International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 16, No.1 (Sep-24)

http://dx.doi.org/10.12785/ijcds/1601103

Performance and Robustness Analysis of Advanced Machine Learning Models for Predicting the Required Irrigation Water Amount

Hamed Laouz[1], Soheyb Ayad[1], Labib Sadek Terrissa[1], Aicha Nabila Benharkat[2] and Samir Merdaci[3]

[1]*LINFI Laboratory, University Mohamed Khider, Biskra, Algeria*
[2]*Institut National des Sciences Appliquées de Lyon, Lyon, France*
[3]*Agronomy Faculty, University of El-oued, El-oued, Algeria*

Received 5 Feb. 2024, Revised 8 Jun. 2024, Accepted 9 Jun. 2024, Published 20 Sep. 2024

Abstract: The agricultural sector plays a pivotal role in ensuring global food security, particularly in light of significant population growth. The demand for food is increasing substantially, while crop production may not sufficiently meet these rising needs. Water scarcity is one of the main problems that poses a significant challenge to the agriculture sector, exacerbated by inefficiencies in traditional irrigation methods. Accurate prediction of plant water requirements is essential to address this issue. This paper proposes advanced machine learning (ML) and deep learning (DL) models to accurately predict the daily water amount (quantity) needs of greenhouse plants using various air and soil data parameters. Various data preprocessing techniques were applied to prepare the data for the proposed models. In addition, due to the different nature of the proposed models, two different data splitting methods were used to split data into inputs and outputs (Simple data preparation for the ML models and time series data preparation for the time series DL models).Results indicate that the Multi-Layer Perceptron (MLP) model consistently outperformed other models, demonstrating superior stability and efficiency across different data optimization phases. Additionally, both ML and Long-Short Term Memory (LSTM) models exhibited strong performance in different data optimization scenarios. Robustness was evaluated through parameter sensitivity analysis, which revealed that ML models were generally more robust than DL models. This robustness is attributed to the limited number of parameters in ML models, which enhances their reliability compared to the more complex DL models. This study ensures the potential of the proposed models to optimize the irrigation practices, thereby addressing water scarcity issues and improving agricultural productivity.

Keywords: Precision irrigation, Water amount prediction, Data-based optimization, Hyper-parameters tuning, DL time series, Sensitivity analysis

1. Introduction

The global population reached 8 billion in November 2022 and is projected to grow to 9 billion by 2050 [1]. This has led to challenges, especially those related to food supply and freshwater resources, as water has become a critical resource that requires precise management, specifically in agriculture [2]. According to a report by the Food and Agriculture Organization (FAO) over the past century, global water use has increased at a rate that is more than twice as fast as population growth. In arid areas, population growth and economic development are putting unprecedented pressure on renewable but limited water supplies. Two-thirds of the world's population may be living in " stressful" circumstances by 2025, with 1.8 billion people predicted to reside in areas with "absolute" water shortage (Defined as less than $500m^3$ of water per person annually) and between 500 and $1000m^3$ of water per person annually [3].

The agricultural sector significantly contributes to water scarcity. According to an FAO report, 70% of the available freshwater resources are used for agriculture, and 60% of this water is wasted due to inefficient irrigation techniques [4]. These issues call for an innovative solution to rationalise water resource usage.

Numerous solutions have been developed to manage the irrigation water usage. Some utilise the evapotranspiration value as the primary indicator of plant water needs, while others define a soil moisture threshold value to determines when to initiate or stop irrigation based on the field measurement of soil moisture. Recent solutions have incorporated AI techniques. Yet comprehensive global monitoring of environmental changes affecting plant growth remains limited.

Many proposed solutions used the evapotranspiration

E-mail address: laouz.hamed@univ-biskra.dz, s.ayad@univ-biskra.dz, terrissa@univ-biskra.dz, nabila.benharkat@insa-lyon.fr, samir-merdaci@univ-eloued.dz

[5], [6], [7], [8], [9] value as information that describes the plant water needs. Evapotranspiration (ET) refers to the water loss to the atmosphere through two processes: evaporation and transpiration. Evaporation involves water loss from open bodies of water, such as lakes, reservoirs, wetlands, bare soil, and snow cover. Transpiration, on the other hand, is the water loss from the surfaces of living plants [10]. The ET-based solutions use different empirical models to obtain the ET value and various parameters such as rainfall, wind speed, solar radiation, and other weather and air parameters [11]. The main drawback of these models is that they need exact weather data to give an accurate ET value, which can be difficult to obtain [12], [13].

Other researches have focused on defining a threshold that controls the irrigation [14], [15], [16], [17], typically using soil moisture levels. When the current soil moisture reaches the predefined threshold, the water pump activates to irrigate the plants. If the soil moisture is above the threshold, the water pump remains off. The choice of soil moisture thresholds can be a challenging point because if the threshold is set too high, the result will be over-irrigation, while setting it too low, will result in under-irrigation and thus production losses [18].

In recent years, researchers have focused on using artificial intelligence (AI) and taking advantage of its ability to solve complex problems, specifically focusing on predicting the precise amount of water needed for irrigation [19], [20], [21], [22], [23], [24], [25]. While AI presents promising solutions for smart irrigation, most proposed solutions focus on predicting irrigation related values such as soil moisture and ET using a limited number of parameters.

Despite the various solutions proposed and extensive research aimed at optimizing the irrigation process and reducing water wastage, there is still a need for a comprehensive solution capable of addressing the significant limitations of previous works. Such a solution should accurately predict plant water needs without any wastage.

In this study, we aim to address the limitations of previous research by introducing advanced ML/DL models capable of accurately predicting the precise daily water quantity required by plants. These predictions utilize a diverse set of air and soil data parameters, enhancing the models' precision. Additionally, we conducted a comprehensive analysis to assess the impact of data optimization on the performance of these ML/DL models, evaluating each model's response to different data processing techniques. Furthermore, we performed a sensitivity analysis to evaluate the robustness of each model, providing insights into their stability under various conditions.

The contributions of this paper are outlined as follows:

- Direct prediction of daily irrigation water amount, utilizing various plant environmental conditions throughout plant evolution, ranging from air parameters to soil parameters.
- Sensitivity analysis conducted on the proposed models to evaluate their efficiency and robustness.

The remaining of this paper is structured as follows: Section 2 provides a detailed data description, materials and methodology used in this study. In Section 3, we present the results of our study, followed by a discussion of their implications in Section 4. Finally, Section 5 offers concluding remarks and suggestions for future research directions.

2. Materials and Methods

The main goal of this work is to accurately predict the daily irrigation water amount influenced by plant environmental factors using a specified model M.

Figure 1 represents the global architecture of the proposed system, which composed mainly of two stages:

- **Training stage:** This stage is conducted offline, where we train our model using specific agricultural data that affect the daily water amount, following necessary data preprocessing and preparation steps. The data used are historical records from a successful plant-growing experiment with optimal irrigation decisions. The training process is repeated with different configurations and during various data optimization tasks until the lowest error score is achieved.
- **Prediction stage:** Once the model is ready, it can be applied in real-time environment to predict the accurate water amount using current daily data values. After a specified period, the real-time data records will be collected and added to the historical dataset to retrain the model, thus improving its performance through data augmentation.

