sustainability

MDPI

Article

Hybrid Statistical and Machine Learning Methods for Daily Evapotranspiration Modeling

Erdem Küçüktopcu [1,*], Emirhan Cemek [2], Bilal Cemek [1] and Halis Simsek [3]

1 Department of Agricultural Structures and Irrigation, Ondokuz Mayıs University, 55139 Samsun, Türkiye
2 Hydraulics and Water Resources Engineering Program, Department of Civil Engineering, Istanbul Technical University, 34469 Istanbul, Türkiye
3 Department of Agricultural and Biological Engineering, Purdue University, West Lafayette, IN 47907, USA
* Correspondence: erdem.kucuktopcu@omu.edu.tr

Abstract: Machine learning (ML) models, including artificial neural networks (ANN), generalized neural regression networks (GRNN), and adaptive neuro-fuzzy interface systems (ANFIS), have received considerable attention for their ability to provide accurate predictions in various problem domains. However, these models may produce inconsistent results when solving linear problems. To overcome this limitation, this paper proposes hybridizations of ML and autoregressive integrated moving average (ARIMA) models to provide a more accurate and general forecasting model for evapotranspiration (ET_0). The proposed models are developed and tested using daily ET_0 data collected over 11 years (2010–2020) in the Samsun province of Türkiye. The results show that the ARIMA–GRNN model reduces the root mean square error by 48.38%, the ARIMA–ANFIS model by 8.56%, and the ARIMA–ANN model by 6.74% compared to the traditional ARIMA model. Consequently, the integration of ML with ARIMA models can offer more accurate and dependable prediction of daily ET_0, which can be beneficial for many branches such as agriculture and water management that require dependable ET_0 estimations.

Keywords: Box–Jenkins; time series modeling; evapotranspiration; artificial intelligence

check for updates

Citation: Küçüktopcu, E.; Cemek, E.; Cemek, B.; Simsek, H. Hybrid Statistical and Machine Learning Methods for Daily Evapotranspiration Modeling. *Sustainability* **2023**, *15*, 5689. https://doi.org/10.3390/su15075689

Academic Editor: Sushant Mehan

Received: 1 March 2023
Revised: 20 March 2023
Accepted: 23 March 2023
Published: 24 March 2023

1. Introduction

Efficient irrigation management is a critical aspect of modern agricultural techniques, and accurately estimating evapotranspiration (ET) is essential for effective water resource management, irrigation scheduling, watershed management, and drainage system planning [1]. ET is a method used to measure the water requirements of a crop, which comprises the movement of water vapor from the soil into the air through evaporation from the soil and transpiration from the plants [2]. The first step in determining the ET of an agricultural system is to calculate the reference evapotranspiration (ET_0), which is a widely accepted method for quantifying the water requirements of a crop. However, estimating ET_0 is a complex task that can be conducted either by direct measurement with lysimeters or indirectly through mathematical models.

Typically, lysimeters that provide accurate measurements are used to develop and evaluate other indirect methods [3–5]. However, this method is often deemed impractical due to its time-consuming and precise measurement requirements. Moreover, the high cost and complexity associated with lysimeters typically limit their application to research settings. Therefore, for practical purposes, mathematical models based on weather station data have become the preferred alternative [6].

In indirect methods, the equations used to calculate ET_0 values are usually complex, nonlinear, contain random factors, and rely on several assumptions. The literature reports about 20 methods that can be used to estimate ET_0 based on meteorological variables. Of these methods, FAO Penman–Monteith (FAO56PM) [2], Thornthwaite [7], Blaney and Criddle [8], Priestly and Taylor [9], and Hargreaves and Samani [10] have been widely and successfully applied. However, accurate estimates obtained by these methods require the application of rigorous optimization procedures, accurate spatiotemporal data, and a thorough understanding of initial conditions [11].

In recent years, the development of artificial intelligence methods based on machine learning (ML) has gained worldwide attention as an alternative approach to estimate ET_0 using a minimal number of weather input parameters [12]. Numerous research papers have found that ML techniques such as artificial neural networks (ANN), generalized neural regression networks (GRNN), and adaptive neuro-fuzzy interface systems (ANFIS) perform better in ET_0 estimation than empirical and semi-empirical methods [13–23].For instance, Kumar et al. [13] compared the performance of ANN and the Penman–Monteith method in predicting ET_0 and concluded that ANN produced better estimates. Similarly, Ladlani et al. [21] found that the GRNN model outperformed the Priestley–Taylor and Hargreaves–Samani models in ET_0 estimation. Pour et al. [22] used ANFIS and ANN to estimate daily ET_0 with various combinations of independent variables consistent with the Hargreaves–Samani, Priestley–Taylor, and FAO56-PM equations. The results indicated that both ANFIS and ANN provided satisfactory performance in estimating ET_0 from available climate data.

One strategy for achieving performance improvements through ML techniques in estimating ET_0 is to use previous days' meteorological data as input to the models in addition to the current day's data. The autoregressive integrated moving average model (ARIMA), commonly known as the Box–Jenkins methodology, is a stochastic model with a strong and reliable performance in predicting a wide range of climatic, hydrologic, and meteorological variables [24–26]. However, a single model for ET_0 prediction using time series data is prone to significant errors. Therefore, hybrid approaches that combine multiple models to improve accuracy could be an alternative solution. Arca et al. [27] reported the superiority of the ARIMA model over the ANN model in estimating daily ET_0 in Italy. Kishore et al. [28] employed ANN and ARIMA models to predict ET_0 in Kanchipuram, India, and noticed that the ARIMA model was very effective and reliable for short-term forecasts. Landeras et al. [29] found that applying ARIMA and ANN models improved the performance of predicting weekly ET_0 one week ahead compared to the model based on annual averages.

These studies have contributed significantly to the knowledge base on using ML and ARIMA models in estimating ET_0. However, the applications of combined ML techniques to reduce the errors of the ARIMA model for ET_0 estimation problems are currently limited, and the knowledge on this topic is still incomplete and fragmented. To fill this gap, this study aims to present a new hybrid model that combines the advantages of ARIMA and ML techniques to achieve accurate daily ET_0 prediction. The main contributions of this study are summarized as follows:

- A time series ARIMA model and various ML techniques, including ANN, GRNN, and ANFIS, are built to predict the daily ET_0 of Samsun, Türkiye, based on climate parameters.

The models' accuracy, applicability, and reliability are compared to the ASCE Penman–Monteith method.

- A novel approach combining ARIMA and ML models is developed for ET_0 predictions.

The remainder of this paper is structured as follows. Section 2 details the ARIMA and ML models used for ET_0 estimates. Section 3 provides details on the methodology employed, including the study area, dataset, model selection, and performance criteria for the models. Section 4 compares the predictive performance of the models and presents the numerical results. Section 5 discusses the study's findings and provides information on the limitations of this study. Finally, Section 6 summarizes the conclusions and offers proposals for future work.

2. Background

2.1. Linear Process: Box–Jenkins ARIMA Model

The Box–Jenkins method is a statistical forecasting technique used to predict and control univariate time series. It was developed assuming that time-dependent events are random and the associated time series are stochastic processes. This method applies to discrete and stationary time series of uniformly distributed observed values [30].

The ARIMA model with linear stochastic structure is commonly used in many research fields for predicting desired data. An ARIMA (p, d, q) model is a combination of autoregression AR (p) (an additive linear function of p past observations), an integer I (d) (d makes a series stationary), and moving average MA (q) (q random error). The partial autocorrelation function (PACF) and autocorrelation function (ACF) of the data are used to determine the order of q and p of the ARIMA model. The model for each environmental factor can be described mathematically in Equation (1).

$$\begin{aligned} X_t^i &= \alpha_t^i + \varphi_1^i X_{t-1}^i + \varphi_2^i X_{t-2}^i + \cdots + \varphi_p^i X_{t-p}^i \\ &\quad - \varepsilon_t^i - \theta_1^i \varepsilon_{t-1}^i - \theta_2^i \varepsilon_{t-2}^i - \cdots - \theta_q^i \varepsilon_{t-q}^i \end{aligned} \quad (1)$$

where X_t^i is the *ith* influencing factor resulting from the differentiation of d times; φ_1^i, φ_2^i, ..., φ_p^i and θ_1^i, θ_2^i, ..., θ_p^i are autoregressive; and moving average coefficients have to be calculated for the *ith* factor. It is assumed that ε_t^i, ε_{t-1}^i, ε_{t-2}^i, ..., ε_{t-p}^i have a mean of zero and a constant variance.

