of 0.104 (0.088), 0.101 (0.088) and 0.088 (0.70) for the periods 1939–2007, 1959–2007 and 1983–2005, respectively. For the 10-step-ahead forecasts, in the periods 1948–2007, 1968–2007 and 1992–2005 gets 0.165 (0.176), 0.154 (0.143) and 0.078 (0.053) and for the 20-step-ahead, for the same periods, 0.230 (0.206), 0.249 (0.228) and 0.169 (0.124), respectively.

In contrast to the ANN models, Abubakar et al. [39] proposed an SVM model to forecast the global land-ocean temperature (GLOT). Data analyzed, including rain, pressure, GT, wind speed and relative humidity, were collected from NASA’s GLOT index for the period between 1880 and 2013. SVM model waskerneled with a Radial kernel Function and the optimal values applied for  $C$ ,  $\epsilon$ ,  $\gamma$

and learning ratio  $\eta$  were 68, 0.001829, 65, and 0.06, respectively. Finally, a support vector of 7613 was chosen based on its accuracy. The performance of the model was compared with an MLPNN with one hidden layer and 11 hidden neurons, trained by means of a Levenberg–Marquardt learning algorithm. In the hidden and output layers, they included sigmoid and linear activation functions. Experimental results show MSE and RMSE of 0.004519 and 0.00121 for SVM and 0.08912 and 1.657110 for MLPNN, respectively.

Hassani et al. [70], on the other hand, predict Global Temperatures by means of 12 parametric and non-parametric univariate (only GT) and multivariate (GT and global CO<sub>2</sub> emission) models. Among the multi-regressive and the nonparametric spectral estimation algorithms, commonly used in time-series forecasting, they analyze the Neural Network Performance using the GT data obtained from the Goddard Institute for Studies (GISS), and the CO<sub>2</sub> data from the Carbon Dioxide Information Analysis Centre. The ANN implemented for the analysis is a feed-forward neural network with a single hidden layer and one hidden node. The algorithm used for training is the rprop+ and the activation function is a sigmoid. RRMSE obtained in this paper is 0.67 average for 1 to 10 steps ahead, showing higher error values compared with other competing models.

Table 1 shows the results of ANN based methods used in GT prediction. In this list, the papers propose different architectures changing input definitions, structure and training algorithms to improve the forecasting accuracy. Although a lot of work have been done for regional estimation of temperature, based on SVM and ANN, most GT forecasting ML-based strategies are focused on ANN. However, taking into account the comparison between SVM and ANN in GT prediction, developed by Abubakar et al. [39], results show a best performance for the SVM model reporting the lowest MSE and RMSE values.

**Table 1.** Representative papers related to Global Surface Temperature Prediction based on ANN

Reference	Input	Dataset	Hidden Neurons	Training Algorithm	Activation Function	Evaluation Criteria/Time Horizon
[20]	GT	[71]	4	Generalized Delta Rule	Sigmoid	1-step RMSE = 0.12 °C
[66]	Surface Warming, Global Ocean Heat Uptake	[72] [73]	10	Levenberg-Marquardt	Sigmoid -Linear	MSE $\approx$ 0.5 °K
[67]	SI-SOD CO <sub>2</sub> -Sulfate ENSO	[73], [74], [75], [76]	4.5	Widrow-Hoff Rule	Normalized Sigmoid	$R = 0.877$
[69]	GT-CO <sub>2</sub>	[77]	11.8	Levenberg-Marquardt	Tanh	1-4-step MAEs(MdAEs) = 0.088 (0.70) °C 10-step MAEs(MdAEs) = 0.078 (0.053) °C 20-step MAEs(MdAEs) = 0.078 (0.053) °C
[39]	Rain, Pressure Wind Speed, GT, Relative Humidity	[77]	11	Levenberg-Marquardt	Sigmoid -Linear	1-step MSE(RMSE) = 0.0891(1.6571) °C
[70]	GT-CO <sub>2</sub>	[77]	1	rprop+	Sigmoid	1-step RRMSE = 0.67 °C

#### 4. Regional Temperature Forecasting

Considering the strong potential impacts on climate, in response to the increase of CO<sub>2</sub> emissions, global temperature forecasting models have been proposed (e.g., General Circulation Models) in order to find strategies to mitigate the possible environmental and economic damages [78].

The resolution of these models is not high enough to give better characterizations in a regional scale. In this case, historical measurements of individual meteorological stations have been used to study the climate change in specific areas. In this section, research developed in air temperature forecasting, at a regional scale with different time horizons, are described.

#### 4.1. Hourly Temperature Forecasting

Accurate forecasting of Hourly Temperature (HT) has an important number of different applications, ranging from electricity load forecasting to crop loss prevention. The inaccuracy and lack of measured HT data avoid any measure to mitigate the damage obtained from extreme temperature events. HT prediction has been studied in different research papers [21,22].

One of the initial ANN-based schemes applied in this field was developed by Hippert et al. [21]. In this work, a hybrid forecasting system, combining a simple autoregressive model and an MLPNN, was structured to predict hourly temperatures using past observed temperatures, forecasts obtained from the linear model, extreme temperatures forecasts, provided by the Weather Service, and the hour (codified as a sinusoid in order to stress its cyclical nature). The analyzed data were collected from a weather station in Rio de Janeiro, Brazil in 1997. In the experiments, AR and ARMA models were tested in conjunction with the MLPNN. For the period February (20–24) and the combinations AR+MLPNN and ARMA + MLPNN, MAPE values were 2.82 and 2.66, respectively. These results are considerably lower than those obtained with only linear models.

Tasadduq et al. [22] implemented an MLPNN for the estimation of hourly mean values of temperature 24 h in advance. Full year hourly values of temperature are used during the MLPNN training for a coastal location—Jeddah, Saudi Arabia. The MLPNN includes only one input node, associated with the temperature of the previous day at the same hour, and is validated with the data from three different years, excluding the one used for training. The MPD calculated for every experiment is 3.16%, 4.17%, and 2.83%, respectively.

Lanza and Cosme [23] proposed a hybrid strategy for HT prediction based on a RBFNN, initialized by means of a Regression Tree. In this approach, each terminal node of the tree is connected to one hidden unit of the RBFNN. The system inputs are the current coded hour and the temperature to predict the next HT. Data used during the validation process were recorded in the Great Energy Predictor Shootout II in Texas during the period 20 May to 20 August. The proposed model is compared with a linear AutoRegressive with eXogenous inputs (ARX) model, showing a better performance with an MAE equal to 0.4466 °C in contrast to 0.5247 °C. It is important to note that a good consideration (at least for load prediction) is to obtain an MAE less or around 0.5 °C.

Abdel-Aal [3], on the other hand, estimates next-hour and next-day HT by training an Abductive Artificial Neural Network (AANN) on 5 years of data (1 January 1985–12 October 1989) and validating on data for the 6th year (1990). The data-set used includes the measured HT data from the Puget power utility in Seattle. For the next-day hourly forecasting model, 24 models for each hour of the day were implemented to estimate the following day HT in one step. Every model has the same set of inputs: 24 hourly temperatures on  $(d - 1)$ -day ( $T_1, T_2, \dots, T_{24}$ ), the measured minimum ( $T_{\min}$ ) and maximum ( $T_{\max}$ ) temperatures on  $(d - 1)$ -day, and the estimated minimum (ET $\min$ ) and maximum (ET $\max$ ) temperatures for  $d$ -day. In the same way, for the next-hour HT estimation, 24 models were implemented based on the full HT data on  $(d - 1)$  day ( $T_1, T_2, \dots, T_{24}$ ). In addition, every available HT on  $d$  day up to the preceding hour ( $NT_1, NT_2, \dots, NT_{h-1}$ ) are used together with the measured minimum ( $T_{\min}$ ) and maximum ( $T_{\max}$ ) values for the  $(d - 1)$ -day and the minimum (ET $\min$ ) and maximum (ET $\max$ ) estimated temperatures for the  $d$ -day. Next-hour and next-day hourly models obtained an overall MAE of 1.68 and 1.05 °F, respectively. These results were compared with an MLPNN, using a node configuration equal to 28-6-1 and a sigmoid transfer function, indicating inferior performance in contrast to the abductive model.

Maqsood et al. [24] used an ensemble of MLPNN, RBFN, Elman Recurrent Neural Network (ERNN), and Hopfield model (HFM), obtained by means of a constructive algorithm, to predict the 24-h-ahead weather parameters for winter, spring, summer, and fall seasons. The input and output parameters used for this analysis were related to HT, Wind Speed, and Relative Humidity values, collected at the Regina Airport by the Meteorological Department in Canada in 2001. The performance of this approach was contrasted with every strategy separately and results showed that the ensemble

of neural networks produced the most accurate forecasts. The proposed strategy can be easily implemented to address HT forecasting applications without increasing the computation complexity.