.

A. Data Description

Collecting data regarding the plant environment is crucial for estimating the quantity of irrigation water needed because this data is considered the main factor affecting the irrigation process. To achieve this, we utilized data from the first edition of the Autonomous Greenhouse Challenge (AGC) [26]. The AGC is a competition held in the Netherlands, involving six teams (five competing teams and one experts team) tasked with managing the growth of cucumber plants within a greenhouse (GH). Each team has its own GH with various sensors. Their job is to design and develop machine learning (ML)/deep learning (DL) models that make accurate decisions regarding the plants ' growing conditions (Irrigation/ ventilation/ heating/ CO_2 dosage ... etc.).

The AGC dataset comprises five collections, each corresponding to a different team. Each collection contains

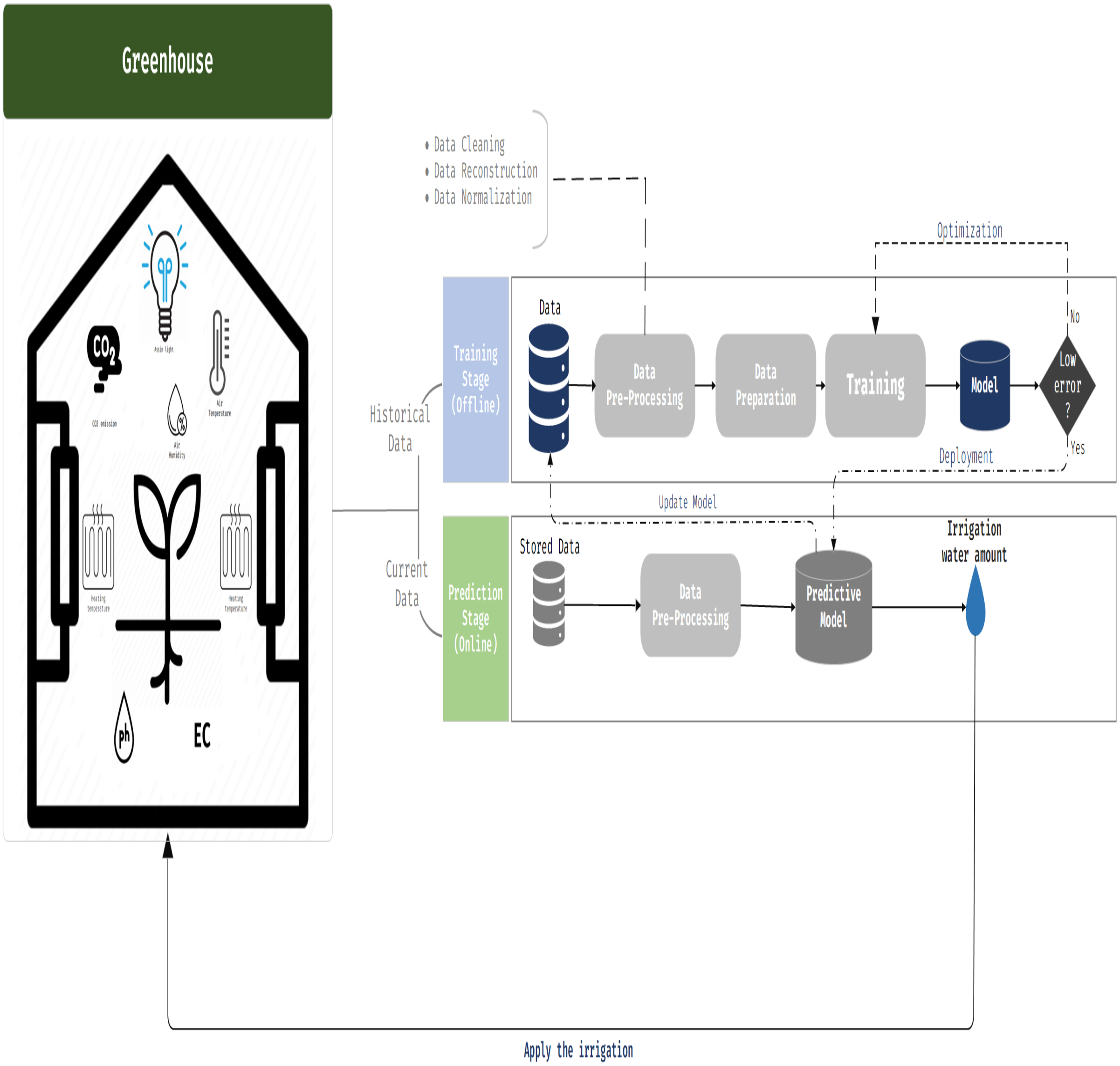


Figure 1. Proposed system process

six datasets that detail the team's management of the plant growth environment within the greenhouse.

- **Crop Management**
- **Greenhouse Climate**
- **VIP**
- **Irrigation**
- **Production**
- **Resources Calculation**

Because the irrigation process can be affected by different plant air environmental changes and soil data [27], we selected the GH Climate and Irrigation datasets from the winning team as the essential data needed to predict the irrigation water amount.

Table I describes the used data parameters. The GH Climate dataset includes 33133 rows of 115 days of growth, with each row representing data recorded every 5 minutes. The irrigation dataset consists of 115 rows, each corresponding to daily irrigation information for the respective day. In the AGC competition, the "drain" parameter was used to determine net water usage, but it is not relevant for this study, so it has been excluded.

TABLE I. Selected Datasets Parameters Description

Dataset	Feature	Description	Unit
GH Climate	Tair	GH air temperature	°C
	RHair	GH relative air humidity	%
	AssimLight	GH artificial light	%
	CO_2air	GH air CO_2 concentration	ppm
	HumDef	GH Humidity deficit	g/m^2
	Ventwind	Ventilation wind speed	%
	PipeLow	Rail pipe heating temperature on the floor	°C
	PipeGrow	Pipe heating temperature on the crop height	°C
Irrigation	pH_Drain	Daily average of drain water PH	[-]
	EC_Drain	Daily average of drain water EC	dS/m
	drain	Daily drain water	l/m^2
	water	Daily irrigation water amount	l/m^2

B. Data Preprocessing

1) Data Cleaning

Data cleaning is a technique used to improve the data quality by detecting and removing errors, and inconsistent and false data [28]. In this work we have applied two different data cleaning techniques:

- **Missing values**: Table II shows the number of missing values of the "GH Climate" and "Irrigation" datasets parameters respectively. For the "GH Climate" dataset, we found out that there are 142 missing values for each column –parameter– of a total of 331334 values, which is less than 01% of the total data. For the "Irrigation" dataset, the only missing value was for the water parameter on day 64, representing 0.87% of the total values (115 rows).
  Since both datasets contain less than 01% missing values, we chose to disregard these values during the data reconstruction phase.