The fitting of an ARIMA model is conducted in 4 steps as follows:

- Identification of the ARIMA (p, d, q) model structure;
- Estimate parameters for ARIMA (p, d, q) model;
- Check residuals to determine model adequacy;
- Predict future data from existing data.

2.2. Artificial Neural Networks (ANN)

The Multilayer Perceptron (MLP) is a popular type of ANN that is increasingly being used to handle applications with imprecise data or complex attribute relations. The MLP comprises an input layer, one or more hidden layers, and an output layer, with each neuron having continuous inputs and outputs, a sum input function, and a nonlinear activation function. The number of hidden nodes needs to be determined to optimize the MLP and obtain the best performance. This has been studied in numerous research papers, and it is generally accepted that one hidden layer is sufficient for approximating complex nonlinear functions [31,32].

The MSE function (Equation (2)), a popular measure of error applied to ANN and ML algorithms, was utilized to evaluate the quantitative error of the network. The equation calculates the average difference between the actual outputs (x) and the expected outputs (y) over n inputs.

$$MSE = \frac{1}{n}\sum_{j=1}^{n}\left(x_j - y_j\right)^2 \quad (2)$$

The stop criteria were set at $MSE = 1 \times 10^{-6}$ and 5000 epochs. Comprehensive information on ANN can be found in Refs. [33–35].

2.3. Generalized Regression Neural Networks (GRNN)

The GRNN proposed by Specht [36] does not require an iterative training process such as the backpropagation technique. Instead, it computes the function between the input and output vectors from the data itself to obtain a more accurate approximation. As the amount of training data increases, the estimation error decreases with minimal constraints on the function. Similar to other regression techniques, GRNN can be used to determine continuous variables and is based on a basic statistical method called kernel regression.

A GRNN is a powerful tool that consists of four layers: input, pattern, summation, and output. In the input layer, data are taken in by a number of observed parameters equal to the number of input units. The pattern layer contains training patterns, while the summation layer comprises two types of neurons: single-division and summation neurons. The single-division neurons are connected to the pattern layer, and the summation neurons are connected to the output layer. The hidden and output layers use Radial basis and linear activation functions, respectively. Lastly, the output layer normalizes the output set by dividing the output of each S-summation neuron by the output of each D-summation neuron, producing the predicted value Y_i for the unknown input vector x [37] (Equations (3) and (4)):

$$Y_i = \frac{\sum_{i=1}^{n} y_i . \exp[-D(x, x_i)]}{\sum_{i=1}^{n} \exp[-D(x, x_i)]} \tag{3}$$

$$D(x, x_i) = \sum_{k=1}^{m} \left(\frac{x_i - x_{ik}}{\sigma} \right)^2 \tag{4}$$

where y_i represents the weighted connection between the *ith* pattern layer neuron and the S-summation neuron; n is the training pattern numbers; D is the Gaussian function; m is the number of input vector elements; and x_k and x_{ik} are the *jth* elements of x and x_i, respectively. The optimal value of the spread parameter, denoted by σ, is determined experimentally.

2.4. Adaptive Neuro-Fuzzy Inference System (ANFIS)

ANFIS is a powerful tool that combines the learning capabilities of neural networks with the decision-making process of fuzzy logic [38,39]. It constructs an optimized neural network based on a training dataset suitable for the task. The accuracy and applicability of the model to the task can then be evaluated by testing on unseen data. ANFIS has an advantage over other ANN, as it is more interpretable thanks to the inclusion of fuzzy logic. The fuzzy logic can justify the weighting values set for the model, which is a major shortcoming of ANN. The ANFIS architecture consists of five layers: fuzzification, rule, normalization, defuzzification, and summation. The fuzzification layer assigns membership functions to the input values to form fuzzy clusters, which are determined by a set of parameters called premise parameters. The degrees of membership are then calculated based on these parameters, as shown in Equations (5) and (6).

$$O_i^1 = \mu_{A_i}(x) = \frac{1}{1 + \left| \frac{x - c_i}{a_i} \right|^{2b_i}} \tag{5}$$

$$O_i^1 = \mu_{B_i}(y) = \frac{1}{1 + \left| \frac{y - c_i}{a_i} \right|^{2b_i}} \tag{6}$$

where x and y are the inputs to node i; A is a linguistic label associated with this node function; O_i is the membership function of A_i; and a_i, b_i, and c_i are parameters of the membership function.

The fuzzification layer takes the input data and assigns a membership value to each fuzzy set. They are converted into firing strengths (w_i) for the rules. The membership values can be calculated using Equation (7):

$$O_i^2 = w_i = \mu_{A_i}(x) . \mu_{B_i}(y) \tag{7}$$

The normalization layer derives a normalized value of firing strengths for each rule. The normalized value results from the ratio of the firing strength of the *ith* rule to the sum of all firing strengths (Equation (8)):

$$O_i^3 = w_{i,avg} = \frac{w_i}{w_1 + w_2 + w_3 + w_4}, \quad i \in \{1, 2, 3, 4\} \tag{8}$$

The weights of the rules determined by using a first-order polynomial in each node of the defuzzification layer are calculated as shown in Equation (9).

$$O_i^4 = w_{i,avg} f_i = w_{i,avg}(p_i x + q_i y + r_i) \tag{9}$$

where $w_{i,avg}$ is the output of the normalization layer and p, q, and r are the parameter sets. These parameters are referred to as consequence parameters. Each rule has one more consequence parameter than the number input.

The summation of the outputs of the defuzzification layer gives the actual output of the ANFIS (Equation (10)).

$$O_i^5 = \sum_i w_{i,avg} f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} \tag{10}$$

3. Methodology

The hybrid method for predicting daily ET_0 values is a four-step process (Figure 1).

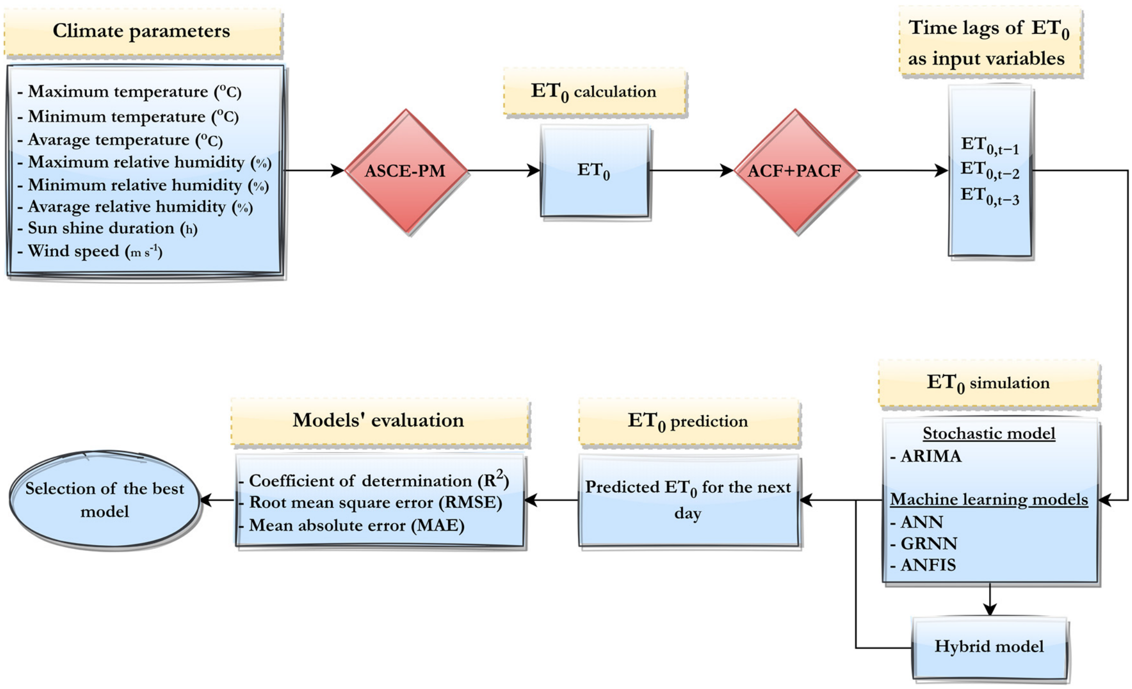


Figure 1. Flowchart for estimating daily ET_0 using hybrid stochastic and machine learning models.