The research reported by Smith et al. [25] included the evaluation of 30 models of MLPNN to forecast HT values up to 12 hours ahead. Input data are composed by five weather variables: HT, Relative Humidity, Wind Speed, Solar Radiation, and rainfall acquired from stations located in southern and central growing regions of Georgia. MLPNN architectures analyzed in this work are based on the Ward-style, which is a network with multiple node types and activation functions. These models had a linear input layer, three equally-sized and a single, logistic output node, which represents the HT at some prediction horizon. In this case, they carried out an analysis based on the training set sizes, obtaining six models (instantiated by 30 networks) with different training patterns. The most accurate network was trained over 50,000 samples and obtained an MAE of 1.51 °C for a 4-h model. In addition, they performed a comparison for the same model with and without seasonal input terms. The most accurate model was found to be with seasonal inputs. Based on the same architecture, the authors proposed an automated year-round temperature prediction [26] using training sets of 1.25 million patterns. In this case, they also evaluated the accuracy effect of adding rainfall input terms, concluding that these additional inputs did not increase the prediction accuracy. The MAE calculation for the year-round forecasting system varied from 0.516 °C for 1-h horizon to 1.873 °C for 12-h horizon. Recently, Jallal et al. [27] proposed an autoregressive MLPNN-based model with delayed exogenous input sequence to analyze the global solar radiation to predict the air temperature in a half hour scale. The analyzed dataset contains the measurements at the weather station Agdal that is installed in the Agdal garden, Marrakesh, Morocco for the year 2014, and the model reports an MSE value of 0.272.

In contrast, SVM regression was introduced in HT prediction by Chevalier et al. [33] in 2011. In this study, identical inputs and subsets of the historical data described in [26] were included in the analysis. For the SVM regression algorithm, the penalty factor  $C$  was set to 25 and the kernel used during the experiments was a radial-based function. In this study, the kernel was arbitrarily selected because it has been shown to be a good general purpose kernel [79]. Results showed that, for a reduced training set with 300,000 patterns, the SVM strategy was slightly more accurate than the MLPNN-based method. However, the MLPNN model predicted more accurately when the number of training patterns increased to 1.25 million (See Table 2).

**Table 2.** Representative papers related to Hourly Temperature Prediction.

Reference	Input	Region	ML Algorithm	Configuration	Evaluation Criteria/Time Horizon
[21]	HT	Brazil	ARMA+MLPNN	Hidden nodes = 10, Algorithm = Levenberg–Marquardt, Activation Function = Tanh-Linear	1-step MAPE = 2.66%
[22]	$T(d - 1, h - 1)$	Saudi Arabia	MLPNN	Hidden nodes = 4, Algorithm = Batch learning	1-step MPD = 3.16%, 4.17%, 2.83%
[23]	coded h, $T(h - 1)$	Texas	RBFNN	RBF = Multi-quadratic, Model Selection = Bayesian Size (hyperrectangles, RBF centres) = 10	1-step MAE = 0.4466 °C
[3]	$(T_{\{1,2,\dots,24\}}(d - 1)),$ $T_{\max}, T_{\min}, ET_{\max}, ET_{\min}$ $(NT_{\{1,2,\dots,h-1\}}(d))$	Seattle	AANN	Models range from (single element-single layer) to (Five-input, two-element, two-layer) Complexity Penalty Multiplier = 1	Next,-h MAE (MAPE) = 1.68 F (3.49%) Next,-d MAE (MAPE) = 1.05 F (2.14%)
[24]	HT, Wind Speed and Relative Humidity	Canada	MLPNN+RBFN +ERNN+HFM	(MLPNN, ERNN): Hidden nodes = 45 Algorithm = one-step secant Activation Function: Tanh, sigmoid (RBFN): 2 hidden layers, 180 nodes Activation Function: Gaussian	24-step Winter MAE = 0.0783 °C 24-step Summer MAE = 0.1127 °C 24-step Spring MAE = 0.0912 °C 24-step Fall MAE = 0.2958 °C
[25]	Up to prior 24 h: HT, Wind Speed Rain,Relative Humidity Solar Radiation (10 k–400 k)	Georgia	Ward MLPNN	Hidden Layer = 3 parallel slabs Hidden nodes: (2–75) nodes per slab Activation Function = Gaussian, Tanh, Sigmoid	1-step MAE = 0.53 °C 4-step MAE = 1.34 °C 8-step MAE = 2.01 °C 12-step MAE = 2.33 °C
[33]			Radial-basis function kernel $\epsilon = 0.05, C = 25, \gamma = 0.0104$	1-step MAE = 0.514 °C 4-step MAE = 1.329 °C 8-step MAE = 1.964 °C 12-step MAE = 2.303 °C	
[26]	Up to prior 24 h: HT,Wind Speed Rain, Relative Humidity Solar Radiation (1.25 million)		Ward MLPNN	Hidden Layer = 3 parallel slabs Hidden nodes: 120 nodes per slab Activation Function = Tanh	1-step MAE = 0.516 °C 4-step MAE = 1.187 °C 8-step MAE = 1.623 °C 12-step MAE = 1.873 °C
[33]			SVM	Radial-basis function kernel $\epsilon = 0.05, C = 25, \gamma = 0.0104$	1-step MAE = 0.513 °C 4-step MAE = 1.203 °C 8-step MAE = 1.664 °C 12-step MAE = 1.922 °C
[27]	Global Solar Radiation	Morocco	AR + MLPNN	2 hidden layers (5 and 8 neurons) Activation function = tanh	1-step MSE = 0.272 °C
[34]	Relative humidity, Precipitation Pressure, Global Radiation HT, Wind Speed and Direction	Spain	SVM Banks	4 SVMs for: zonal, mixed, meridional, transition Gaussian Function Kernels	1-step RMSE = 0.61 °C 2-step RMSE = 0.94 °C 4-step RMSE = 1.21 °C 6-step RMSE = 1.34 °C

Table 2. Cont.

Reference	Input	Region	ML Algorithm	Configuration	Evaluation Criteria/Time Horizon
[35]	$T_{h-1}, T_{h-2}, T_{h-3}, T_{h-4}$	Saudi Arabia	LS-SVM	Radial-basis function kernel Optimal combination ( $C, \gamma$ ) for a MSE = 0.0001	1-step MAPE = 1.20%
			MLPNN	Hidden layers = 2, Hidden Nodes = 24, 19	1-step MAPE = 2.36%
			RBFNN	Hidden layers = 1, Hidden Nodes = 22	1-step MAPE = 1.98%
			RNN	Hidden layers = 1, Hidden Nodes = 17	1-step MAPE = 1.62%
			PNN	Hidden layers = 3, Hidden Nodes = 4, 3, 2	1-step MAPE = 1.58%
[80]	Previous 24 h values of HT, barometric pressure, humidity and wind speed	Nevada	SDAE	Hidden Layers = 3, Hidden nodes = 384 Learning Rate = 0.0005, Noise = 0.25	1-step RMSE = 1.38%
			MLPNN	Hidden Layers = 3, Hidden nodes = 384 Learning Rate = 0.1	1-step RMSE = 4.19%
[40]	Surface temperature and pressure, wind, rain, humidity snow, and soil temperature	Simulated Data	LSTM	5 layers, activation functions: linear, tanh Learning Rate = 0.01, Adam Optimizer	1-step MSE = 0.002041361°K
			CRNN	5 layers (Filter size: 32, 64, 128, 256,512) Learning Rate = 0.01, Adam Optimizer	1-step MSE = 0.001738656°K

In the same line of thought, Ortiz-García et al. [34] present a HT prediction system (up to 6 h ahead) based on SVM regression banks, constructed using synoptic information of the data by means of the Hess–Brezowsky classification (HBC) algorithm. For this study, seven meteorological variables were acquired from the Barcelona-El Prat International Airport automatic station (1 January 2009 to 31 December 2009), in a mean hourly scale. The authors grouped the SVM bank in terms of four synoptic variables, which characterize the atmospheric flow and weather patterns: three main groups of circulation types (zonal, mixed and meridional), and one group to cover unclassified situations called the transition situation. Then, the samples are divided and different SVMs are trained for each group. The next value predicted is obtained by checking the current synoptic situation and then applying the suitable SVMs. The authors show that this solution performs better than an alternative prediction method based on the Extreme Learning Machine (ELM) algorithm.

Mellit et al. [35] proposed an alternative variation of the traditional SVM called LS-SVM (Least Squares-Support Vector Machine), which solves linear equations instead of the classic quadratic programming problem. The data-set recorded for the air temperature prediction were acquired at Medina city (Kingdom of Saudi Arabia) during the period from 1 January–31 December 2011 with a mean hourly scale. For a single-step (1 h ahead) prediction, inputs in these experiments were associated with the previous four HT values ( $T_{h-1}, T_{h-2}, T_{h-3}, T_{h-4}$ ). Finally, the authors evaluated the effectiveness of the designed LS-SVM predictor in comparison with other ANN architectures (e.g., MLPNN, RBFNN, Recurrent Neural Network (RNN) and Probabilistic Neural Network (PNN)), concluding that the LS-SVM and PNN predictors offer more accurate results than other investigated ANN architectures.