TABLE II. Missing values for each parameter

Dataset	Feature	Nb. Missing Values
GH Climate	Tair	142
	RHair	143
	AssimLight	142
	CO2air	142
	HumDef	142
	Ventwind	142
	AssimLight	142
	PipeLow	142
	PipeGrow	142
Irrigation	water	1
	EC_Drain	0
	pH_Drain	0

- **Handle outliers**: An outlier is a data point that deviates significantly from the other data points in a dataset [29], it's existence in the data may affect the model's performance. For that detecting and handling the outliers is a mandatory step to do. To detect the outliers, we have used the z-score technique which is a common method for detecting the outliers. The z-score is simply a test that measures the divergence of a different experimental observation from the most probable result, the mean [30].
  Table III shows the number of rows containing outliers accompanying their percentage of the total data. Because the detected outliers carry real data (Not missing values or out of range), the techniques of modifying the outliers may affect the results of our models. So, we decided to study the impact of deleting the outliers on the model's performance by analysing the results with the original data and without the existence of outliers. Thus, we have chosen the outliers detected when *Threshold* = 3 because it contains only 14% of total data (Too small 03% when threshold =4 / Too many when threshold=2).

TABLE III. Detected Outliers using z-score test

Threshold	Nb. Detected Outliers	Percentage
02	37	33%
03	16	14%
04	03	03%
05	01	0.87%

2) Data Reconstruction

Because of the difference in the time intervals between the parameters of the two used datasets, we have reconstructed the used datasets by combining them into a new dataset that contains a unified time interval (Daily interval) by calculating the daily average of the GH Climate (initially recorded at minute intervals) dataset parameters, except for the "*AssimLight*" parameter, for which we calculated its total daily working time (in minutes).

The reason for calculating the daily working time of the *"AssimLight"* is because this parameter is categorical in the original dataset (GH Climate dataset), which means that the value of this parameter has only two values: 0%(OFF) or 100%(ON).

Algorithm 1 describes the process of data reconstruction that we have used. The new dataset *Ndata* is the form of a matrix of *n* columns and *m* rows(lines), where *n* corresponds to the number of the parameters used, and *m* corresponds to the number of samples used which are the growing days. Also, when we make the sum of values, we ignore the missing values that occurred in the original data.

Algorithm 1 Data reconstruction

```
Require: GH_Climate
Require: Irrigation
      Remove unwanted parameters
  GH_Climate ← GH_Climate − {"time_index"}
  Irrigation ← Irrigation − {"drain"}
  Ndata ← {∅}
  for k = 1; i ≤ N do
      for i = 0; i ≤ 114 do
          sum ← 0
          for j = 1; j ≤ 288 do
              current ← GH_Climate[K][i ∗ 288 + j]
              if (k is "AssimLight" and current=100) then
                  sum ← sum + 5
              end if
              if (k is not "AssimLight") then
                  sum ← sum + current
              end if
          end for
          if (k is "AssimLight) then
              Ndata[k][i] ← sum
          else
              avg ← sum/288
              Ndata[k][i] ← avg
          end if
      end for
      Ndata[k][i] ← avg
  end for
      return (Ndata)
```

N.B: the value "**288**" mentioned in the algorithm represents the number of rows that form a complete day of data.

3) Data Normalization

- Data scaling: Feature scaling is the process of standardizing data so that the features have similar magnitudes, units, and ranges. This is essential when the data varies widely, as unscaled data can cause some machine learning algorithms to underperform by not properly accounting for the variance in feature set data [31].
Due to the various type of features used with the different ranges (%, °C, [1-10] for PH,...etc), scaling the data is an essential step before training the models in order to prevent any model's under-perform caused by the difference in the data ranges. Thus, we have used the *MinMax Scaler* with a feature range of [0-1] to scale our data.

$$X'_i = \frac{X_i - min(X_i)}{max(X_i) - min(X_i)} \quad (1)$$

Where X'_i corresponds to the scaled value, $min(X_i)$ and $max(X_i)$ are the minimum and maximum values of the parameter i.

- Stationary data: Stationarity means that a stochastic feature does not change even if time changes [32]. Many statistical tests can be used to detect the data stationarity status. One of the most used tests is The augmented Dickey-Fuller (ADF) test. The ADF is the extended version of the simple Dickey-Fuller test, which suppose at first that the data is not stationary (null hypothesis), and then the ADF calculates the p-value. If this value is less than the significant level (usually 0.5), the ADF reject the null hypothesis and accepts the alternative. The alternative supposes that the time series data is stationary [33]. After passing the ADF test to all the parameters, we find out that:
  - *"water", "Pipe Grow Temp", "Daily Pipe Low", "Ventilation wind", "GH Temp" and "Assim Light"* are stationary data.
  - The remaining parameters are non-stationary data.

The excising of non-stationarity parameters means that these parameters are related to the time factor. This may invoke a problem with the models' performance. To avoid any possible problem, we need to change this data into stationary data and observe its impact on the models' behaviour. **"Differencing data"** is a common method that changes the non-stationary data into stationary using the following equation:

$$X'_i = X_i - X_{i-1} \quad (2)$$

Where: X_i: Non-stationary data at time i, X_{i-1}: Non-stationary data at time $i-1$, and X'_i: Stationary data at time i. Differencing the data can help stabilize the mean of a time series by removing changes in the series' level. This process effectively reduces or eliminates trends and seasonality [34].
After one order of data differencing, all the used parameters have become stationary. This step is used to analyze the impact of the stationary data on the model's performance.

C. Data Preparation

This section describes the different methods used for splitting the data into inputs and outputs.

1) Simple Data Preparation

Because the used ML models and MLP model cannot deal with the data as a sequence, we dealt with this data as a discrete problem, meaning that to predict the daily irrigation at day i we gave the model the inputs, which are the environmental change of the plant at the day i with the day index as information that describe the current position of the sequence.

Algorithm 2 describes the process of the simple data preparation.

Algorithm 2 Simple Data Preparation

Require: $data$
Require: day_index
  $Y \leftarrow data[\text{"water"}]$ ▷ Output
  $X \leftarrow data - \{data[\text{"water"}]\}$ ▷ All columns except water column (output)
  return (X, Y)

2) Time Series Data Preparation

Since the used data includes timestamps, and given the utilization of certain time series Deep Learning DL models in this study, we have introduced a second approach for data preparation for the time series models. This method involves employing the window slide concept to segment the data.

Algorithm 3 outlines the data preparation process utilizing the *window size* parameter. In this context, the output (water amount) on the day i is not only dependent on the inputs (plant's environment parameters) of the day (i), but also on the inputs ranging from $i - window_size$ to the day i.