Step 1: The ASCE Penman–Monteith method was used to calculate daily ET_0 values by utilizing meteorological data from the Atakum weather station;

Step 2: The ARIMA, ANN, GRNN, and ANFIS models used $ET_{0(t-1)}$, $ET_{0(t-2)}$, and $ET_{0(t-3)}$ as inputs to predict daily ET_0;

Step 3: The hybrid models, including ARIMA–ANN, ARIMA–GRNN, and ARIMA–ANFIS, used the error values of the periods $t-1$, $t-2$, and $t-3$ as input parameters to predict daily ET_0;

Step 4: Statistical and graphical methods were employed to evaluate the predictive accuracy of all models for ET_0 values.

3.1. Study Area and Dataset

Daily meteorological measurement data for a 10-year period (from January 2010 to January 2020) were obtained from the Atakum Weather Station (41°34′ north latitude, 36°25′ east longitude, and 4 m altitude), operated by the General Directorate of Meteorology of Samsun province in Türkiye. The city of Samsun is situated between the deltas of the

Yeşilırmak and Kızılırmak rivers, covering an area of 9083 km^2. The climate in Samsun is generally temperate, with distinct characteristics on the coast and in the interior regions. The effects of the Black Sea climate are felt on the coast, leading to hot summers and rainy winters. Inland, the city is influenced by the Akdağ and Canik Mountains, with cold, rainy, and snowy winters and cool summers [40]. The average annual temperature in Samsun is 14.6 °C, which is higher than the Turkish average of 13.6 °C. The average minimum temperature is 3.9 °C in February, while the average maximum temperature is 27.1 °C in August. The ASCE Penman–Monteith method was used to calculate daily ET_0 using meteorological data from the Atakum weather station.

3.2. Data Pre-Processing

The process of data preprocessing is critical to achieve higher performance and accuracy for ML models. It includes noisy, missing, and inconsistent data. In this study, daily weather data (temperature, relative humidity, wind speed, and sunshine duration) were merged as inputs with calculated ET_0 data. Further preprocessing of the data included data cleaning, transformation, and splitting. Standardization of the data to a range of 0 and 1 was implemented to overcome the challenges associated with processing data with different units of measurement and to avoid complications associated with extreme values. To evaluate the predictive performance of the different models, each dataset was divided into two subsets: a training set and a testing set. The training set, comprising 70% of the data, was used exclusively for model development. After model development, the testing set, comprising 30% of the data, was used to evaluate the proposed models.

3.3. Selection of Component Models

A number of linear statistical models have been proposed in the literature for time series models, which are generally divided into two categories: linear and nonlinear models. ARIMA models are one of the most popular linear time series models widely used in hydrology and meteorology [41–45]. ARIMA is popular for its statistical properties and the well-known Box–Jenkins method [30]. By using a differencing process, the ARIMA model can effectively convert non-stationary data into stationary data, so it can be used to fit non-stationary time series. Therefore, in this paper, the ARIMA model is chosen for developing the proposed hybrid approach. ANN, GRNN, and ANFIS are employed to identify the nonlinear component of the time series, based on their successful applications in modeling time series data [46–49].

3.4. Performance Criteria of Model

To accurately assess the performance of the models in this study, three commonly used metrics were employed: the coefficient of determination (R^2), root mean square error ($RMSE$), and mean absolute error (MAE). These equations, outlined in Equations (11)–(13) by Waller [50], provide a comprehensive and quantifiable assessment of the models' performances.

$$R^2 = 1 - \frac{\sum_{i=1}^{n} \left(Z_{i,m} - Z_{i,p}\right)^2}{\sum_{i=1}^{n} \left(Z_{i,m} - Z_{i,avg}\right)^2} \tag{11}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} \left(Z_{i,m} - Z_{i,p}\right)^2}{n}} \tag{12}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n} \left|Z_{i,m} - Z_{i,p}\right| \tag{13}$$

where $Z_{i,m}$ is the measured value; $Z_{i,p}$ is the predicted value; $Z_{i,avg}$ is the average value measured; and n is the number of data.

Additionally, a Taylor diagram was used to analyze the standard deviation (SD) and correlation coefficients (R) between the predicted and observed values. This analysis provided further insight into how closely the model predictions matched the observed data.

4. Results

In this paper, ET_0 estimation was performed in two stages. In the first stage, $ET_{0(t-1)}$, $ET_{0(t-2)}$, and $ET_{0(t-3)}$ were taken as input parameters for predicting daily ET_0 using the ARIMA, ANN, GRNN, and ANFIS models. In the second stage of the study, the error values of the periods $t-1$, $t-2$, and $t-3$ were used as input parameters. The difference between the results of the ARIMA model and the observed ET_0 data served as the output value. Following that, the ARIMA combination value was calculated with re-estimated errors, considering the discrepancies between the errors estimated by ANN, GRNN, and ANFIS, the initial ARIMA, and the actual ET_0 values.

The implementation of the ARIMA model is based on a time series and should be stationary. It is influenced by climate, soil, and crop conditions that affect the daily ET_0 value. The stationarity of the daily ET_0 was studied considering the factors affecting crop water use. The distributed lag model is a time series model in which the effect of a single predictor, such as the lag of the predictor itself, on the response variable can vary over time. The lag length, or the time period that best explains the effects of the predictor and its lagged values on the response variable, can be determined by examining the model's fit to the data and measuring the model's predictive accuracy. The lag length can be determined by examining the model's shape and computing the Akaike Information Criterion (AIC). In this study, the quadratic polynomial equation as a function of lag length was used.

The Box–Jenkins technique was employed to analyze the autocorrelation function (ACF) and partial autocorrelation function (PACF) of ET_0 values for the ARIMA model. The ACF and PACF plots indicated that the time series of ET_0 was stationary (Figure 2a,b). Therefore, the ARIMA (1, 0, 1) model was implemented for the ET_0 estimation. The coefficients of AR1 and MA1 parameters were statistically significant ($p < 0.01$) based on the results of Maximum Likelihood Estimation (MLE), as shown in Table 1.

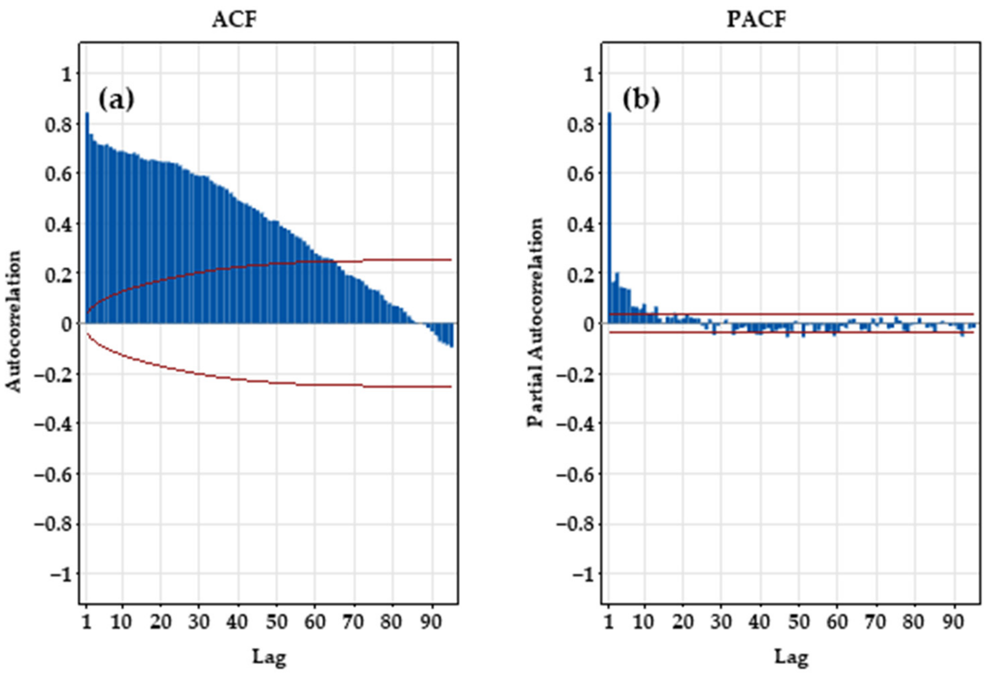


Figure 2. Plots of (**a**) ACF and (**b**) PACF of model ARIMA (1, 0, 1) for ET_0 time series.