Although most of the Deep Learning applications have been focused on classification problems, some research has successfully applied this approach in solving prediction problems. Recently, Hossain et al. [80] applied Stacked Denoising Auto-Encoders (SDAE) to predict HT based on the prior 24 h of HT meteorological data in northwestern Nevada. The results show a significant improvement in the HT prediction domain, as it achieves 97.94% accuracy compared to a simple ANN which achieves 94.92% accuracy. In addition, Hewage et al. [40] proposed a temporal modeling approach to perform the prediction based on convolutional and Long Short Term Memory (LSTM) recurrent neural networks. The validation is carried out with weather parameters obtained from GRIB data using the weather research and forecasting model. In particular, the surface temperature from January 2018 to May 2018 and for June 2018 are used for training, and test, respectively. A lower MSE is obtained for the LSTM network in comparison with the convolutional ANN-based approach.

Table 2 shows the results of ML methods used in hourly temperature forecasting. In this summary, it can be seen that research involving ANN and SVM give similar results in terms of prediction, but it can be deduced that SVM approaches are easier to use than ANN, considering the number of parameters to adjust. In addition, the optimization process for SVM could be automatic while it is more complex for the best improvements of the ANN case. However, although a few research papers have been developed using Deep Learning strategies, the latest advances have considerably improved the accuracy rates in this particular application.

#### 4.2. Daily Temperature Forecasting

In particular, Daily Temperature (DT) forecasting is a relevant issue in the energy field, since this specific variable can be used for load forecasting [81] or to estimate solar radiation [82], which is an important factor for photovoltaic farms and devices. In this case, when the predicted loads are not accurate, the power market participants are forced to buy higher-priced electricity or to sell lower-priced electricity [36]. In that context, short-term load forecasting is an important topic for the power system risk management. In literature, a relevant amount of research has addressed the study of DT forecasting by means of ML strategies. In this sense, Pal et al. [28] proposed to use a Self-Organizing Feature Map (SOFM) to find clusters in the data, and, based on these results, the training of an MLPNN for each cluster was carried out. The authors collected nine meteorological variables from the Regional

Meteorological Centre in Calcutta, India, for the period 1983–1995. In this case, input features vector contains the information of the previous three days for the daily temperature prediction. Finally, a comparison with a single RBFNN and MLPNN was developed, showing that the proposed hybrid SOFM-MLP network consistently performs better than conventional networks.

Likewise, Maqsood and Abraham [29] presented a comparative analysis of different ANN architectures (MLPNN, RBFN, and ERNN) and a proposed ensemble of these models. These strategies are trained and tested using daily weather data of temperature, wind speed, and relative humidity in southern Saskatchewan, Canada for the year 2001. According to the authors, the proposed ensemble approach produced the most accurate forecast, while the MLPNN was the architecture that obtained relatively less accurate results during the temperature forecasting. A similar analysis was proposed by Ustaoglu et al. [30] to forecast daily mean, maximum, and minimum temperature in Turkey. In this survey, the authors implemented three different ANN-based strategies: MLPMM, RBFNN, and a Generalized Regression Neural Network (GRNN). For most of the experiments involved in this work, RBFNN performances were quite satisfactory providing close estimates compared with GRNN and MLPNN.

In the same line of research, Hayati and Mohebi [31] proposes an alternative configuration for the MLPNN architecture to predict the one-day-ahead temperature for Kermanshah city, west of Iran. In this study, a three layer MLPNN with six hidden neurons was trained and tested using ten years (1996–2006) of meteorological measurements. Based on the fact that Back Propagation training algorithms are generally quite slow for practical problems, they improved the convergence times by implementing the scaled conjugate gradient.

The same architecture was proposed by Dombaycı and Gölcü [7] to predict mean ambient temperatures in Denizli, southwestern Turkey in the period 2003–2006. Final configuration differences with the previous work lie in the optimization algorithm used for the implementation and the inputs selected for the forecasting system. In order to define the optimal parameters associated with the MLPNN architecture, Abhishek et al. [32] developed a performance analysis of the maximum DT forecasting system while varying the number of hidden layers, neurons, and transfer functions. The data analyzed in this work were collected from the station Toronto Lester in Ontario, Canada from period 1999–2009. Experimental results showed the best performance for a configuration defined by a 5 hidden-layer network with 10 or 16 neurons and a tan-sigmoid transfer function. An alternative Elman ANN approach was proposed by Afzali et al. [83] to predict mean, minimum, and maximum temperature during the years 1961–2004 in Kerman city, located in the south east of Iran. The one-day and one-month ahead air temperature is predicted slightly more precisely with this approach compared to the traditional MLPNN. Furthermore, Husaini et al. [84] proposed a Recurrent Higher Order Neural Network (RHONN) called Jordan Pi-Sigma Network (JPSN) to predict next-day temperature using measurements of five years (2005–2009) from the Malaysian Meteorological Department. More accurate results are found using this strategy, in comparison with classical MLPNNs.

In addition, a combination of classical MLPNN with the Wavelet Neural Network (WNN) has been presented for DT forecasting by Rastogi et al. [85] and Sharma and Agarwal [86]. In the research developed by Rastogi et al. [85], the input is associated exclusively with DT values, while Sharma and Agarwal [86] considered the cloud density as well. Both works analyzed the data obtained in Taipei during the years 1995–1996. Experimental results reported MAE values in the range of 0.7–0.9 and 0.25–0.62 in June, July, August, and September, for [85] and [86], respectively. These values represent better results in comparison to different time-variant fuzzy time series models. Mori and Kanaoka [36]; on the other hand, they introduced SVM regression to predict daily maximum air temperature. The proposed method was applied to nine input variables for real data acquired from AMEDAS (Automated Meteorological Data Acquisition System of the Japan Meteorological Agency) in Tokyo, for summer time from 1999 to 2001.

The authors showed that, by using the SVM-based approach, the average error of 1-day ahead maximum air temperature is reduced by 0.8% and 0.1% in comparison with an MLPNN and an RBFNN.



However, this conclusion was drawn with models that were trained on a relatively small data-set containing 366 patterns and validated with 122 patterns. In a similar way, Radhika and Shashi [37] proposed an SVM to predict the maximum DT based on the daily maximum temperatures for a span of previous  $n$  days (2 to 10), measured by the University of Cambridge for the period from 2003 to 2008. Results were compared with an MLPNN, showing that, based on a proper selection of configuration parameters, SVM performs better than classical approximations of ANN. An analogous proposal has been put forward by Paniagua-Tineo et al. [38], which employed an SVM approach to model and predict maximum DT in several European countries. Weather related features, in this case, included a 10-year period of data for temperature, precipitation, relative humidity, and air pressure, specifically synoptic situation of the day and monthly cycle. The authors showed that this approach performed well when compared with MLPNNs. In this line, Wang et al. [87] improved the SVM-based temperature prediction model through the implementation of a heuristic global optimization method called Particle Swarm Optimization (PSO). The resulting PSVM approach was validated on daily minimum temperature values from 2005 to 2009 in Beijing. The experimental results showed that the proposed strategy performs better than some other SVM model such as Generalized Support Vector Machine (GSVM) and basic SVM using a considerably small sample size.

In order to enhance the performance of the SVM models for this particular task, some previous DT values have been included in the prediction system. However, taking into account several weather variables for some locations and for several days generates a large feature vector, which makes it necessary to establish a feature selection strategy to decrease the model complexity. In this way, Karevan et al. [88] presented a combination of k-Nearest Neighbor and Elastic Net (EN) to reduce the number of features. This study carries out the minimum and maximum temperature forecasting from one to up to six days ahead for Brussels, considering data from 70 stations, most of which are located in North America, Europe, and East Asia, during a period from the beginning of 2007 until mid 2014. Results are compared with an LS-SVM algorithm to show the accuracy improvement of the proposed approach.

In more recent research, Karevan and Suykens [89] takes into account the spatio-temporal properties of the same data-set to carry out the feature selection, by means of an algorithm called Least Absolute Shrinkage and Selection Operator (LASSO). A similar analysis to that described above was developed in this work for one to up to three days ahead in DT prediction for Brussels, based on meteorological data from 10 cities. The experimental results show that Spatio-Temporal LASSO improves, in most cases, the performance in comparison with the LS-SVM approach. However, results are not compared with the strategy proposed in [88].

A few research papers focused on Deep Learning have been developed in this field. Recently, Roesch and Günther [41] presented a Recurrent Convolutional Neural Network (RCNN), trained and tested on 25 years of climate data, to forecast meteorological attributes, such as temperature, pressure, and wind velocity. The authors used the ERA-Interim re-analysis of the European Centre for Medium-Range Weather Forecast (ECMWF) to get the data for training and evaluation.