Algorithm 3 Window Sliding Data Preparation

```
Require: data
Require: window_size
  Y ← data["water"]
  Z ← data − {data["water"]}
  X ← {∅}
  for i = 1to len(data) − window_size do
      T ← {∅}
      for j = i; j ≤ i + window_size do
          T ← T ⋃{Z[i]}
      end for
      X ← X ⋃{T}
  end for
  return (X,Y)
```

D. Proposed Models

In this work, we developed various models to predict the required irrigation water amount that varying from the ML models to the DL models ending with the advanced time series Models:

- Ada Boost regressor (ABR)
- Extra Tree regressor (ETR)
- Gradient Boost regressor (GBR)
- Multi-layer perceptron (MLP)
- Recurrent Neural Network (RNN)
- Long Short Term Memory (LSTM)

E. Evaluation Metrics

In this work, the performance of the suggested models was assessed using two metrics:

$$Mean\,Absolute\,Error\,(MAE) = \frac{\sum_{i=1}^{n} |Y_i - Y'_i|}{n} \tag{3}$$

$$Root\,Mean\,Squared\,Error\,(RMSE) = \sqrt{\frac{\sum_{i=1}^{n} (Y_i - Y'_i)^2}{n}} \tag{4}$$

Where Y_i corresponds to water amount needs for plant at the day i, and Y'_i represents the predicted output (water amount).

F. Hyper-parameter Tuning

This section aims to identify the optimal hyper-parameter sets for each model to achieve the best results. We compared these results using MAE (The hyper-parameter that gives the minimum MAE will be selected).

1) Grid-Search CV

The grid search is an exhaustive search based on defined subset of the hyper-parameter space [35]. Table IV represents the configuration of the hyper-parameter tuning and the range of possible values of the proposed ML models using the Grid-Search CV method.

TABLE IV. ML hyper-parameters' configuration ranges

Model	Parameter	Interval
ABR GBR ETR	$n\_estimator$	[1-100]

2) Bayesian Optimization

Because the DL models contain many hyperparameters that need to tune to obtain the best results, and because the Grid-Search CV method will be much more expensive, we chose to use the Bayesian optimization to find the best hyperparameters configuration that achieves the best results.

Bayesian optimization is an approach that optimizes objective functions that take a long time to evaluate. It works well when optimizing continuous domains with fewer than 20 dimensions [36].

Table V represents the configuration for the DL models using the Bayesian optimization. Because the Bayesian optimization method search for the max value possible, and we need to find the lowest possible error rate, we have converted the target value (MAE) into a negative value. Therefore, when the Bayesian method finds the max negative value, it will be the lowest MAE value. In addition, we have fixed the "Nb Iterations" parameter at 100 iterations.

3. Results

A. Models' Performance Results

This section presents the results obtained during various data optimization phases to evaluate the performance of the proposed models. The goal of this step is to identify the most accurate model with the smallest error margin and stable performance across different data optimization treatments, ensuring consistent and reliable performance.

Firstly, we have analyzed the results before applying any data optimization. Then, we have studied the impact of detecting the outliers on the models' behavior and results. Also, because the used data is time-stamped where the irrigation process is applied during all of the growing days and each water quantity could affect , we have studied the impact of stationary/non-stationary data on the proposed models (especially RNN and LSTM). Lastly, both of these optimization techniques have been applied simultaneously to analyze their impact on the models' performance when used together.

1) Obtained Result without Data Optimization

Figure 2 shows the obtained results of MAE and RMSE of the different used models without any data optimization where we can see that the MLP and LSTM models gave the lowest MAE and RMSE values (0.038 and 0.047 respectively for MAE and 0.048 and 0.058 respectively for RMSE).

On the other hand, the RNN model gave the highest MAE and RMSE scores compared to the other models. Also, the $RMSE - MAE$ value (the gap between MAE and RMSE) is the highest, which means that: there is a possible massive error that may occur in this model.

2) Comparison of Results after Removing Outliers

Because outliers are one of the main factors that may affect the performance of the models, especially the DL models [37], we decided to study the impact of outliers on the models performance.

Figure 3 shows the obtained results of the ML/DL models after removing the outliers, and Figure 4 shows the difference between these results with the initial results.

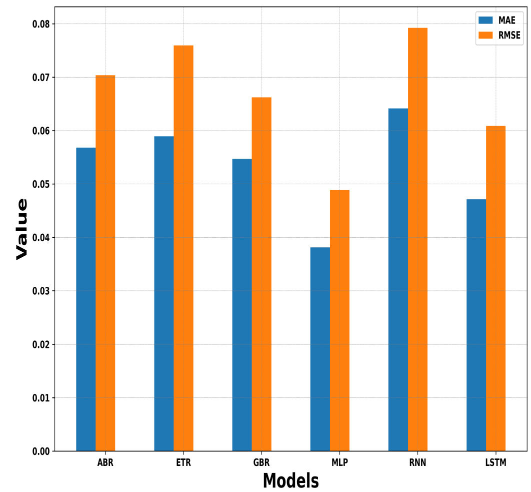


Figure 2. Obtained results without data optimization

Removing the outliers made the ML models record the worst scores and even worst than the initial results, especially the ABR and ETR models. On the other hand, the DL models gave better results compared to the ML models. Although the LSTM was the best model that gave the lowest MAE and RMSE values among all the models, this did not make any improvement compared to the initial results. For the RNN, we got a decent MAE value. However, the RMSE value was too big (Huge gap between MAE and RMSE), which can be explained by a possible overfitting problem that happened to the RNN model caused by the lack of data by removing the outliers (Data lost).

Although there were differences in the obtained results among the models, with some yielding better results than others, the overall performance of the proposed models significantly decreased after outliers were removed. This decline can likely be attributed to the loss of significant data volume due to the removal of outliers. It is well-known that ML/DL models are highly sensitive to data quantity, and the reduction in data volume negatively impacted their performance.

3) Comparison of Results after Differencing Data

Figures 5 and 6 show the results obtained after differencing the data.

These results show a significant improvement concerning all the ML models, especially the ETR method, which gave the lowest results among all the models (0.03 for MAE and 0.03 for RMSE). For the DL model, the RNN model was the only model that took the benefit of differencing the data where it recorded very low MAE and RMSE values compared to the initial results.