Table 1. AR1 and MA1 coefficients of final ARIMA equation.

Parameters	Coefficient	Standard Error	t-Statistic	*p*-Value
Constant	0.12072	0.00820	14.72	0.000
AR1	0.95476	0.00671	142.37	0.000
MA1	0.46610	0.01990	23.41	0.000

In the ANN technique, a feedforward backpropagation MLP with a single hidden layer was utilized, and the Levenberg–Marquardt training algorithm was used to train the network. The hidden and output layers were constructed using the tangent sigmoid (tansig) and linear transfer (purelin) functions, respectively. The number of hidden nodes was optimized by incrementally increasing it from three to seven, and the best-performing model was selected based on the lowest MSE value in the testing phase. The ANN (3, 5, 1) model provided the best accuracy during the test period compared to the other ANN models based on the R^2, RMSE, and MAE criteria. In this model, three represents the input variables, namely, $ET_{0(t-1)}$, $ET_{0(t-2)}$, and $ET_{0(t-3)}$), five is the number of hidden layer neurons, and one is the output variable (ET_0).

In the GRNN technique, various spread factors (from 0.1 to 1) were tested to find the optimal model using a trial-and-error approach. The GRNN (3, 0.1) model with inputs of $ET_{0(t-1)}$, $ET_{0(t-2)}$, and $ET_{0(t-3)}$ and a spread value of 0.1 demonstrated higher accuracy than other GRNN models during the testing phase.

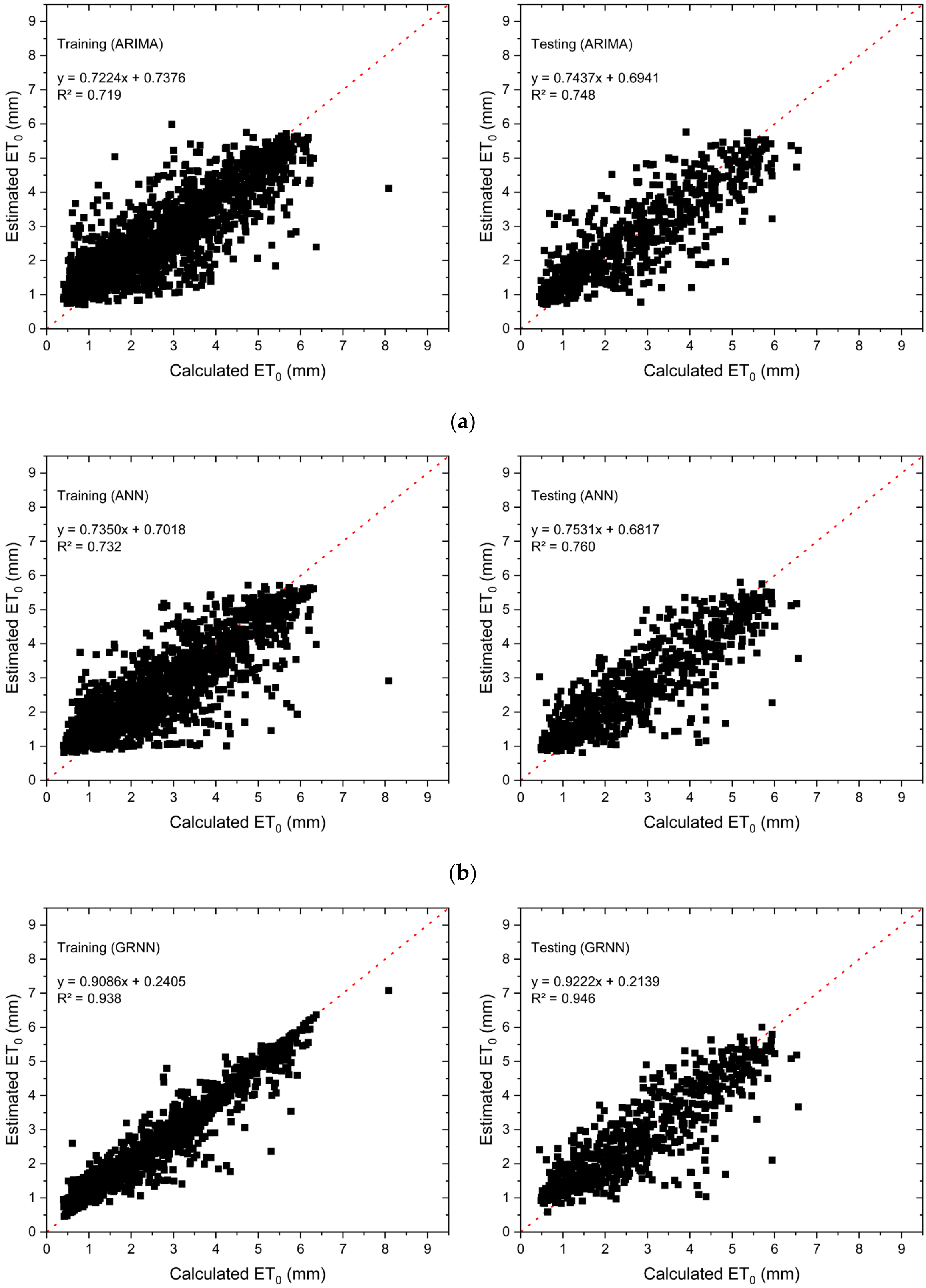
To evaluate the performance of the ANFIS technique, different types of membership functions (MFs), such as Gaussian, trapezoidal, and triangular, were tested with varying numbers of MFs. The ANFIS model with three triangular MFs for the inputs $ET_{0(t-1)}$, $ET_{0(t-2)}$, and $ET_{0(t-3)}$ was found to be the most effective among all the ANFIS models during the testing phase.

Table 2 summarizes the training and testing performance of the best ARIMA, ANN, GRNN, and ANFIS models. The results indicate that the GRNN model achieved the highest accuracy during the test phase (R^2 = 0.946, *RMSE* = 0.697 mm, and *MAE* = 0.527 mm) compared to the other models, followed by ANFIS, ANN, and ARIMA. Figure 3 shows the scatter plots of these models for the observed and predicted daily ET_0 values in both training and testing periods.

Table 2. Comparison of ARIMA, ANN, GRNN, and ANFIS for daily ET_0 prediction using training and testing datasets.

Model	Model Structure	Training			Testing		
		MAE	*RMSE*	R^2	*MAE*	*RMSE*	R^2
ARIMA	(1, 0, 1)	0.562	0.778	0.719	0.580	0.771	0.748
ANN	(3, 5, 1)	0.548	0.759	0.732	0.563	0.755	0.760
GRNN	(3, 0.1)	0.511	0.702	0.938	0.527	0.697	0.946
ANFIS	(3, trimf)	0.531	0.736	0.749	0.558	0.762	0.774

The second stage of the study aimed to construct a combined model, which utilizes a residual from the ARIMA model as input to a new ML network. This is due to the nonlinear aspect of the ARIMA model, which allows for more complex features to be incorporated into the model. The training and testing results of the models obtained with the hybrid approach are shown in Table 3. As seen from the table, the ARIMA–GRNN hybrid model achieved the lowest RMSE (0.400 mm) and MAE (0.270 mm) values in the testing stage, demonstrating its efficiency in predicting ET_0 values. Moreover, hybrid approaches were the most accurate overall, as evidenced by the statistical indices. The scatter plots of these hybrid models for the observed and predicted daily ET_0 values in the training and testing periods are displayed in Figure 4.