In particular, around Zurich (Switzerland), they extracted a time series in a  $7 \times 7$  grid, based on spatial features. The application developed in this work allowed for overviewing annual, monthly, and daily patterns associated with the time series. Based on the previously described research, Table 3 summarizes the ML methods used in daily temperature forecasting.

**Table 3.** Representative papers related to Daily Temperature Prediction.

Ref.	Input	Region	Algorithm	Configuration	Evaluation Criteria/Time Horizon
[28]	For 3 previous days: 2 measures of mean sea level and vapor pressures, and relative humidity, Max DT, Min DT, Rainfall	Calcutta	SOFM+ MLPNN	Hidden layer = 1, Hidden Nodes = 10 Activation Function= Sigmoid	Error (Max DT) $\leq 2$ °C in 88.6% cases Error (Min DT) $\leq 2$ °C in 87.3% cases
			MLPNN	Hidden layer = 1, Hidden Nodes = 15 Activation Function= Sigmoid	Error (Max DT) $\leq 2$ °C in 83.8% cases Error (Min DT) $\leq 2$ °C in 85.2% cases
			RBFNN	Size (RBF centres) = 50	Error (Max DT) $\leq 2$ °C in 80.65% cases Error (Min DT) $\leq 2$ °C in 81.66% cases
[29]	Average DT, Wind Speed and Relative Humidity	Canada	MLPNN	Hidden layer = 1, Hidden Nodes = 45 Levenberg–Marquardt Algorithm	MAPE = 6.05% RMSE = 0.6664 °C MAE = 0.5561 °C
			ERNN	Hidden layer = 1, Hidden Nodes = 45 Levenberg–Marquardt Algorithm	MAPE = 5.52% RMSE = 0.5945 °C MAE = 0.5058 °C
			RBFNN	Hidden = 2, RBF Nodes = 180 Gaussian Activation Function	MAPE = 2.49% RMSE = 0.2765 °C MAE = 0.2278 °C
			Ensemble	Arithmetic mean and weighted average of all the results	MAPE = 2.14% RMSE = 0.2416 °C MAE = 0.1978 °C
[30]	Daily mean, maximum and minimum temperature	Turkey	MLPNN	Levenberg–Marquardt Algorithm Hidden Layers = 1, Hidden Nodes = 5	Mean RMSE (Tmean) = 1.7767 °C Mean RMSE (Tmin,Tmax) = 2.21, 2.86 °C
			RBFNN	RBF Nodes = 5–13 Spread parameter= 0.99	Mean RMSE (Tmean) = 1.79 °C Mean RMSE (Tmin,Tmax) = 2.20, 2.75 °C
			GRNN	Spread Parameter = 0.05	Mean RMSE (Tmean) = 1.817 °C Mean RMSE (Tmin,Tmax) = 2.24, 2.87 °C
[31]	Daily Gust Wind, mean, minimum and maximum DT, precipitation, mean humidity, mean pressure, sunshine, radiation and evaporation	Iran	MLPNN	Hidden layer = 1, Hidden Nodes = 6 Scaled Conjugate Gradient Activation Function (Hidden/Output) = Tanh-Sig /Pure Linear	MAE $\approx 1.7$ °C
[7]	Month of the year, day of the month and Mean DT of the previous day	Turkey	MLPNN	Hidden layer = 1, Hidden Nodes = 6, Levenberg–Marquardt Algorithm, Activation Function = Tanh-Sig	RMSE (train) = 1.85240 °C RMSE (test) = 1.96550 °C
[32]	Previous 365 DT	Toronto	MLPNN	Hidden layer (nodes) = 5 (10–16), Levenberg–Marquardt Algorithm, Activation Function = Tanh-Sig	MSE = 0.201 °C
[83]	Previous Mean, Maximum and Minimum DT	Iran	ERNN	Hidden Layers = 1, Hidden Nodes = 15 Levenberg–Marquardt Algorithm	MSE (Max DT) = 0.008 °C MAE (Max DT) = 0.064 °C
			MLPNN	Activation Function (hidden) = Tanh-Sig Activation Function (output) = Pure Linear	MSE (Max DT) = 0.008 °C MAE (Max DT) = 0.067 °C
[84]	Mean DT	Malaysia	JPSN	Hidden Nodes = 2–5, Gradient Descent Algorithm	MSE, MAE = 0.006462, 0.063458 °C
			MLPNN	Activation Function (Hidden/Output) = Sigmoid/Pure Linear	MSE, MAE = 0.006549, 0.063646 °C

Table 3. Cont.

Ref.	Input	Region	Algorithm	Configuration	Evaluation Criteria/Time Horizon
[85]	Previous DT			Window Size = 3, Hidden Layers = 2	MAE = 0.7–0.9 °C
[86]	Previous DT and cloud Density	Taipei	WNN	Feed Forward Back Propagation, Learning Rate = 0.01	MAE = 0.25–0.62 °C
[36]	Maximum, minimum and average DT, Average and Minimum Daily Humidity, Maximum Daily Wind Speed, Daily Wind Direction and Daylight, Daily Isolation	Tokyo	SVM	Mahalanobis Kernel, $\epsilon = 0.1, \gamma = 0.1$	MAPE = 2.6%
			MLPNN	Hidden layer = 1, Hidden Nodes = 12, Learning Rate = 0.2	MAPE = 3.4%
			RBFNN	RBF Nodes = 12, Learning Rate = 0.05	MAPE = 2.7%
[37]	5 previous values of DT	Cambridge	SVM	Radial Basis Function, Grid Search for optimal $C, \gamma, \epsilon$	MSE = 7.15
			MLPNN	Hidden layer = 1, Hidden Nodes = $2 \times \text{num\_inputs} + 1$	MSE = 8.07
[38]	Maximum, minimum DT, global radiation, precipitation, sea level pressure, relative humidity, synoptic situation and monthly cycle	10 stations in	SVM	Gaussian Kernel Grid Search for optimal $C, \gamma, \epsilon$	RMSE (Norway) = 1.5483 °C
		Europe	MLPNN	Levenberg–Marquardt algorithm, Sigmoid Activation Function	RMSE (Norway) = 1.5711 °C
[87]	Previous Minimum DT	Beijing	PSVM	Gaussian Kernel, $\sigma = 12.2658, \gamma = 5.5987, P_{size} = 100$	MSE = 1.1026 °C
			SVM	Gaussian Kernel, $\sigma = 9.2568, \gamma = 8.9874$	MSE = 1.3058 °C
[88]	Minimum and maximum DT, precipitation, humidity, wind speed and sea level pressure	Brussels	K-M+EN +LS-SVM	$k \in \{10, 17, 27\}$ $v \in \{0.2, 0.5, 0.8\}$	1-step MAE(MaxT) = 1.07, (MinT) = 1.15 °C 6-step MAE(MaxT) = 1.73, (MinT) = 1.50 °C
			LS-SVM	Radial Function Base Kernel Parameter Tuning: Cross-Validation	1-step MAE(MaxT) = 1.35, (MinT) = 1.38 °C 6-step MAE(MaxT) = 2.03, (MinT) = 2.34 °C
			ST-LASSO +LS-SVM	$L_1$ Penalization $v \in \{0.2, 0.5, 0.8\}$	1-step MAE(MaxT)=2.11, (MinT)=1.33 °C 3-step MAE(MaxT) = 2.44, (MinT) = 2.01 °C
			LS-SVM	Radial Function Base Kernel Parameter Tuning: Cross-Validation	1-step MAE(MaxT) = 2.21, (MinT) = 1.38 °C 3-step MAE(MaxT) = 2.40, (MinT) = 2.02 °C
[41]	Temperature, Wind and Surface Pressure	Zurich	RCNN	8 Convolutional Filters ( $3 \times 3$ ) + Max Pooling ( $2 \times 2$ ) + 2 LSTM RNN	MAE = 0.88 °K

### 4.3. Monthly Temperature Forecasting

Climate change impact assessment requires a data analysis based on the temporal resolution at which impacts occur [90]. In this way, the evaluation of the current status and the future integrity of diverse environmental features (fauna and flora), required to assess the climate change, involve the construction of monthly and annual mean temperature models.

For this purpose, Bilgili and Sahin [91] predicted Long Term monthly air temperature using an MLPNN in Turkey. Inputs in this model were associated with geographical variables (latitude, longitude, and altitude) from 76 measuring stations and time. During the validation, the values determined by the ANN model were compared with the actual data (1975–2006), obtaining a minimum MAE of 0.508 °C. These geographical inputs also were analyzed by Kisi and Shiri [92] to predict long-term monthly air temperature in Iran. In the study, they evaluated the performance of a classical ANN and an Adaptive Neuro-Fuzzy Inference System (ANFIS) model, which is a combination of an adaptive ANN and a Fuzzy Inference System (FIS). Through the evaluation process, they illustrated that ANN strategy performed better than ANFIS in the test period based on the values of RMSE, MAE, and other coefficient statistics. In the same way, De and Debnath [93] implemented an MLPNN to predict the mean monthly surface temperature in the monsoon months (June, July, and August) over India. In this case, three models were developed associated with each monsoon month for both maximum and minimum temperature for the period 1901–2003. In the majority of the cases, prediction error was below 5%.

