Intelligent System Forecasting the Temperature Distribution in Urban Regions

A Case Study Over Basra Airport, Iraq

Hiba J. Toama
Computer Science Department, College of Compute Science and Information Technology, Basrah University, Iraq
Prof. Dr. Karim Q. Hussein
Computer Science Department, College of Science, Mustansiriyah University, Iraq

Abstract: The vast amount of available solar energy makes it a serious source for electricity production. Approximately 30% of solar radiation is radiated back into space, while the land, atmosphere, and oceans absorb the rest. Therefore, it was important to focus on this sustainable resource by analyzing its quantities and distribution and predicting its potential, with the aim of exploiting it in modern technologies. Data for this study was collected over a 14-year period (2010–2024) from Basra Airport in southern Iraq, a year-round sunny area, with 122,734 reads. This research presents the use of five machine learning models: XGBoost, random forests, k-nearest neighbors (KNN), linear regression, and artificial neural networks (ANN), to predict temperature as a tool for assessing the sustainability of solar energy. The performance of the models was evaluated using R^2, MSE, and MAE metrics. The results showed that the XGBoost and Random Forest models outperformed the other models, achieving the highest performance with R^2 values of 0.974 and 0.973, respectively. Python libraries such as Pandas, Seaborn, Matplotlib, Scikit-learn, and TensorFlow were used, and the models were implemented in Google Colab, which provides flexible and modern tools for data analysis. This study aims to build an accurate predictive model to measure solar energy sustainability in southern Iraq and support smart decision-making in renewable energy management.

INTRODUCTION

The accelerating pace of urbanization coupled with the ever-increasing energy demands of cities has brought forth an urgent need for innovative and sustainable solutions. Smart cities, characterized by their integration of advanced technologies and intelligent urban planning, aim to enhance the

quality of life of their residents while minimizing their environmental footprint. In this pursuit, renewable energy sources have emerged as pivotal components, with solar power standing out as a beacon of clean and accessible energy[1]. Everyday earth receives sunlight above (1366W approx.) This is an unlimited source of energy which is available at no cost. The major benefit of solar energy over other conventional power generators is that the sunlight can be directly converted into solar energy with the use of smallest photovoltaic (PV) solar cells. the most advantages of solar energy are that it is free, reachable to common people and available in large quantities of supply compared to the price of various fossil fuels and oils in the past ten years. Moreover, solar energy requires considerably lower manpower expenses over conventional energy production technology[2].

Artificial intelligent (AI) techniques are rapidly opening up a new frontier in industry and business, it has the potential to revolutionize the way we discover, learn, live, communicate, and work. It has tremendous potential for the economy and society[3].ML methods are currently a hot spot in renewable energy prediction problems, with hundreds of new algorithmic proposals and real-world applications, including well-established artificial intelligence methods, hybridization with numerical methods, development of ensemble methods for prediction, or novel trends in the field such as deep learning, among others[4].

This research paper has the primary objective of this study to compare the performance of the linear regression, extreme gradient boosting (XGBoost), K-nearest neighbor (KNN), random forest regressor, and artificial neural network (ANN) models within a 14-year dataset context in order to predict the distribution of temperatures and their stability throughout the seasons of the year for the selected area to be used in generating electrical energy in sustainable, environmentally friendly ways and at the lowest costs.

This study aims to use artificial intelligence techniques to accurately predict temperatures, with the goal of assessing the potential and sustainability of solar energy in basrah southern of Iraq. By analyzing 14 years of climate data, advanced machine learning models are applied to extract accurate patterns that support effective energy management decisions.

The rest of this paper is structured as follows. In Section 2, we provide a review of related research. In Section 3, we describe the methodology that we use. In Section 4, we present and discuss the results of our work. In Section 5, we inserted the conclusion of our work. Finally, in Section 6, we close with concluding remarks and recommendations for future work.