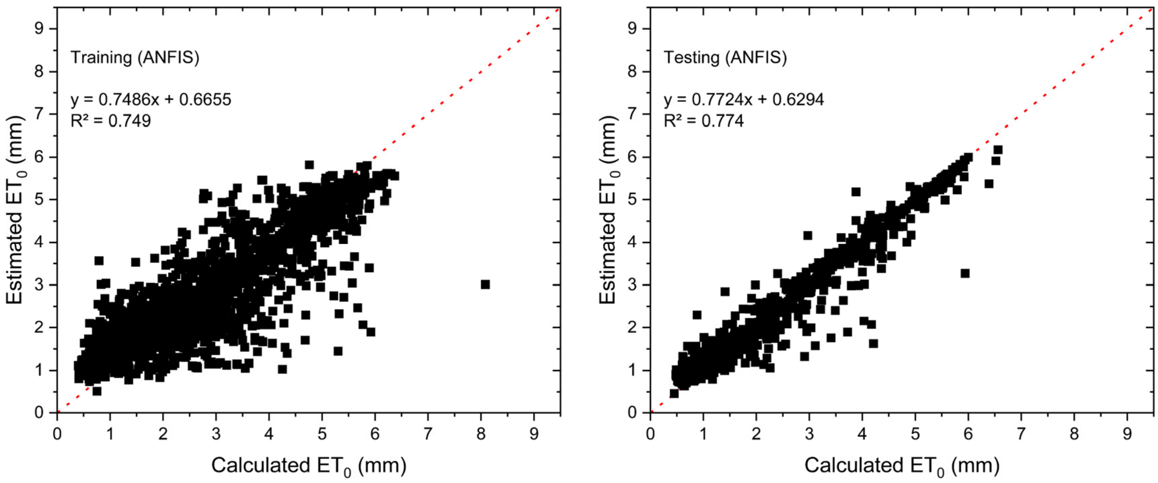


(**a**)

(**b**)

(**c**)

Figure 3. *Cont*.

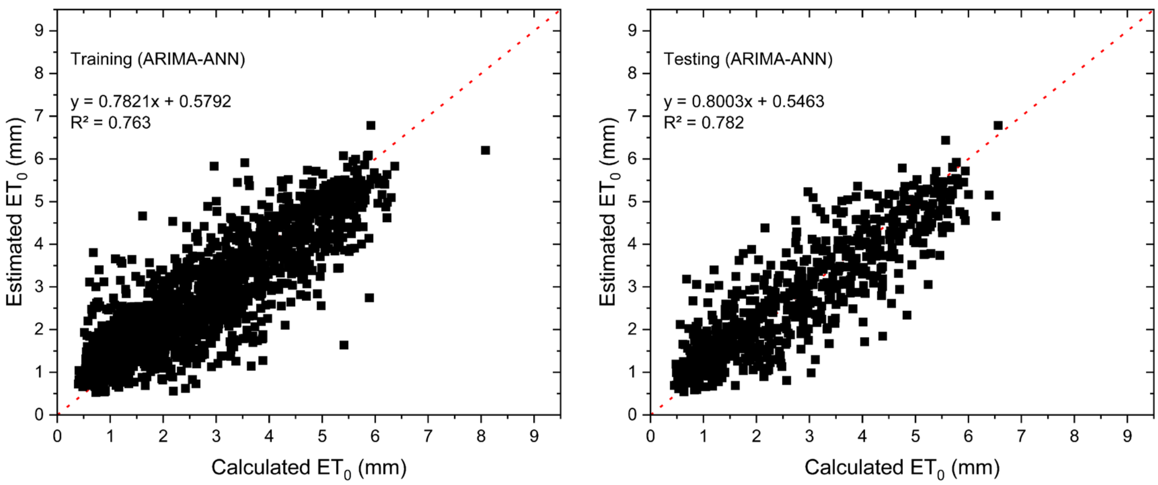


(**d**)

Figure 3. Comparison of calculated and estimated ET_0 values using (**a**) ARIMA, (**b**) ANN, (**c**) GRNN, and (**d**) ANFIS for training (n = 1826) and testing (n = 727) datasets.

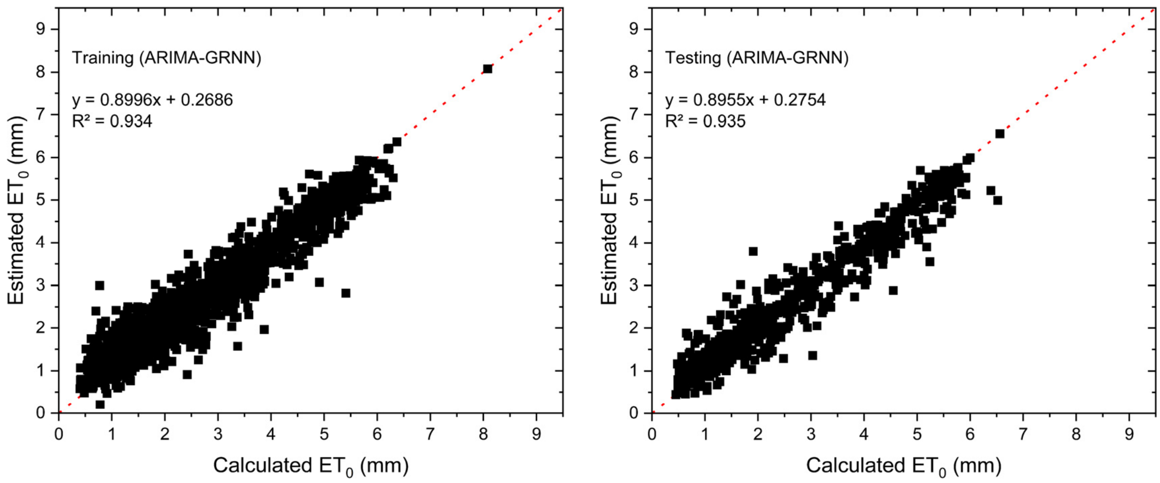
Table 3. Comparison of combined approaches (ARIMA–ANN, ARIMA–GRNN, and ARIMA–ANFIS) for daily ET_0 prediction using training and testing datasets.

Model	Model Structure	Training			Testing		
		MAE	*RMSE*	R^2	*MAE*	*RMSE*	R^2
ARIMA–ANN	(3, 5, 1)	0.521	0.715	0.763	0.541	0.719	0.782
ARIMA–GRNN	(3, 0.1)	0.254	0.381	0.934	0.269	0.398	0.935
ARIMA–ANFIS	(3, trimf)	0.508	0.700	0.773	0.534	0.705	0.791

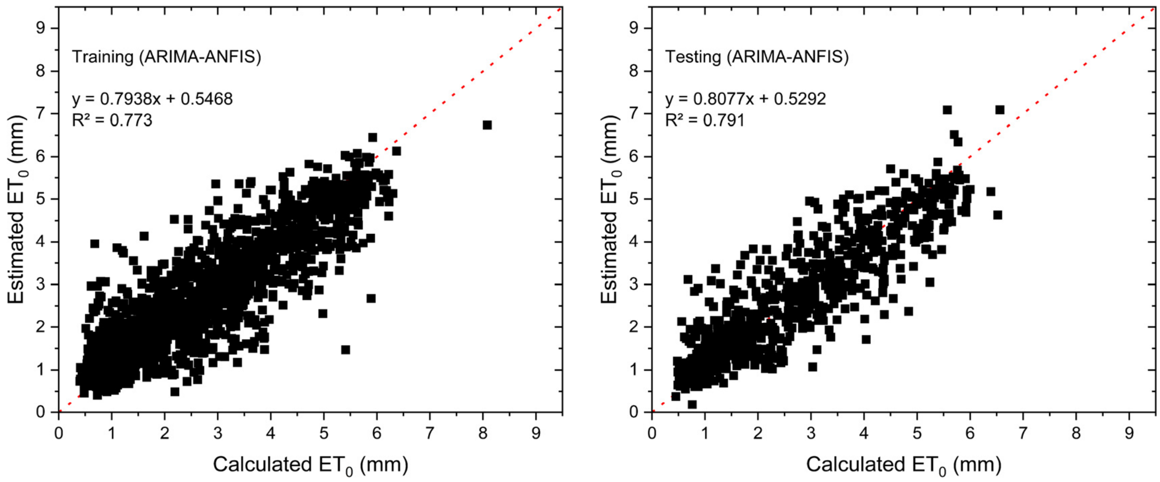


(**a**)

Figure 4. *Cont*.