In the same line, Ashrafi et al. [90] used the MLPNN approach to predict mean temperature values in Iran. However, in this case, input values were associated with the mean temperature, dew point temperature, relative humidity, wind speed, solar radiation, cloudiness, rainfall, station level pressure, and green house gases of nine different climatic regions. In order to predict monthly mean temperature, the system analyzed one month, six months, 12 months, and 24 months before recorded data. In addition, the authors implemented three optimization methods: back-propagation (BP), Genetic Algorithm (GA) and combined GA-Particle Swarm Optimization (PSO), showing a better performance in the BP results. Research developed by Afzali et al. [83], described in the previous section, addressed the monthly temperature prediction as well. In this case, an ENN was proposed as a suitable solution, in comparison with the MLPNN.

On the other hand, Liu et al. [94] introduced the application of Wavelet coefficients (WT), based on SVM, to predict the air temperature in Tangshan monthly. During the experiments, the authors analyzed the monthly temperature data from 1960 to 2010, indicating that the accuracy obtained by means of an SVM method based on wavelet transform is significantly higher than that based on SVM and MLPNN-based models. In this context, Salcedo-Sanz et al. [95] examined the performance of SVM and MLPNN in the problem of monthly mean air temperature prediction in Australia and New Zealand. In this work, the authors analyzed data from a total of eight stations in Australia, three urban stations (1900 to 2010), and five rural stations (1910 to 2010), and other two stations in New Zealand (1930 to 2010). A performance comparison with MLPNN was carried out to show the accuracy improvement of using SVM. A similar study was presented more recently by Papacharalampous et al. [96]. In this work, the authors evaluated SVM and MLPNN techniques to forecast mean monthly temperature observed in Greece. During the evaluation, the authors assessed the one and twelve-step ahead forecasting performance of the algorithms. Based on the findings, they suggest that neural networks algorithm can produce forecasts of many different qualities for a particular individual case, in comparison with the SVM algorithm. This fact can be evidenced in the RMSE values, which range from 0.63 °C to 6.05 °C for the MLPNN case and from 0.73 °C to 2.30 °C for the SVM approach.

## 5. Discussion and Research Gaps Identification

The comparative evaluations developed in the papers reported in this work show different factors that affect the ML strategies performance. Among them, the input features, the optimization

algorithms, the configuration parameters, and the corresponding evaluation measures are of the utmost importance. Air temperature forecasting systems have used meteorological and geographical variables as input parameters.

Among them, it can be mentioned: maximum, minimum, and average temperature, precipitation, pressure, Mean Sea Level, Wind Speed and Direction, Relative Humidity, Sunshine, Evaporation, Daylight, Time (Hour, day or Month), Solar Radiation, cloudiness, CO<sub>2</sub> emissions, latitude, longitude, and altitude. However, the maximum, minimum, and mean values of temperature are found to be the common parameter for all the research. In fact, a relevant amount of works use only these features as model inputs.

Taking into account that prediction accuracy is strongly dependent on the time period, the time horizon, and the location of the weather stations analyzed during the validation and other criteria, it is difficult to conclude about the quality of the estimations based only on the accuracy metrics (RMSE, MSE, MAE, etc). In this way, in order to perform the accuracy comparison of different prediction system, it is better to use a common data-set in the validation stage. In this context, Tables 1–4 show, when the paper reported the evaluations, the comparative results between SVM and ANN-based strategies for the same data-set.

Most of the research developed in this area (monthly, daily and hourly) are focused on ANN strategies (57%) in comparison with the other widely used strategy SVM (43%). However, it is possible to see that, when SVM and ANN were compared, in most cases, SVM reported a better performance compared with classical ANN-based strategies.

Diverse ANN models (i.e., MLPNN, RBFNN, ERNN, GRNN, JPSN, RCNN, SDAE) have been proposed for air temperature forecasting, the MLPNN and the RBFNN being the most used architectures for the ANN-based Approaches. Levenberg–Marquardt and Gradient Descent are the most used optimization algorithms, with Levenberg–Marquardt showing a better performance due to its learning rate and the smaller prediction errors. Likewise, the most used combination of activation functions reported is the Hyperbolic Tangent or the Sigmoid for the hidden Layer and the Pure Linear for the output Layer. For SVM-based approaches, Radial Function Base Kernels are the most implemented functions. In addition, a considerable amount of works use Grid Search or Cross-validation as a strategy to set the hyper-parameters involved.

Issues related to time series modeling are addressed in these research works during the corresponding algorithm's implementation. As a particular case, the data-set size is limited by the amount of measurements acquired for the analysis, unless underlying physical models or alternative simulations systems are used for data generation [40]. As such, DL-based approaches, for instance, require the acquisition of long time series or complementary simulation systems to generate enough samples to perform the training-validation process. In addition, in the research works reviewed in this paper, authors have analyzed one, two, three, or more years of temperature data as available to build ML-based models, and the training and testing data sets have a size of minimum three years and one year, respectively, in order to predict air temperature accurately.

Based on the published literature, the parameters that impact the most on the forecasting are many, so it could be problematic to take exactly the results of the parameter evaluation from other research. Based on this idea, in order to draw reliable conclusions, these reported parameters just could give an idea of the methodology developed, but they should be assessed for a data-set obtained from a new location.

In this review, some of the proposed approaches also use variations or combinations of strategies, as it can be seen in Tables 1–4. Based on the evaluation results, the ensemble of strategies or the significant variations offers a better accuracy than single classical algorithms but again the best combined strategy is difficult to define, due to the data-set changes. A considerable amount of work is required in order to determine the best ANN or SVM based methodology among those available, or the possible equivalence. In any case, this task is very difficult based only on the limited cases reported in the literature.

**Table 4.** Representative papers related to Monthly Temperature Prediction

Ref.	Input	Output	Region	Algorithm	Configuration	Evaluation Criteria/Time Horizon
[91]	Latitud, Longitude, Altitude, Month	Monthly Temperature	Turkey	MLPNN	Hidden Layer = 1, Hidden Nodes = 32 Levenberg–Marquardt algorithm Activation Function (Hidden) = Log-Sig	1-step MAE = 0.508 °C
[92]			Iran	MLPNN	Hidden Layers = 1, Hidden Nodes = 15 Levenberg–Marquardt Algorithm Activation Function = Tanh-Sig	Station with Min RMSE = 1.53 °C Station with Min MAE = 1.27 °C
[93]	January to May maximum and minimum temperature	Max and Min Monthly Temperature	India	MLPNN	Hidden Layer = 1, Hidden Nodes = 2 Steepest Descent algorithm Learning rate = 0.9	June MAE(Tmin, Tmax) = 0.0154, 0.0197 °C July MAE (Tmin, Tmax) = 0.0107, 0.0162 °C Aug MAE (Tmin, Tmax) = 0.01013, 0.0099 °C
[90]	For 1, 6, 12 and 24 months before: Mean temperature, dew point temperature, relative humidity, wind speed, solar radiation, , cloudiness, rainfall, station level pressure and green house gases	Monthly Temperature		BP- MLPNN	Not Specified	MSE (Testing) = 0.0196 °C
			Iran	GA- MLPNN	Not Specified	MSE (Testing) = 0.0224 °C
				PSO- MLPNN	Not Specified	MSE (Testing) = 0.0228 °C
[83]	Previous Mean, Maximum and Minimum Temperature	Monthly Mean,		ERNN	Hidden Layers = 1, Hidden Nodes = 15 Levenberg–Marquardt Algorithm	1-step MSE (Tmin, Tmax) = 0.081, 0.060 °C 1-step MAE (Tmin, Tmax) = 0.228, 0.193 °C
		Max, and Min Temperature	Iran	MLPNN	Activation Function (hidden) = Tanh-Sig Activation Function (output) = Linear	1-step MSE (Tmin, Tmax) = 0.083, 0.064 °C 1-step MAE (Tmin, Tmax) = 0.223, 0.201 °C
[94]	Mean Monthly Temperature	Monthly Temperature		WT+ SVM	$C = 10-20$ , $\epsilon = 0.1-0.5$ $\sigma = 0.05-0.55$ , Radial basis Kernel	Min. MSE = 0.0937 °C
			Tangshan	SVM	$C = 10-20$ , $\epsilon = 0.1-0.5$ $\sigma = 0.05-0.55$ , Radial basis Kernel	Min. MSE = 0.5451 °C
				MLPNN	Not Specified	Min. MSE = 1.0076 °C
[95]	Mean Monthly Temperature	Monthly	Australia	SVM	Gaussian Kernel Grid Search for optimal $C, \gamma, \epsilon$	Mean MAE = 1.0073 °C
		Temperature	and New Zealand	MLPNN	Levenberg– Marquardt algorithm Activation Function = Logistic	1-step Mean MAE = 1.0662 °C
[96]	Mean Monthly Temperature	Monthly		SVM	Gaussian Kernel $C = 1$ and $\epsilon = 0.1$	1-step Mean RMSE = 1.31 °C
		Temperature	Greece	MLPNN	Hidden Layers = 1, Hidden Nodes=5 Activation Function = Logistic	Mean RMSE = 1.7 °C