TABLE V. DL hyper-parameters' configuration ranges

Layer	Parameter	Range
Fully Connected	*Nb Layers*	[1-10]
	Nb Neurons	[1-500]
	Dropout	[0.1-0.8]
RNN/LSTM	*Nb Layers*	[1-10]
	Nb Neurons	[1-500]
	Dropout	[0.1-0.8]
	Sequence_length/ Window_size	[2-10]
General	*Optimization function*	[*adam, rmsprop, sgd, adadelta, adagard, adamax, nadam*]
	Batch_size	[1-100]
	Epochs	[1-500]
	Layers' Activation function	[*relu, tanh, sigmoid, softmax, softplus, elu, selu*]

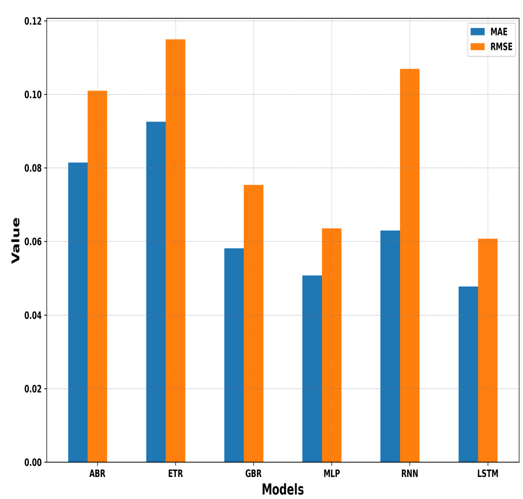
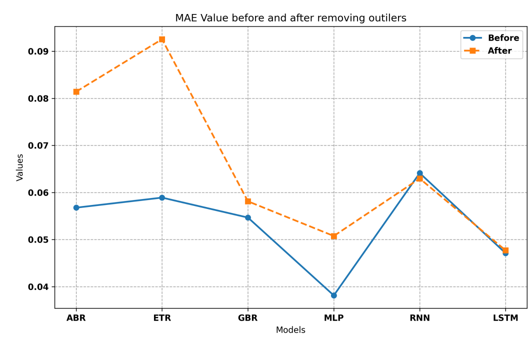


Figure 3. Obtained results after removing outliers.

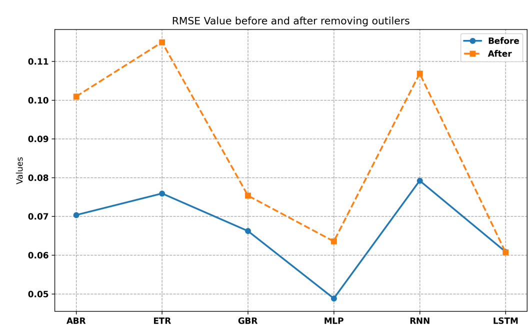
4) Comparison of Results after Applying all Data-based Optimization

Figures 7 and 8 show the results after applying all Data-based optimization. The results show that applying both optimization methods did not help at all to improve the model's results. Instead, it gave a much worse score compared to the initial results.

N.B: The differencing method changes the data values, which could affect the possible outliers. For that, we applied the z-test initially before removing the data.



(a) Comparison of MAE values



(b) Comparison of RMSE values

Figure 4. Comparison of results before and after removing the outliers

B. Comparison of Models' Robustness

Finding a model that minimizes error is necessary but not sufficient for deployment. Evaluating a model's robustness is equally crucial. A models' robustness refers

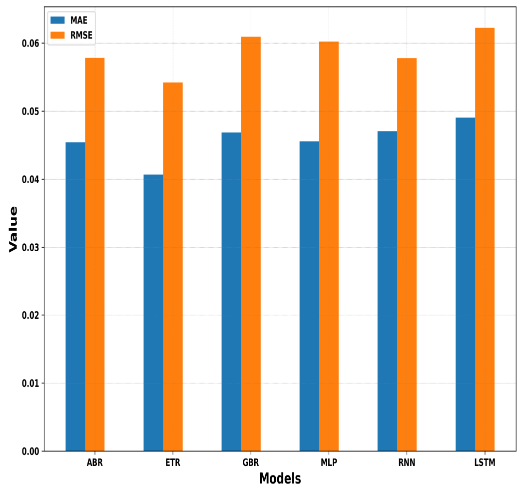


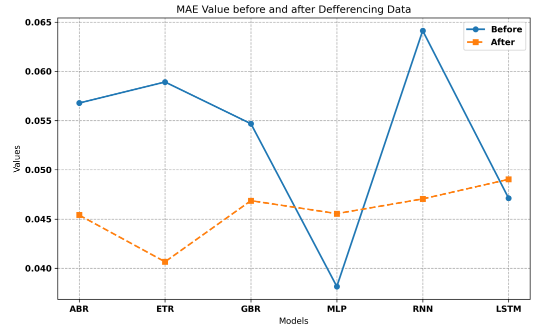
Figure 5. Obtained results after differencing data

to the ability of a given model to maintain its performance in different conditions [38]. Sensitivity analysis can be used to rank the influence of different hyper-parameters on the model's performance [39], which can help identify the model's robustness by analyzing the distribution of the results during the hyper-parameters tuning.

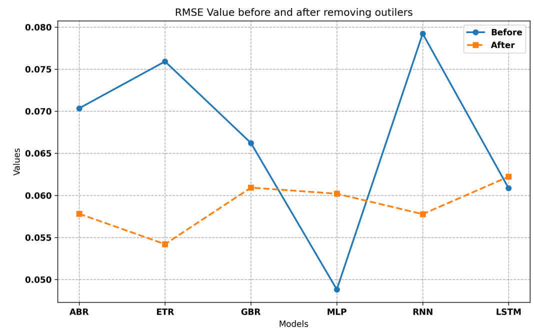
Figure 9 and Table VI illustrate the result of sensitivity analysis of different hyper-parameters configurations, allowing us to track each model's performance and gain insights into their robustness.

For the ML models, we see that both the ABR and ETR models gave the lowest standard deviation values (Around 0.001), which means that the obtained results of these two models during the different hyperparameters configuration are always close to the mean. This can be explained by the fact that these models have only one parameter that has been tuned, so the deviation of the results will not going to be very wide around the mean. On the other side, the GBR model was the most ML mode sensitive to hyperparameters changes with a notable extensive standard deviation (Around 0.01) compared to ABR and ETR.

For the DL models, the interval of the obtained results has notably expanded compared to the ML models where the MLP and RNN gave an MAE and RMSE close to 0.8 in some hyperparameters configurations while the minimum values obtained are too small in some cases where the MLP we got MAE=0.038 which is the best result obtained among all the models, thus has made the standard deviation for these models too huge compared to the ML models, that means the results are skewed to either to the left (Close to the zero) or to the right (Close the max) because the *mean* > *median* in all the DL models that means that these results



(a) Comparison of MAE values



(b) Comparison of RMSE values

Figure 6. Comparison of results before and after differencing the data

have a left-skewed where most of the results distributed in the left of the median.

This massive difference between ML and DL models in result distributions can be attributed to the variety of hyperparameters used for DL models versus the single parameter for ML models. However, since most DL results are near the best outcomes (close to zero), it suggests that DL models can still be robust despite their higher variability.

4. Discussion

Overall, all proposed models performed well in predicting irrigation amount, with noticeable differences among them. The MLP was the best model, which gave the best results during all the data optimization phases since the MLP was the model that gave the lowest MAE (0.038 when no data-based optimization applied) value with a small error variation magnitude (Small $RMSE-MAE$ value). The LSTM model also recorded stable results during almost all optimization processes (differencing data) with a low MAE (generally 0.05) and a small *RMSE-MAE* value. The ML models also gave decent scores with a not too big *RMSE-MAE* value except the GBR model, which was a little bit worse than the other ML models, where this model recorded the highest MAE and RMSE scores. On the other hand, the RNN model gave the highest MAE and RMSE scores.