(**b**)



(**c**)

Figure 4. Comparison of calculated and estimated ET_0 values using (**a**) ARIMA–ANN, (**b**) ARIMA–GRNN, and (**c**) ARIMA–ANFIS for training (*n* = 1826) and testing (*n* = 727) datasets.

The hybrid models (ARIMA–GRNN, ARIMA–ANFIS, and ARIMA–ANN) were able to outperform both the traditional ML models (GRNN, ANFIS, and ANN) and the traditional ARIMA model by significantly reducing RMSE values. Specifically, the ARIMA–GRNN model reduced the RMSE by 48.38%, the ARIMA–ANFIS model by 8.56%, and the ARIMA–ANN model by 6.74%, compared with the traditional ARIMA model. Similarly, the ARIMA–GRNN model reduced RMSE by 42.90%, the ARIMA–ANFIS model by 7.48%, and the ARIMA–ANN model by 4.77% compared with the non-ARIMA models.

The Taylor diagram, presented in Figure 5, provides an evaluation of the different model's performance in terms of bias, consistency, and scatter. The black point on the diagram represents the observed ET_0 value, while the other points correspond to the models' predictions. The ARIMA (R = 0.87 and SD = 1.32 mm) and ANN (R = 0.87 and SD = 1.33 mm) models are significantly far apart, indicating their poor performance in estimating ET_0 values. On the other hand, the ANFIS (R = 0.88 and SD = 1.35 mm), ARIMA–ANFIS (R = 0.89 and SD = 1.40 mm), and ARIMA–ANN (R = 0.88 and SD = 1.39 mm) models are extremely close, with no noticeable difference between them. Lastly, the GRNN (R = 0.97 and SD = 1.46 mm) and ARIMA–GRNN (R = 0.97 and SD = 1.97 mm) models are the closest to the calculated ET_0 results.

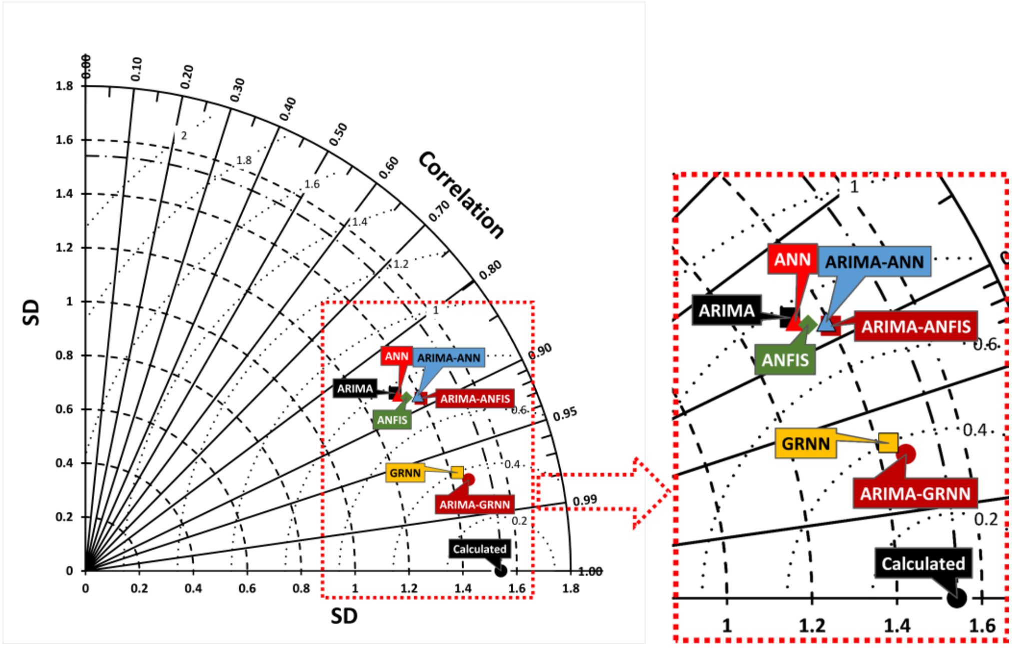


Figure 5. Taylor diagram of the predicted ET_0 values in the testing period. The arc represents the correlation (*R*) value, and the radial axis shows the standard deviation (SD) of the predicted data.

5. Discussion

Both time series and ML models have been independently evaluated for their ability to predict ET_0 values. Time series models are useful because they account for trends in the data and can be used to predict future values. ML models have also been shown to be effective in predicting ET_0 values because they can better identify more complex patterns in the data [15].

To achieve accurate prediction of ET_0 values, a combination of time series and ML models can be used. This approach needs to be customized to the specific dataset to yield the best outcomes. Firstly, the time series model should be employed to identify any existing trends in the data. These trends are then used as the basis for the ML model. The ML model should then refine the predictions further by taking into consideration any other factors that could affect the outcome.

This paper proposed hybridizations of ML and ARIMA models to create a more general and accurate ET_0 forecasting model. Among the models used in this study, the GRNN and ARIMA–GRNN models had lower *RMSE* and *MAE* values, making them slightly better than the others. Previous studies have independently evaluated the performance of time series [51–53], ML techniques [54,55], and hybrid models [56,57] for predicting ET_0 rates. For example, Ferreira et al. [58] utilized an ANN model to estimate daily ET_0, yielding *RMSE* of 0.70 mm and R^2 of 0.82. This study produced similar results, with *RMSE* of 0.755 mm and R^2 of 0.760. Antonopoulos and Antonopoulos [59] found that ANN models could estimate daily ET_0 with an accuracy ranging from an RMSE of 0.574 to 1.33 mm. In the study by Pour-Ali Baba et al. [22], the ANFIS models showed good performance, with RMSE values ranging from 0.474 to 0.851 mm. Dou and Yang [60] suggested that the ANFIS method is an valuable complement to traditional methods for estimating ET due to its robustness and flexibility. Landeras et al. [29] found that ARIMA and ANN models were more accurate than a mean year model based on historical averages, with a significant improvement in accuracy of 6–8% in reducing the mean squared differences of forecasts. These findings are consistent with the results of the present study. Arca et al. [27] reported the superiority of the ARIMA model in predicting daily ET_0 rates over the ANN model, while e Lucas et al. [60] found no significant difference between ARIMA and ML models in predicting ET_0 rates.

Although this study has yielded significant findings, there are some limitations that should be taken into consideration. For instance, the study only used data from a single location (Samsun province of Türkiye) and for a limited period (2010–2020). Additionally, the study only considered three ML methods (ANN, GRNN, and ANFIS) and ARIMA models, and other variables may also play an important role. In future studies, Deep Learning models with hybrid techniques need to be explored to improve the accuracy of ET_0 prediction further.

6. Conclusions

A new combined ARIMA soft computing modeling approach is proposed for estimating the daily water consumption of plants. Time series data were used to estimate ET_0 in Samsun province, northern Türkiye. Before applying ARIMA and ML models, the data stationarity was checked, and then the best ARIMA and ML models were developed from these stationary data. The behavior of ACF and PACF criteria was analyzed to determine the appropriate ARIMA model.

The hybrid methods, which included ARIMA–ANN, ARIMA–GRNN, and ARIMA–ANFIS, showed the best accuracy compared to individual ARIMA model, as indicated by statistical performance criteria (*RMSE*, *MAE*, R^2). The ARIMA–GRNN model showed the highest improvement compared to ARIMA with a 48.38% decrease in RMSE values. The results of the hybrid models were satisfactory, as demonstrated by the comparison between observed and predicted values using different methods.