Considering these preliminaries, the research gaps that can be identified in this review, to continue with the research in this field, are summarized as:

- Most of the research presented in this review is focused on the local analysis of the air temperature. However, there is not an extensive study about the anomalies prediction of temperature at a global level by means of these ML-based approaches. Taking into account the robust data currently available in diverse web sites, different ML-strategies and input features could be used to accurately predict temperature anomalies at the global level.
- Research reported at the regional level has not deeply analyzed the dependency of the temperature values of the surrounding area in the temperature estimation. A study oriented to analyze the impact of using temperature values of surrounding stations as inputs, based on the distance each other, could be of particular interest.
- A large number of the works described in this review do not include a time horizon analysis. The lack of these results makes it difficult to have a better idea of the accuracy of the method proposed. Likewise, a set of evaluation measures must be calculated in order to facilitate the comparison with other methods which may use the same data-set.
- Taking into account that accuracy results strongly depend on the data-set analyzed, a comprehensive study of the influence of the data-set size for training and testing should be done to offer a more fair comparison between strategies.
- A comparative analysis with all the available ANN-based techniques (MLPNN, RBFNN, ERNN, GRNN, JPSN, RCNN, and SDAE) and SVM variations (LS-SVM, PSVM, WT+SVM) should be carried out in order to determine the best strategy and algorithms to forecast air temperature for different time horizon. In this sense, as well as it is developed in other areas, a competition using a complete standard data-set could help in this objective.
- The effect analysis of each variable, such as maximum, minimum, and average temperature, precipitation, pressure, Mean Sea Level, Wind Speed and Direction, Relative Humidity, Sunshine, Evaporation, Daylight, Time (Hour, day or Month), Solar Radiation, geographical variables (latitude, longitude, and altitude), cloudiness, and CO<sub>2</sub> emissions, used in the prediction is required to be taken into account to increase the temperature prediction accuracy.
- A further study about the feature selection, based on their relevance, should be performed. Different strategies, such as Automatic Relevance Determination, closely-related sparse Bayesian learning, or Niching genetic algorithm have not been taken into account.
- Recently, Deep Learning strategies have shown a great performance for classification tasks [97]. However, a few studies have proven, with promising results, that prediction could be accurately done by means of these techniques. More further analysis should be developed in this area.
- For the evaluation of RNN, the size of the time series required to accurately predict a single value of temperature should be studied. Likewise, a comprehensive study about the structure of the recurrent unit should be included.
- In-depth analysis using statistical significance tests is required in order to assess the forecasting model's performance in terms of its ability to generate both unbiased and accurate forecasts. In these cases, the respective accuracy is evaluated by using both error magnitude and directional change error criteria.

**Author Contributions:** Conceptualization, J.C., G.M., A.B., and J.R.; methodology, J.C., G.M., A.B., and J.R.; formal analysis, J.C., G.M., A.B., and J.R.; investigation, J.C. and G.M.; resources, J.C., G.M., A.B., and J.R.; data curation, J.C.; writing—original draft preparation, J.C.; writing—review and editing, J.C., G.M., A.B., and J.R.; visualization, J.C., G.M., A.B., and J.R.; supervision, A.B. and J.R.; All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

AANN	Abductive Artificial Neural Network
ANFIS	Adaptive Neuro-Fuzzy Inference System
ANN	Artificial Neural Network
AR	Auto-Regressive
DT	Daily Temperature
EN	Elastic Net
ENSO	El Niño Southern Oscillation
ERNN	Elman Recurrent Neural Network
GLOT	Global Land-Ocean Temperature
GSVM	Generalized Support Vector Machine
GT	Global Temperature
HFM	Hopfield model
HT	Hourly Temperature
JPSN	Jordan Pi-Sigma Network
LASSO	Least Absolute Shrinkage and Selection Operator
LS-SVM	Least Squares-Support Vector Machine
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
MdAE	Median Absolute Error
ML	Machine Learning
MLPNN	MultiLayer Perceptron Neural Network
MSE	Mean Squared Error
PNN	Probabilistic Neural Network
PSO	Particle Swarm Optimization
RBFNN	Radial Basis Functions Neural Network
RCNN	Recurrent Convolutional Neural Network
RMSE	Root Mean Squared Error
RNN	Recurrent Neural Network
SDAE	Stacked Denoising Auto-Encoders
SI	Solar Irradiance
SOD	Stratospheric Optical Depth
SOFM	Self-Organizing Feature Map
SVM	Support Vector Machine
WNN	Wavelet Neural Network

## References

1. Tol, R.S. Estimates of the damage costs of climate change. Part 1: Benchmark estimates. *Environ. Resour. Econ.* **2002**, *21*, 47–73. [[CrossRef](#)]
2. Pachauri, R.K.; Allen, M.R.; Barros, V.R.; Broome, J.; Cramer, W.; Christ, R.; Church, J.A.; Clarke, L.; Dahe, Q.; Dasgupta, P.; et al. *Climate Change 2014: Synthesis Report. Contribution of Working Groups I, II and III to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change*; Intergovernmental Panel on Climate Change: Geneva, Switzerland, 2014.
3. Abdel-Aal, R. Hourly temperature forecasting using abductive networks. *Eng. Appl. Artif. Intell.* **2004**, *17*, 543–556. [[CrossRef](#)]
4. Li, S.; Goel, L.; Wang, P. An ensemble approach for short-term load forecasting by extreme learning machine. *Appl. Energy* **2016**, *170*, 22–29. [[CrossRef](#)]
5. Ruano, A.E.; Crispim, E.M.; Conceição, E.Z.; Lúcio, M.M.J. Prediction of building's temperature using neural networks models. *Energy Build.* **2006**, *38*, 682–694. [[CrossRef](#)]
6. García, M.A.; Balenzategui, J. Estimation of photovoltaic module yearly temperature and performance based on nominal operation cell temperature calculations. *Renew. Energy* **2004**, *29*, 1997–2010. [[CrossRef](#)]