TABLE VI. Statistical analysis of the results of the proposed models

Model	Metric	Min	Max	Mean	Std	Median
ABR	MAE	0.056	0.062	0.058	0.001	0.05
	RMSE	0.069	0.077	0.072	0.001	0.072
ETR	MAE	0.058	0.081	0.07	0.001	0.07
	RMSE	0.075	0.102	0.086	0.002	0.086
GBR	MAE	0.054	0.086	0.073	0.01	0.076
	RMSE	0.066	0.105	0.09	0.013	0.094
MLP	MAE	0.038	0.72	0.127	0.131	0.07
	RMSE	0.048	0.722	0.14	0.128	0.084
RNN	MAE	0.064	0.064	0.21	0.127	0.148
	RMSE	0.079	0.717	0.255	0.123	0.2
LSTM	MAE	0.047	0.593	0.117	0.102	0.076
	RMSE	0.058	0.597	0.13	0.099	0.088

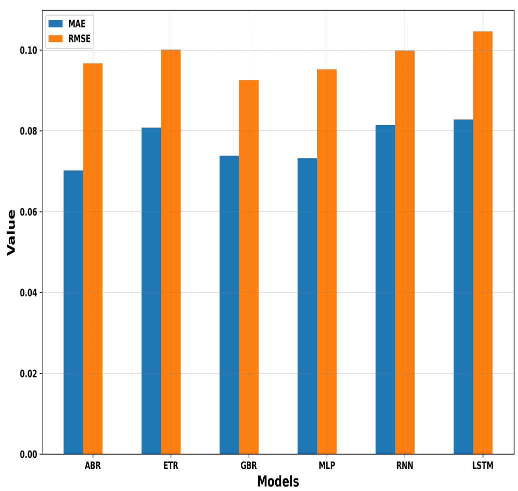
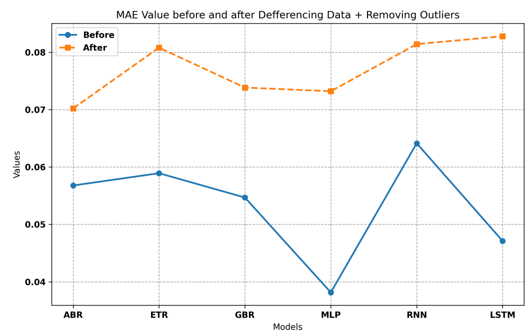


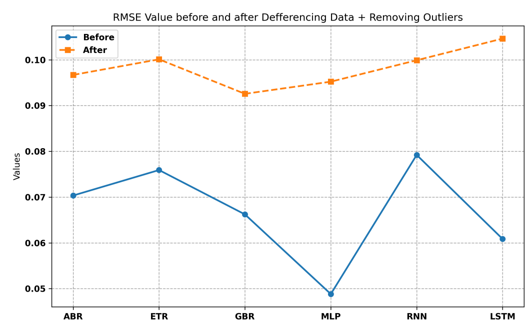
Figure 7. Obtained results after removing outliers + differncing data.

Although the RNN model recorded the highest error score among the models, its results aren't too poor because even if the RNN gave the worst MAE value (0.08 l/m^3), it still could be considered as a small error margin in the irrigation water amount.

Each data optimization task made a different impact on the proposed model's performance. Concerning the task of removing the outliers, almost all the models recorded unacceptable reactions to this optimization. This reaction can be attributed to the substantial loss of data, which had previously aided these models in achieving better generalization. Regarding the impact on stationary data, almost all of them exhibited positive responses to this optimization, yielding noticeably improved results compared to the initial outcomes. This improvement can be attributed to the time series nature of the data used in this work.



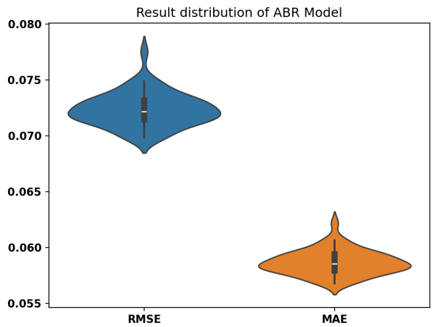
(a) Comparison of MAE values



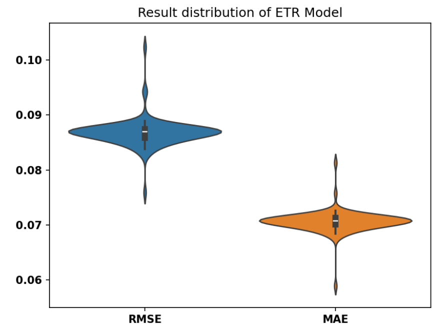
(b) Comparison of RMSE values

Figure 8. Comparison of results before and after applying all the optimization

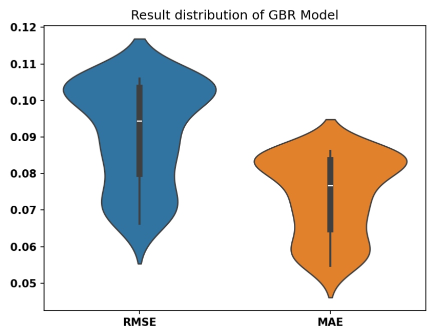
The robustness analysis is critical before deploying any model in real-world situations. The sensitivity analysis of the proposed models shows that the ML models could be more robust than the DL models. However, the possibility of in-time re-training the DL models based on the feedback could be a high advantage that could cover their weakness in terms of robustness.



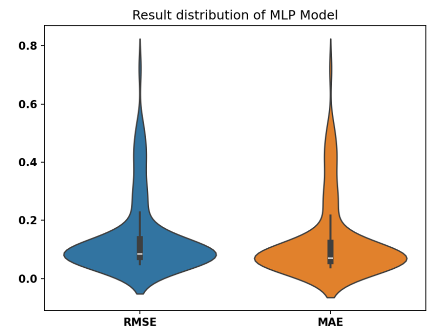
(a) ABR



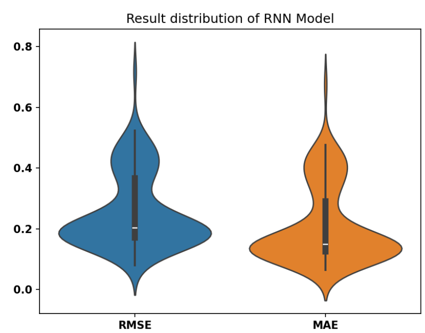
(b) ETR



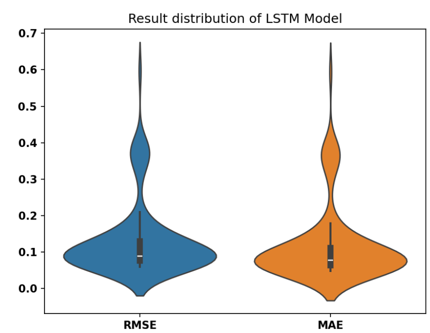
(c) GBR



(d) MLP



(e) RNN



(f) LSTM

Figure 9. MAE and RMSE results distributions during the hyper-parameters tuning.