The hybrid approach has several advantages by combining the strengths of ARIMA and ML models, resulting in more accurate predictions. It also simplifies the process and allows for faster decision making. The findings of this study are noteworthy; however, it has limitations in terms of the data (2010–2020) and models used (ARIMA, ANN, ANFIS, and GRNN). Therefore, future studies could further improve the accuracy of ET_0 predictions by incorporating different hybrid techniques and deep learning models.

Author Contributions: Conceptualization, B.C. and E.K.; methodology, E.K., E.C. and H.S.; software, E.K. and E.C.; validation, B.C. and H.S.; formal analysis, E.K. and B.C.; data curation, B.C. and E.C.; resources, E.C. and H.S.; writing—original draft preparation, E.K. and E.C.; writing—review and editing, B.C. and H.S.; visualization, E.K., E.C. and H.S.; supervision, B.C. and H.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the first author. The data are not publicly available for ethical reasons.

Acknowledgments: The authors would like to thank the anonymous reviewers and the editor for their constructive comments and suggestions for this paper.

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

References

1. Tunca, E.; Köksal, E.S.; Torres-Rua, A.; Kustas, W.P.; Nieto, H. Estimation of bell pepper evapotranspiration using two-source energy balance model based on high-resolution thermal and visible imagery from unmanned aerial vehicles. *J. Appl. Remote Sens.* **2022**, *16*, 022204. [CrossRef]
2. Allen, R.G.; Pereira, L.S.; Raes, D.; Smith, M. Crop evapotranspiration-Guidelines for computing crop water requirements-FAO Irrigation and drainage paper 56. *FAO Rome* **1998**, *300*, D05109.
3. López-Urrea, R.; de Santa Olalla, F.M.; Fabeiro, C.; Moratalla, A. Testing evapotranspiration equations using lysimeter observations in a semiarid climate. *Agric. Water Manag.* **2006**, *85*, 15–26. [CrossRef]
4. Fidantemiz, Y.F.; Jia, X.; Daigh, A.L.; Hatterman-Valenti, H.; Steele, D.D.; Niaghi, A.R.; Simsek, H. Effect of water table depth on soybean water use, growth, and yield parameters. *Water* **2019**, *11*, 931. [CrossRef]

5. Kadioglu, H.; Hatterman-Valenti, H.; Jia, X.; Chu, X.; Aslan, H.; Simsek, H. Groundwater table effects on the yield, growth, and water use of canola (*Brassica napus* L.) plant. *Water* **2019**, *11*, 1730. [CrossRef]
6. Ferreira, L.B.; da Cunha, F.F.; de Oliveira, R.A.; Fernandes Filho, E.I. Estimation of reference evapotranspiration in Brazil with limited meteorological data using ANN and SVM—A new approach. *J. Hydrol.* **2019**, *572*, 556–570. [CrossRef]
7. Thornthwaite, C.W. An approach toward a rational classification of climate. *Geogr. Rev.* **1948**, *38*, 55–94. [CrossRef]
8. Blaney, H.F. *Determining Water Requirements in Irrigated Areas from Climatological and Irrigation Data*; US Department of Agriculture: Washington, DC, USA, 1950.
9. Priestley, C.H.B.; Taylor, R.J. On the assessment of surface heat flux and evaporation using large-scale parameters. *Mon. Weather Rev.* **1972**, *100*, 81–92. [CrossRef]
10. Hargreaves, G.H.; Samani, Z.A. Reference crop evapotranspiration from temperature. *Appl. Eng. Agric.* **1985**, *1*, 96–99. [CrossRef]
11. Prasad, R.; Deo, R.C.; Li, Y.; Maraseni, T. Input selection and performance optimization of ANN-based streamflow forecasts in the drought-prone Murray Darling Basin region using IIS and MODWT algorithm. *Atmos. Res.* **2017**, *197*, 42–63. [CrossRef]
12. Krishnashetty, P.H.; Balasangameshwara, J.; Sreeman, S.; Desai, S.; Kantharaju, A.B. Cognitive computing models for estimation of reference evapotranspiration: A review. *Cogn. Syst. Res.* **2021**, *70*, 109–116. [CrossRef]
13. Kumar, M.; Raghuwanshi, N.; Singh, R.; Wallender, W.; Pruitt, W. Estimating evapotranspiration using artificial neural network. *J. Irrig. Drain Eng.* **2002**, *128*, 224–233. [CrossRef]
14. Sattari, M.T.; Mahesh, P.; Yürekli, K.; Ünlükara, A. M5 model trees and neural network based modelling of ET_0 in Ankara, Turkey. *Turk. J. Eng. Environ. Sci.* **2013**, *37*, 211–219. [CrossRef]
15. Banda, P.; Cemek, B.; Küçüktopcu, E. Estimation of daily reference evapotranspiration by neuro computing techniques using limited data in a semi-arid environment. *Arch. Agron. Soil Sci.* **2018**, *64*, 916–929. [CrossRef]
16. Rahimi Khoob, A. Comparative study of Hargreaves's and artificial neural network's methodologies in estimating reference evapotranspiration in a semiarid environment. *Irrig. Sci.* **2008**, *26*, 253–259. [CrossRef]
17. Traore, S.; Wang, Y.-M.; Kerh, T. Artificial neural network for modeling reference evapotranspiration complex process in Sudano-Sahelian zone. *Agric. Water Manag.* **2010**, *97*, 707–714. [CrossRef]
18. Gocić, M.; Arab Amiri, M. Reference evapotranspiration prediction using neural networks and optimum time lags. *Water Res. Manag.* **2021**, *35*, 1913–1926. [CrossRef]
19. Abrishami, N.; Sepaskhah, A.R.; Shahrokhnia, M.H. Estimating wheat and maize daily evapotranspiration using artificial neural network. *Theor. Appl. Climatol.* **2019**, *135*, 945–958. [CrossRef]
20. Granata, F. Evapotranspiration evaluation models based on machine learning algorithms—A comparative study. *Agric. Water Manag.* **2019**, *217*, 303–315. [CrossRef]
21. Ladlani, I.; Houichi, L.; Djemili, L.; Heddam, S.; Belouz, K. Modeling daily reference evapotranspiration (ET_0) in the north of Algeria using generalized regression neural networks (GRNN) and radial basis function neural networks (RBFNN): A comparative study. *Meteorol. Atmos. Phys.* **2012**, *118*, 163–178. [CrossRef]
22. Pour-Ali Baba, A.; Shiri, J.; Kisi, O.; Fard, A.F.; Kim, S.; Amini, R. Estimating daily reference evapotranspiration using available and estimated climatic data by adaptive neuro-fuzzy inference system (ANFIS) and artificial neural network (ANN). *Hydrol. Res.* **2013**, *44*, 131–146. [CrossRef]
23. Kisi, O.; Zounemat-Kermani, M. Comparison of two different adaptive neuro-fuzzy inference systems in modelling daily reference evapotranspiration. *Water Res. Manag.* **2014**, *28*, 2655–2675. [CrossRef]
24. Gairaa, K.; Khellaf, A.; Messlem, Y.; Chellali, F. Estimation of the daily global solar radiation based on Box–Jenkins and ANN models: A combined approach. *Renew. Sustain. Energy Rev.* **2016**, *57*, 238–249. [CrossRef]
25. Musarat, M.A.; Alaloul, W.S.; Rabbani, M.B.A.; Ali, M.; Altaf, M.; Fediuk, R.; Vatin, N.; Klyuev, S.; Bukhari, H.; Sadiq, A. Kabul river flow prediction using automated ARIMA forecasting: A machine learning approach. *Sustainability* **2021**, *13*, 10720. [CrossRef]
26. Wang, W.; Du, Y.; Chau, K.; Chen, H.; Liu, C.; Ma, Q. A comparison of BPNN, GMDH, and ARIMA for monthly rainfall forecasting based on wavelet packet decomposition. *Water* **2021**, *13*, 2871. [CrossRef]
27. Arca, B.; Duce, P.; Snyder, R.L.; Spano, D.; Fiori, M. Use of numerical weather forecast and time series models for predicting reference evapotranspiration. In Proceedings of the IV International Symposium on Irrigation of Horticultural Crops 664, Davis, CA, USA, 1–6 December 2003; pp. 39–46.
28. Kishore, V.; Pushpalatha, M. Forecasting Evapotranspiration for Irrigation Scheduling using Neural Networks and ARIMA. *Int. J. Appl. Eng. Res.* **2017**, *12*, 10841–10847.
29. Landeras, G.; Ortiz-Barredo, A.; López, J.J. Forecasting weekly evapotranspiration with ARIMA and artificial neural network models. *J. Irrig. Drain Eng.* **2009**, *135*, 323–334. [CrossRef]
30. Box, G.; Jenkins, G.M. *Time Series Analysis: Forecasting and Control*; John Wiley & Sons: Hoboken, NJ, USA, 1976.
31. Hornik, K.; Stinchcombe, M.; White, H. Multilayer feedforward networks are universal approximators. *Neural Netw.* **1989**, *2*, 359–366. [CrossRef]
32. Jalili Ghazi Zade, M.; Noori, R. Prediction of Municipal Solid Waste Generation by Use of Artificial Neural Network: A Case Study of Mashhad. *Int. J. Environ. Res.* **2008**, *2*, 13–22.
33. Atkinson, P.M.; Tatnall, A.R. Introduction neural networks in remote sensing. *Int. J. Remote Sens.* **1997**, *18*, 699–709. [CrossRef]
34. Haykin, S.S. *Neural Networks and Learning Machines*; Pearson Education India: Delhi, India, 2009.