7. Dombaycı, Ö.A.; Gölcü, M. Daily means ambient temperature prediction using artificial neural network method: A case study of Turkey. *Renew. Energy* **2009**, *34*, 1158–1161. [CrossRef]
8. Camia, A.; Bovio, G.; Aguado, I.; Stach, N. Meteorological fire danger indices and remote sensing. In *Remote Sensing of Large Wildfires*; Springer: Berlin/Heidelberg, Germany, 1999; pp. 39–59. [CrossRef]
9. Ben-Nakhi, A.E.; Mahmoud, M.A. Cooling load prediction for buildings using general regression neural networks. *Energy Convers. Manag.* **2004**, *45*, 2127–2141. [CrossRef]
10. Mihalakakou, G.; Santamouris, M.; Tsangrassoulis, A. On the energy consumption in residential buildings. *Energy Build.* **2002**, *34*, 727–736. [CrossRef]
11. Smith, D.M.; Cusack, S.; Colman, A.W.; Folland, C.K.; Harris, G.R.; Murphy, J.M. Improved surface temperature prediction for the coming decade from a global climate model. *Science* **2007**, *317*, 796–799. [CrossRef]
12. World Meteorological Organization. 2019. Available online: <https://public.wmo.int/en/our-mandate/what-we-do> (accessed on 1 February 2019).
13. Penland, C.; Magorian, T. Prediction of Nino 3 sea surface temperatures using linear inverse modeling. *J. Clim.* **1993**, *6*, 1067–1076. [CrossRef]
14. Penland, C.; Matrosova, L. Prediction of tropical Atlantic sea surface temperatures using linear inverse modeling. *J. Clim.* **1998**, *11*, 483–496. [CrossRef]
15. Johnson, S.D.; Battisti, D.S.; Sarachik, E. Empirically derived Markov models and prediction of tropical Pacific sea surface temperature anomalies. *J. Clim.* **2000**, *13*, 3–17. [CrossRef]
16. Newman, M. An empirical benchmark for decadal forecasts of global surface temperature anomalies. *J. Clim.* **2013**, *26*, 5260–5269. [CrossRef]
17. Figura, S.; Livingstone, D.M.; Kipfer, R. Forecasting groundwater temperature with linear regression models using historical data. *Groundwater* **2015**, *53*, 943–954. [CrossRef]
18. Bartos, I.; Jánosi, I. Nonlinear correlations of daily temperature records over land. *Nonlinear Process. Geophys.* **2006**, *13*, 571–576. [CrossRef]
19. Bonsal, B.; Zhang, X.; Vincent, L.; Hogg, W. Characteristics of Daily and Extreme Temperatures over Canada. *J. Clim.* **2001**, *14*, 1959–1976. [CrossRef]
20. Miyano, T.; Girosi, F. *Forecasting Global Temperature Variations by Neural Networks*; Technical Report; Massachusetts Institute of Technology, Cambridge Artificial Intelligence Laboratory: Cambridge, MA, USA, 1994.
21. Hippert, H.S.; Pedreira, C.E.; Souza, R.C. Combining neural networks and ARIMA models for hourly temperature forecast. In Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks, IJCNN 2000, Neural Computing: New Challenges and Perspectives for the New Millennium, Como, Italy, 27 July 2000.
22. Tasadduq, I.; Rehman, S.; Bubshait, K. Application of neural networks for the prediction of hourly mean surface temperatures in Saudi Arabia. *Renew. Energy* **2002**, *25*, 545–554. [CrossRef]
23. Lanza, P.A.G.; Cosme, J.M.Z. A short-term temperature forecaster based on a novel radial basis functions neural network. *Int. J. Neural Syst.* **2001**, *11*, 71–77. [CrossRef]
24. Maqsood, I.; Khan, M.R.; Abraham, A. An ensemble of neural networks for weather forecasting. *Neural Comput. Appl.* **2004**, *13*, 112–122. [CrossRef]
25. Smith, B.A.; McClendon, R.W.; Hoogenboom, G. Improving air temperature prediction with artificial neural networks. *Int. J. Comput. Intell.* **2006**, *3*, 179–186.
26. Smith, B.A.; Hoogenboom, G.; McClendon, R.W. Artificial neural networks for automated year-round temperature prediction. *Comput. Electron. Agric.* **2009**, *68*, 52–61. [CrossRef]
27. Jallal, M.A.; Chabaa, S.; El Yassini, A.; Zeroual, A.; Ibyaich, S. Air temperature forecasting using artificial neural networks with delayed exogenous input. In Proceedings of the 2019 International Conference on Wireless Technologies, Embedded and Intelligent Systems (WITS), Fez, Morocco, 3–4 April 2019.
28. Pal, N.R.; Pal, S.; Das, J.; Majumdar, K. SOFM-MLP: A hybrid neural network for atmospheric temperature prediction. *IEEE Trans. Geosci. Remote. Sens.* **2003**, *41*, 2783–2791. [CrossRef]
29. Maqsood, I.; Abraham, A. Weather analysis using ensemble of connectionist learning paradigms. *Appl. Soft Comput.* **2007**, *7*, 995–1004. [CrossRef]
30. Ustaoglu, B.; Cigizoglu, H.K.; Karaca, M. Forecast of daily mean, maximum and minimum temperature time series by three artificial neural network methods. *Meteorol. Appl.* **2008**, *15*, 431–445. [CrossRef]

31. Hayati, M.; Mohebi, Z. Application of artificial neural networks for temperature forecasting. *World Acad. Sci. Eng. Technol.* **2007**, *28*, 275–279.
32. Abhishek, K.; Singh, M.; Ghosh, S.; Anand, A. Weather Forecasting Model using Artificial Neural Network. *Procedia Technol.* **2012**, *4*, 311–318. [[CrossRef](#)]
33. Chevalier, R.F.; Hoogenboom, G.; McClendon, R.W.; Paz, J.A. Support vector regression with reduced training sets for air temperature prediction: A comparison with artificial neural networks. *Neural Comput. Appl.* **2010**, *20*, 151–159. [[CrossRef](#)]
34. Ortiz-García, E.; Salcedo-Sanz, S.; Casanova-Mateo, C.; Paniagua-Tineo, A.; Portilla-Figueras, J. Accurate local very short-term temperature prediction based on synoptic situation Support Vector Regression banks. *Atmos. Res.* **2012**, *107*, 1–8. [[CrossRef](#)]
35. Mellit, A.; Pavan, A.M.; Benghanem, M. Least squares support vector machine for short-term prediction of meteorological time series. *Theor. Appl. Climatol.* **2013**, *111*, 297–307. [[CrossRef](#)]
36. Mori, H.; Kanaoka, D. Application of support vector regression to temperature forecasting for short-term load forecasting. In Proceedings of the 2007 International Joint Conference on Neural Networks, Orlando, FL, USA, 12–17 August 2007.
37. Radhika, Y.; Shashi, M. Atmospheric Temperature Prediction using Support Vector Machines. *Int. J. Comput. Theory Eng.* **2009**, 55–58. [[CrossRef](#)]
38. Paniagua-Tineo, A.; Salcedo-Sanz, S.; Casanova-Mateo, C.; Ortiz-García, E.; Cony, M.; Hernández-Martín, E. Prediction of daily maximum temperature using a support vector regression algorithm. *Renew. Energy* **2011**, *36*, 3054–3060. [[CrossRef](#)]
39. Abubakar, A.; Chiroma, H.; Zeki, A.; Uddin, M. Utilising key climate element variability for the prediction of future climate change using a support vector machine model. *Int. J. Glob. Warm.* **2016**, *9*, 129–151. [[CrossRef](#)]
40. Hewage, P.; Trovati, M.; Pereira, E.; Behera, A. Deep learning-based effective fine-grained weather forecasting model. *Pattern Anal. Appl.* **2020**, 1–24. [[CrossRef](#)]
41. Roesch, I.; Günther, T. Visualization of Neural Network Predictions for Weather Forecasting. *Comput. Graph. Forum* **2018**, *38*, 209–220. [[CrossRef](#)]
42. Alpaydin, E. *Introduction to Machine Learning*; MIT Press: Cambridge, MA, USA, 2009.
43. Duveiller, G.; Fasbender, D.; Meroni, M. Revisiting the concept of a symmetric index of agreement for continuous datasets. *Sci. Rep.* **2016**, *6*, 19401.
44. Kalogirou, S.A. Artificial neural networks in renewable energy systems applications: A review. *Renew. Sustain. Energy Rev.* **2001**, *5*, 373–401. [[CrossRef](#)]
45. Haykin, S. *Neural Networks*; Prentice Hall: New York, NY, USA, 1994; Volume 2.
46. Haykin, S.S.; Haykin, S.S.; Haykin, S.S.; Elektroingenieur, K.; Haykin, S.S. *Neural Networks and Learning Machines*; Pearson Education: Bengaluru, India, 2009; Volume 3.
47. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [[CrossRef](#)]
48. Vapnik, V. *The Nature of Statistical Learning Theory*; Springer Science & Business Media: New York, NY, USA, 2013.
49. Cristianini, N.; Shawe-Taylor, J. *An Introduction to Support Vector Machines and other Kernel-Based Learning Methods*; Cambridge University Press: Cambridge, MA, USA, 2000.
50. Gunn, S.R.; Support vector machines for classification and regression. In *ISIS Technical Report*; University of Southampton: Southampton, UK, 1998; Volume 14, pp. 5–16.
51. Mellit, A. Artificial Intelligence technique for modelling and forecasting of solar radiation data: A review. *Int. J. Artif. Intell. Soft Comput.* **2008**, *1*, 52–76. [[CrossRef](#)]
52. Maier, H.R.; Dandy, G.C. Neural networks for the prediction and forecasting of water resources variables: A review of modelling issues and applications. *Environ. Model. Softw.* **2000**, *15*, 101–124. [[CrossRef](#)]
53. Argiriou, A. Use of neural networks for tropospheric ozone time series approximation and forecasting—A review. *Atmos. Chem. Phys. Discuss.* **2007**, *7*, 5739–5767. [[CrossRef](#)]
54. Wang, W.; Xu, Z.; Weizhen Lu, J. Three improved neural network models for air quality forecasting. *Eng. Comput.* **2003**, *20*, 192–210. [[CrossRef](#)]
55. Ko, C.N.; Lee, C.M. Short-term load forecasting using SVR (support vector regression)-based radial basis function neural network with dual extended Kalman filter. *Energy* **2013**, *49*, 413–422. [[CrossRef](#)]
56. Topcu, I.B.; Saridemir, M. Prediction of compressive strength of concrete containing fly ash using artificial neural networks and fuzzy logic. *Comput. Mater. Sci.* **2008**, *41*, 305–311. [[CrossRef](#)]