Despite the potential impact of the proposed models, some limitations should be noted:

- The data reconstruction phase calculates the daily average for each per-minute parameter to match the daily recorded parameters. This method leads to losing a huge volume of data by condensing the total daily recorded data into a single value. Since DL models are more sensitive to data, this creates a significant challenge. Dropping these enormous amounts of data can potentially impact the model's accuracy and effectiveness, as it may remove valuable information within the day.
- DL models pose a significant limitation in terms of robustness, as sensitivity analysis results show a wider results distribution of the DL models during the different hyper-parameters configurations.
- The proposed ML/DL models were trained on a data from only one experiment of growing a cucumber crop. Although the experiment was conducted a team

that won a growing competition , training the models solely on this data may bias them towards the team's irrigation strategy and could hinder their generalizability.

Future work may focus on addressing the above-mentioned limitations.

5. Conclusions and future work

This work aimed to address the issue of water wastage in greenhouse environments by accurately predicting the daily water amount (quantity) requirements using a variety of DL and ML models. The models were fed with different set of input parameters, encompassing both air plant environment factors (such as air temperature, humidity, heating temperature) and underground parameters (PH and EC). Different data preprocessing techniques were employed to handle the corrupt and inaccurate data with specific time series data preparation applied for LSTM and RNN models.

Results indicate that the MLP model outperformed other models, demonstrating superior accuracy with consistent stability throughout all optimization phases. Concerning robustness, sensitivity analysis revealed the greater robustness of ML models compared to DL models, attributed to the limited number of hyperparameters employed in ML models (typically one parameter). Among DL models, the analysis indicated that the LSTM model exhibited potential robustness, as evidenced by a lower results distribution compared to RNN and MLP models.

Despite these significant results, it is essential to note that losing an extensive amount of data during the data reconstruction phase, particularly from the "GH climate" dataset parameters, may compromise the models' performance. These omitted data could potentially contain crucial information about the dynamic status of the plants throughout the day.

As a future research, an enhanced data reconstruction method may be proposed to unify the time interval between the used parameters without losing significant data. This could improve accuracy in predicting water amounts and further contribute to the advancement of sustainable water management practices in greenhouse environments.

Acknowledgements

This work is supported by the Algerian Directorate General for Scientific Research and Technological Development (DGRSDT) under the National research projects program in association with the Technical Institute for the Development of Saharan Agronomy (ITDAS) of Biskra.

References

[1] United Nations Department, "World population prospects 2022: Summary of results," Tech. Rep. UN DESA/POP/2022/TR/NO. 3, 2022.

[2] S. Wanniarachchi and R. Sarukkalige, "A review on evapotranspiration estimation in agricultural water management: Past, present, and future," *Hydrology*, vol. 9, no. 7, 2022. [Online]. Available: https://www.mdpi.com/2306-5338/9/7/123

[3] P. Steduto, J. M. Faurès, J. Hoogeveen, J. Winpenny, and J. Burke, "Coping with water scarcity: an action framework for agriculture and food security," Rome, Tech. Rep., 2012.

[4] K. Chartzoulakis and M. Bertaki, "Sustainable water management in agriculture under climate change," *Agriculture and Agricultural Science Procedia*, vol. 4, pp. 88–98, 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2210784315000741

[5] M. Dukes, M. Shedd, and S. Davis, "Smart irrigation controllers: Operation of evapotranspiration-based controllers1," *EDIS*, vol. 2009, 04 2009.

[6] S. Davis, M. Dukes, and G. Miller, "Landscape irrigation by evapotranspiration-based irrigation controllers under dry conditions in southwest florida," *Agricultural Water Management*, vol. 96, no. 12, pp. 1828–1836, 2009. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0378377409002340

[7] J. T R, N. S. Reddy, and U. D. Acharya, "Modeling daily reference evapotranspiration from climate variables: Assessment of bagging and boosting regression approaches," *Water Resources Management*, vol. 37, no. 3, pp. 1013–1032, Feb 2023.

[8] Y. Ma, S. Liu, L. Song, Z. Xu, Y. Liu, T. Xu, and Z. Zhu, "Estimation of daily evapotranspiration and irrigation water efficiency at a landsat-like scale for an arid irrigation area using multi-source remote sensing data," *Remote Sensing of Environment*, vol. 216, pp. 715–734, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0034425718303523

[9] P. Droogers, W. Immerzeel, and I. Lorite, "Estimating actual irrigation application by remotely sensed evapotranspiration observations," *Agricultural Water Management*, vol. 97, no. 9, pp. 1351–1359, 2010. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0378377410001149

[10] R. L. Hanson, "Evapotranspiration and Droughts," https://geochange.er.usgs.gov/sw/changes/natural/et/, 1991, [Accessed 08-Jun-2023].

[11] F. Poyen Phd, d. ghosh, and D. kundu, "Review on different evapotranspiration empirical equations," *International Journal of Advanced Engineering, Management and Science (IJAEMS)*, vol. 2, pp. 17–24, 03 2016.

[12] D. Choudhary, "Methods of evapotranspiration," https://www.researchgate.net/publication/323945435_Methods_of_Evapotranspiration, 03 2018.

[13] A. Mokhtar, N. Al-Ansari, W. El-Ssawy, R. Graf, P. Aghelpour, H. He, S. M. Hafez, and M. Abuarab, "Prediction of irrigation water requirements for green beans-based machine learning algorithm models in arid region," *Water Resources Management*, Mar 2023.

[14] T. Boutraa, A. Akhkha, A. Alshuaibi, and R. Atta, "Evaluation of the effectiveness of an automated irrigation system using wheat crops," *Agriculture and Biology Journal of North America*, vol. 88, pp. 2151–7517, 01 2011.

[15] M. Goodchild, K. Kühn, A. Burek, M. Jenkins, and A. Dutton, "A method for precision closed-loop irrigation using a modified pid

control algorithm," *Sensors and Transducers*, vol. 188, pp. 61–68, 05 2015.

[16] J. Gutierrez, J. Villa Medina, A. Nieto-Garibay, and M. Porta-Gandara, "Automated irrigation system using a wireless sensor network and gprs module," *Instrumentation and Measurement, IEEE Transactions on*, vol. 63, pp. 166–176, 01 2014.

[17] V. V. Borodychev and M. N. Lytov, "Irrigation management model based on soil moisture distribution profile," *IOP Conference Series: Earth and Environmental Science*, vol. 577, no. 1, p. 012022, sep 2020. [Online]. Available: https://dx.doi.org/10.1088/1755-1315/577/1/012022

[18] T. Kelly, T. Foster, D. M. Schultz, and T. Mieno, "The effect of soil-moisture uncertainty on irrigation water use and farm profits," *Advances in Water Resources*, vol. 154, p. 103982, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0309170821001378

[19] Y. Huang, H. Jiang, W. feng Wang, W. Wang, and D. Sun, "Soil moisture content prediction model for tea plantations based on SVM optimised by the bald eagle search algorithm," *Cognitive Computation and Systems*, vol. 3, no. 4, pp. 351–360, Sep. 2021.