35. Yadav, N.; Yadav, A.; Kumar, M. *An Introduction to Neural Network Methods for Differential Equations*; Springer: Berlin/Heidelberg, Germany, 2015; Volume 1.
36. Specht, D.F. A general regression neural network. *IEEE Trans. Neural Netw.* **1991**, *2*, 568–576. [CrossRef] [PubMed]
37. Kim, B.; Lee, D.W.; Park, K.Y.; Choi, S.R.; Choi, S. Prediction of plasma etching using a randomized generalized regression neural network. *Vacuum* **2004**, *76*, 37–43. [CrossRef]
38. Daneshmand, H.; Tavousi, T.; Khosravi, M.; Tavakoli, S. Modeling minimum temperature using adaptive neuro-fuzzy inference system based on spectral analysis of climate indices: A case study in Iran. *J. Saudi Soc. Agric. Sci.* **2015**, *14*, 33–40. [CrossRef]
39. Küçüktopcu, E.; Cemek, B. Comparison of neuro-fuzzy and neural networks techniques for estimating ammonia concentration in poultry farms. *J. Environ. Chem. Eng.* **2021**, *9*, 105699. [CrossRef]
40. Keskin, A.Ü.; Beden, N.; Demir, V. Analysis of annual, seasonal and monthly trends of climatic data: A case study of Samsun. *Nat. Sci.* **2018**, *13*, 51–70.
41. Khan, M.M.H.; Muhammad, N.S.; El-Shafie, A. Wavelet based hybrid ANN-ARIMA models for meteorological drought forecasting. *J. Hydrol.* **2020**, *590*, 125380. [CrossRef]
42. Nguyen, X.H. Combining statistical machine learning models with ARIMA for water level forecasting: The case of the Red river. *Adv. Water Resour.* **2020**, *142*, 103656.
43. Alsharif, M.H.; Younes, M.K.; Kim, J. Time series ARIMA model for prediction of daily and monthly average global solar radiation: The case study of Seoul, South Korea. *Symmetry* **2019**, *11*, 240. [CrossRef]
44. Wu, X.; Zhou, J.; Yu, H.; Liu, D.; Xie, K.; Chen, Y.; Hu, J.; Sun, H.; Xing, F. The development of a hybrid wavelet-ARIMA-LSTM model for precipitation amounts and drought analysis. *Atmosphere* **2021**, *12*, 74. [CrossRef]
45. Lai, Y.; Dzombak, D.A. Use of the autoregressive integrated moving average (ARIMA) model to forecast near-term regional temperature and precipitation. *Weather Forecast.* **2020**, *35*, 959–976. [CrossRef]
46. Arora, S.; Keshari, A.K. ANFIS-ARIMA modelling for scheming re-aeration of hydrologically altered rivers. *J. Hydrol.* **2021**, *601*, 126635. [CrossRef]
47. Sarıca, B.; Eğrioğlu, E.; Aşıkgil, B. A new hybrid method for time series forecasting: AR–ANFIS. *Neural Comput. Appl.* **2018**, *29*, 749–760. [CrossRef]
48. Wei, W.; Jiang, J.; Liang, H.; Gao, L.; Liang, B.; Huang, J.; Zang, N.; Liao, Y.; Yu, J.; Lai, J. Application of a combined model with autoregressive integrated moving average (ARIMA) and generalized regression neural network (GRNN) in forecasting hepatitis incidence in Heng County, China. *PLoS ONE* **2016**, *11*, e0156768. [CrossRef] [PubMed]
49. Shahriar, S.A.; Kayes, I.; Hasan, K.; Hasan, M.; Islam, R.; Awang, N.R.; Hamzah, Z.; Rak, A.E.; Salam, M.A. Potential of Arima-ann, Arima-SVM, dt and catboost for atmospheric PM2. 5 forecasting in bangladesh. *Atmosphere* **2021**, *12*, 100. [CrossRef]
50. Waller, D.L. *Operations Management: A Supply Chain Approach*; Cengage Learning Business Press: Boston, MA, USA, 2003.
51. Abedi-Koupai, J.; Dorafshan, M.-M.; Javadi, A.; Ostad-Ali-Askari, K. Estimating potential reference evapotranspiration using time series models (case study: Synoptic station of Tabriz in northwestern Iran). *Appl. Water Sci.* **2022**, *12*, 212. [CrossRef]
52. Aghelpour, P.; Norooz-Valashedi, R. Predicting daily reference evapotranspiration rates in a humid region, comparison of seven various data-based predictor models. *Stoch. Environ. Res. Risk Assess.* **2022**, *36*, 4133–4155. [CrossRef]
53. Mossad, A.; Alazba, A. Simulation of temporal variation for reference evapotranspiration under arid climate. *Arab. J. Geosci.* **2016**, *9*, 1–9. [CrossRef]
54. Chen, Z.; Zhu, Z.; Jiang, H.; Sun, S. Estimating daily reference evapotranspiration based on limited meteorological data using deep learning and classical machine learning methods. *J. Hydrol.* **2020**, *591*, 125286. [CrossRef]
55. Dimitriadou, S.; Nikolakopoulos, K.G. Artificial neural networks for the prediction of the reference evapotranspiration of the Peloponnese Peninsula, Greece. *Water* **2022**, *14*, 2027. [CrossRef]
56. Kushwaha, N.L.; Rajput, J.; Sena, D.; Elbeltagi, A.; Singh, D.; Mani, I. Evaluation of data-driven hybrid machine learning algorithms for modelling daily reference evapotranspiration. *Atmos. Ocean* **2022**, *60*, 519–540. [CrossRef]
57. Sharma, G.; Singh, A.; Jain, S. A hybrid deep neural network approach to estimate reference evapotranspiration using limited climate data. *Neural Comput. Appl.* **2022**, *34*, 4013–4032. [CrossRef]
58. Ferreira, L.B.; da Cunha, F.F. New approach to estimate daily reference evapotranspiration based on hourly temperature and relative humidity using machine learning and deep learning. *Agric. Water Manag.* **2020**, *234*, 106113. [CrossRef]
59. Antonopoulos, V.Z.; Antonopoulos, A.V. Daily reference evapotranspiration estimates by artificial neural networks technique and empirical equations using limited input climate variables. *Comput. Electron. Agric.* **2017**, *132*, 86–96. [CrossRef]
60. Dou, X.; Yang, Y. Evapotranspiration estimation using four different machine learning approaches in different terrestrial ecosystems. *Comput. Electron. Agric.* **2018**, *148*, 95–106. [CrossRef]