57. Hyndman, R.J.; Koehler, A.B. Another look at measures of forecast accuracy. *Int. J. Forecast.* **2006**, *22*, 679–688. [[CrossRef](#)]
58. Shcherbakov, M.; Brebels, A. Outliers and anomalies detection based on neural networks forecast procedure. In Proceedings of the 31st Annual International Symposium on Forecasting, ISF, Prague, Czech Republic, 26–29 June 2011.
59. Armstrong, J.; Collopy, F. Error measures for generalizing about forecasting methods: Empirical comparisons. *Int. J. Forecast.* **1992**, *8*, 69–80. [[CrossRef](#)]
60. Banhatti, A.G.; Deka, P.C. Effects of Data Pre-processing on the Prediction Accuracy of Artificial Neural Network Model in Hydrological Time Series. In *Urban Hydrology, Watershed Management and Socio-Economic Aspects*; Springer International Publishing: Cham, Switzerland, 2016; pp. 265–275. [[CrossRef](#)]
61. Chen, C.; Twycross, J.; Garibaldi, J.M. A new accuracy measure based on bounded relative error for time series forecasting. *PLoS ONE* **2017**, *12*, e0174202. [[CrossRef](#)]
62. Shcherbakov, M.V.; Brebels, A.; Shcherbakova, N.L.; Tyukov, A.P.; Janovsky, T.A.; Kamaev, V.A. A survey of forecast error measures. *World Appl. Sci. J.* **2013**, *24*, 171–176.
63. Solomon, S.; Qin, D.; Manning, M.; Averyt, K.; Marquis, M. *Climate Change 2007—The Physical Science Basis: Working Group I Contribution to the Fourth Assessment Report of the IPCC*; Cambridge University Press: Cambridge, MA, USA, 2007; Volume 4.
64. Lee, T.C.; Zwiers, F.W.; Zhang, X.; Tsao, M. Evidence of decadal climate prediction skill resulting from changes in anthropogenic forcing. *J. Clim.* **2006**, *19*, 5305–5318. [[CrossRef](#)]
65. Stott, P.A.; Kettleborough, J.A. Origins and estimates of uncertainty in predictions of twenty-first century temperature rise. *Nature* **2002**, *416*, 723. [[CrossRef](#)]
66. Knutti, R.; Stocker, T.; Joos, F.; Plattner, G.K. Probabilistic climate change projections using neural networks. *Clim. Dyn.* **2003**, *21*, 257–272. [[CrossRef](#)]
67. Pasini, A.; Lorè, M.; Ameli, F. Neural network modelling for the analysis of forcings/temperatures relationships at different scales in the climate system. *Ecol. Model.* **2006**, *191*, 58–67. [[CrossRef](#)]
68. Pasini, A.; Pelino, V.; Potestà, S. A neural network model for visibility nowcasting from surface observations: Results and sensitivity to physical input variables. *J. Geophys. Res. Atmos.* **2001**, *106*, 14951–14959. [[CrossRef](#)]
69. Fildes, R.; Kourentzes, N. Validation and forecasting accuracy in models of climate change. *Int. J. Forecast.* **2011**, *27*, 968–995. [[CrossRef](#)]
70. Hassani, H.; Silva, E.S.; Gupta, R.; Das, S. Predicting global temperature anomaly: A definitive investigation using an ensemble of twelve competing forecasting models. *Phys. A Stat. Mech. Its Appl.* **2018**, *509*, 121–139. [[CrossRef](#)]
71. Jones, P.; Wigley, T.; Wright, P. Global temperature variations between 1861 and 1984. *Nature* **1986**, *322*, 430–434. [[CrossRef](#)]
72. Jones, P.; New, M.; Parker, D.E.; Martin, S.; Rigor, I.G. Surface air temperature and its changes over the past 150 years. *Rev. Geophys.* **1999**, *37*, 173–199. [[CrossRef](#)]
73. University of East Anglia. Climatic Research Unit. 2019. Available online: <http://www.cru.uea.ac.uk/> (accessed on 1 March 2019).
74. GesDisc. Solar Irradiance Anomalies. 2019. Available online: <http://www.soda-pro.com/> (accessed on 1 March 2019).
75. GISS. Stratospheric Aerosol Optical Thickness. 2019. Available online: <https://data.giss.nasa.gov/modelforce/strataer/> (accessed on 1 March 2019).
76. NCEI. Ocean Carbon Data System (OCADS). 2019. Available online: <https://www.nodc.noaa.gov/ocads/> (accessed on 1 March 2019).
77. NASA. GISS Surface Temperature Analysis (GISTEMP v3). 2019. Available online: [https://data.giss.nasa.gov/gistemp/index\\_v3.html](https://data.giss.nasa.gov/gistemp/index_v3.html) (accessed on 1 March 2019).
78. Xuejie, G.; Zongci, Z.; Yihui, D.; Ronghui, H.; Giorgi, F. Climate change due to greenhouse effects in China as simulated by a regional climate model. *Adv. Atmos. Sci.* **2001**, *18*, 1224–1230. [[CrossRef](#)]
79. Hsu, C.W.; Chang, C.C.; Lin, C.J. A Practical Guide to Support Vector Classification. 2003. Available online: <https://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf> (accessed on 1 March 2019).
80. Hossain, M.; Rekabdar, B.; Louis, S.J.; Dascalu, S. Forecasting the weather of Nevada: A deep learning approach. In Proceedings of the 2015 International Joint Conference on Neural Networks (IJCNN), Killarney, Ireland, 12–17 July 2015. [[CrossRef](#)]

81. Pardo, A.; Meneu, V.; Valor, E. Temperature and seasonality influences on Spanish electricity load. *Energy Econ.* **2002**, *24*, 55–70. [[CrossRef](#)]
82. Samani, Z. Estimating solar radiation and evapotranspiration using minimum climatological data. *J. Irrig. Drain. Eng.* **2000**, *126*, 265–267. [[CrossRef](#)]
83. Afzali, M.; Afzali, A.; Zahedi, G. The Potential of Artificial Neural Network Technique in Daily and Monthly Ambient Air Temperature Prediction. *Int. J. Environ. Sci. Dev.* **2012**, *3*, 33–38. [[CrossRef](#)]
84. Husaini, N.A.; Ghazali, R.; Nawawi, N.M.; Ismail, L.H. Jordan Pi-Sigma Neural Network for Temperature Prediction. In *Communications in Computer and Information Science*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 547–558. [[CrossRef](#)]
85. Rastogi, A.; Srivastava, A.; Srivastava, V.; Pandey, A. Pattern analysis approach for prediction using Wavelet Neural Networks. In Proceedings of the 2011 Seventh International Conference on Natural Computation, Shanghai, China, 26–28 July 2011. [[CrossRef](#)]
86. Sharma, A.; Agarwal, S. Temperature Prediction using Wavelet Neural Network. *Res. J. Inf. Technol.* **2012**, *4*, 22–30. [[CrossRef](#)]
87. Wang, G.; Qiu, Y.F.; Li, H.X. Temperature Forecast Based on SVM Optimized by PSO Algorithm. In Proceedings of the 2010 International Conference on Intelligent Computing and Cognitive Informatics, Kuala Lumpur, Malaysia, 22–23 June 2010. [[CrossRef](#)]
88. Karevan, Z.; Mehrkanoon, S.; Suykens, J.A. Black-box modeling for temperature prediction in weather forecasting. In Proceedings of the 2015 International Joint Conference on Neural Networks (IJCNN), Killarney, Ireland, 12–17 July 2015. [[CrossRef](#)]
89. Karevan, Z.; Suykens, J.A.K. Spatio-temporal feature selection for black-box weather forecasting. In Proceedings of the 24th European Symposium on Artificial Neural Networks, ESANN 2016, Bruges, Belgium, 27–29 April 2016.
90. Ashrafi, K.; Shafiepour, M.; Ghasemi, L.; Araabi, B. Prediction of climate change induced temperature rise in regional scale using neural network. *Int. J. Environ. Res.* **2012**, *6*, 677–688.
91. Bilgili, M.; Sahin, B. Prediction of Long-term Monthly Temperature and Rainfall in Turkey. *Energy Sources Part A Recover. Util. Environ. Eff.* **2009**, *32*, 60–71. [[CrossRef](#)]
92. Kisi, O.; Shiri, J. Prediction of long-term monthly air temperature using geographical inputs. *Int. J. Climatol.* **2013**, *34*, 179–186. [[CrossRef](#)]
93. De, S.; Debnath, A. Artificial neural network based prediction of maximum and minimum temperature in the summer monsoon months over India. *Appl. Phys. Res.* **2009**, *1*, 37. [[CrossRef](#)]
94. Liu, X.; Yuan, S.; Li, L. Prediction of Temperature Time Series Based on Wavelet Transform and Support Vector Machine. *J. Comput.* **2012**, *7*. [[CrossRef](#)]
95. Salcedo-Sanz, S.; Deo, R.C.; Carro-Calvo, L.; Saavedra-Moreno, B. Monthly prediction of air temperature in Australia and New Zealand with machine learning algorithms. *Theor. Appl. Climatol.* **2015**, *125*, 13–25. [[CrossRef](#)]
96. Papacharalampous, G.; Tyrallis, H.; Koutsoyiannis, D. Univariate Time Series Forecasting of Temperature and Precipitation with a Focus on Machine Learning Algorithms: A Multiple-Case Study from Greece. *Water Resour. Manag.* **2018**, *32*, 5207–5239. [[CrossRef](#)]
97. Gamboa, J.C.B. Deep learning for time-series analysis. *arXiv* **2017**, arXiv:1701.01887.