[20] R. Togneri, D. Felipe dos Santos, G. Camponogara, H. Nagano, G. Custódio, R. Prati, S. Fernandes, and C. Kamienski, "Soil moisture forecast for smart irrigation: The primetime for machine learning," *Expert Systems with Applications*, vol. 207, p. 117653, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417422009563

[21] J. Brinkhoff, J. Hornbuckle, and C. Ballester Lurbe, "Soil moisture forecasting for irrigation recommendation," *IFAC-PapersOnLine*, vol. 52, no. 30, pp. 385–390, 2019, 6th IFAC Conference on Sensing, Control and Automation Technologies for Agriculture AGRICONTROL 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2405896319325078

[22] Y.-A. Chen, W.-H. Hsieh, Y.-S. Ko, and N.-F. Huang, "An ensemble learning model for agricultural irrigation prediction," in *2021 International Conference on Information Networking (ICOIN)*, 2021, pp. 311–316.

[23] P. Kumar, A. Udayakumar, A. Anbarasa Kumar, K. Senthamarai Kannan, and N. Krishnan, "Multiparameter optimization system with dcnn in precision agriculture for advanced irrigation planning and scheduling based on soil moisture estimation," *Environmental Monitoring and Assessment*, vol. 195, no. 1, p. 13, Oct 2022. [Online]. Available: https://doi.org/10.1007/s10661-022-10529-3

[24] O. Rozenstein, L. Fine, N. Malachy, A. Richard, C. Pradalier, and J. Tanny, "Data-driven estimation of actual evapotranspiration to support irrigation management: Testing two novel methods based on an unoccupied aerial vehicle and an artificial neural network," *Agricultural Water Management*, vol. 283, p. 108317, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0378377423001828

[25] S. Hussain, M. Mubeen, W. Nasim, S. Fahad, M. Ali, M. A. Ehsan, and A. Raza, "Investigation of irrigation water requirement and evapotranspiration for water resource management in southern punjab, pakistan," *Sustainability*, vol. 15, no. 3, 2023. [Online]. Available: https://www.mdpi.com/2071-1050/15/3/1768

[26] S. S. Hemming, H. F. de Zwart, A. A. Elings, I. Righini, and A. A. Petropoulou, "Autonomous greenhouse challenge, first edition (2018)," 2019. [Online]. Available: https://data.4tu.nl/articles/dataset/Autonomous_Greenhouse_Challenge_First_Edition_2018_/12717758/1

[27] U. Ahmad, A. Alvino, and S. Marino, "Solar fertigation: A sustainable and smart iot-based irrigation and fertilization system for efficient water and nutrient management," *Agronomy*, vol. 12, no. 5, 2022. [Online]. Available: https://www.mdpi.com/2073-4395/12/5/1012

[28] E. Rahm and H. Do, "Data cleaning: Problems and current approaches," *IEEE Data Eng. Bull.*, vol. 23, pp. 3–13, 01 2000.

[29] F. E. Grubbs, "Procedures for detecting outlying observations in samples," *Technometrics*, vol. 11, no. 1, pp. 1–21, 1969. [Online]. Available: https://www.tandfonline.com/doi/abs/10.1080/00401706.1969.10490657

[30] V. Aggarwal, V. Gupta, P. Singh, K. Sharma, and S. Neetu, "Detection of spatial outlier by using improved z-score test," 04 2019, pp. 788–790.

[31] A. Geron, *Hands-on machine learning with scikit-learn, keras, and tensorflow: Concepts, tools, and techniques to build intelligent systems*, 2nd ed. O'Reilly Media, Inc., 2019.

[32] R. H. Shumway and D. S. Stoffer, *Time Series Analysis and Its Applications*. Springer New York, 2000.

[33] R. Mushtaq, "Augmented dickey fuller test," *SSRN Electronic Journal*, 2011.

[34] H. Wickham, *ggplot2*. Springer International Publishing, 2016.

[35] I. Syarif, A. Prugel-Bennett, and G. Wills, "Svm parameter optimization using grid search and genetic algorithm to improve classification performance," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 14, no. 4, p. 1502, Dec. 2016. [Online]. Available: http://dx.doi.org/10.12928/telkomnika.v14i4.3956

[36] P. I. Frazier, "A tutorial on bayesian optimization," 2018.

[37] A. Khamis, Z. I. ., K. H. ., and A. T. M. ., "The effects of outliers data on neural network performance," *Journal of Applied Sciences*, vol. 5, no. 8, pp. 1394–1398, Jul. 2005.

[38] E. Duesterwald, A. Murthi, G. Venkataraman, M. Sinn, and D. Vijaykeerthy, "Exploring the hyperparameter landscape of adversarial robustness," 2019.

[39] R. Novak, Y. Bahri, D. Abolafia, J. Pennington, and J. Sohl-Dickstein, "Sensitivity and generalization in neural networks: an empirical study," 02 2018.

Hamed Laouz is a PhD student in the computer science department at the university of Biskra (Algeria), specializing in artificial intelligence applications. He holds a Master's degree in artificial intelligence from university of Biskra. His research interests focus on integrating AI techniques in the smart-farming sector, particularly in irrigation tasks and greenhouse climate control.

Soheyb Ayad is an associate professor in the computer science Department at the university of Biskra (Algeria) and a member of LINFI laboratory. His research interests span various areas, including AI-enabled Networks, Internet of Things, 5G and Beyond Networks, SDN/NFV and smart systems applications. Additionally, he has authored several international publications in these fields.

Labib Sadek Terrissa is a full professor in computer science at Biskra University, Algeria. He is also a senior consultant in digitalization project management. After receiving an engineering degree in electronics, he received his Ph.D. in computer engineering in 2006 from Le Havre University, France. His current research interests include cloud computing, Machine learning, Smart maintenance, and Prognostic and Health Management.

Aicha Nabila Benharkat is Associate Professor at the Computer Science Department (INSA de LYON) since 1992. In this role, she worked on the Integration of heterogeneous databases using Description Logics. Currently, her research interests include the interoperability domain, schema matching techniques in small and large scale, business process as well as the interoperability in service oriented information system and quality and Web services discovery. Since 2013, her work has evolved into the cloud domain and particularly to the development of a cloud application development methodology, to the migration of applications in the cloud and finally to the protection of the privacy during all the lifecyle of data resulting from connected objects.

Samir Merdaci has a PhD on soil and water preservation in arid lands and his position is an assistant professor in the agronomy department at the University of El-Oued-Algeria. He was also the head of the experimental station at the Technical Institute for the Development of Saharan Agronomy in Biskra-Algeria. His research interests are the use of remote sensing and Geographic Information Systems GIS in agriculture and environment.