RELATED WORK

In this section we have included the latest literature related to our topic for the last three years in order to keep up with the latest techniques and tools used.Vu Tran et al (2024)They evaluated the effectiveness of DNN-based domain adaptation for daily maximum temperature forecasting in experimental

low-resource settings. We used an attention-based transformer deep learning architecture as the core forecasting framework and used kernel mean matching (KMM) for domain adaptation. Domain adaptation significantly improved forecasting accuracy in most experimental settings, thereby mitigating domain differences between source and target regions. Specifically, we observed that domain adaptation is more effective than exclusively training on a mall amount of target-domain training data. This study reinforces the potential of using DNNs for temperature forecasting and underscores the benefits of domain adaptation using KMM. It also highlights the need for caution when using small amounts of target-domain data to avoid overfitting[5]. Pannee Suanpang et al (2024) They presented a study comparing two machine learning models, Light Gradient Boosting Machine (LGBM) and K-Nearest Neighbors (KNN), for solar power generation forecasting in microgrid applications. The analysis evaluates accuracy, reliability, training times, and memory usage. LGBM outperforms KNN with higher accuracy (R^2: 0.84 vs. 0.77), lower errors (RMSE: 5.77 vs. 6.93; MAE: 3.93 vs. 4.34), and robust handling of outliers. However, it requires longer training times (120s vs. 90s) and more memory (500MB vs. 300MB). These results highlight LGBM's suitability for optimizing energy management in microgrids, contributing to sustainable practices and efficient solar power forecasting[1].Fister, D et al (2023) They proposed three different computational frameworks for air temperature prediction: a Convolutional Neural Network (CNN) with video-to-image translation, several ML approaches including Lasso regression, Decision Trees, and Random Forest, and finally a CNN with a pre-processing step using Recurrence Plots, which convert time series into images. Using these frameworks, a very good prediction skill has been obtained in both the Paris and Córdoba regions, showing that the proposed approaches can be an excellent option for seasonal climate prediction problems[6]. Md. Atikur Rahman et al (2023) Used data analytics techniques and machine learning methods to predict the weather accurately. So, in this experiment, we propose a new knowledge-based system for weather prediction using KNN, SVM, NB, DT, RF, and LR for data modeling, and we got a maximum of 95.89% accuracy from the Gaussian Naive Bayes (GNB) algorithm. So, our plan is to develop weather prediction using the machine learning concept[7]. Yuan Yuan et al (2023) They construct a user's future location prediction model—dubbed the Loc-PredModel—by employing the Extreme Gradient Boosting (XGBoost) algorithm to forecast users' trip destinations and arrival times. Anchored in the anticipated outcomes of user travel behavior, personalized weather data reports are formulated. Experimental results underscore the Loc-PredModel's remarkable predictive prowess, demonstrating a root mean squared error (RMSE) value of 0.208 and a coefficient of determination (R2) value of 0.935, affirming its efficacy in prognosticating users' trip destinations and arrival times[8].Patil Malini et al. (2022) present Cauchy Particle Swarm Optimization (CPSO), a technique for finding the hyperparameters of the LSTM. The proposed technique minimizes the validation mean square error rate (MSER) to improve accuracy. They test the proposed

technique on 30-year Riyadh city temperature datasets. In experimental evaluation, the proposed CPSO-LSTM outperforms LSTM and grid-search LSTM by 50% and 55%, respectively[9].

METHODS AND MODELS

We mainly used five prediction algorithms for supervised machine learning (ML): linear regression, eXtreme Gradient Boosting (XGBoost), K-Nearest Neighbor (KNN), Random Forest Regressor, and artificial neural network (ANN). Before moving to the model training phase, we implemented thorough data preparation and feature engineering techniques. Each of these techniques has been covered in detail in the following sections. Figure 1 shows a flow diagram for the temperature prediction model. We followed nine different steps, which included data collection, data description, data processing, feature selection, data split, chosen algorithms, model tuning, model performance evaluation, and, finally, prediction.

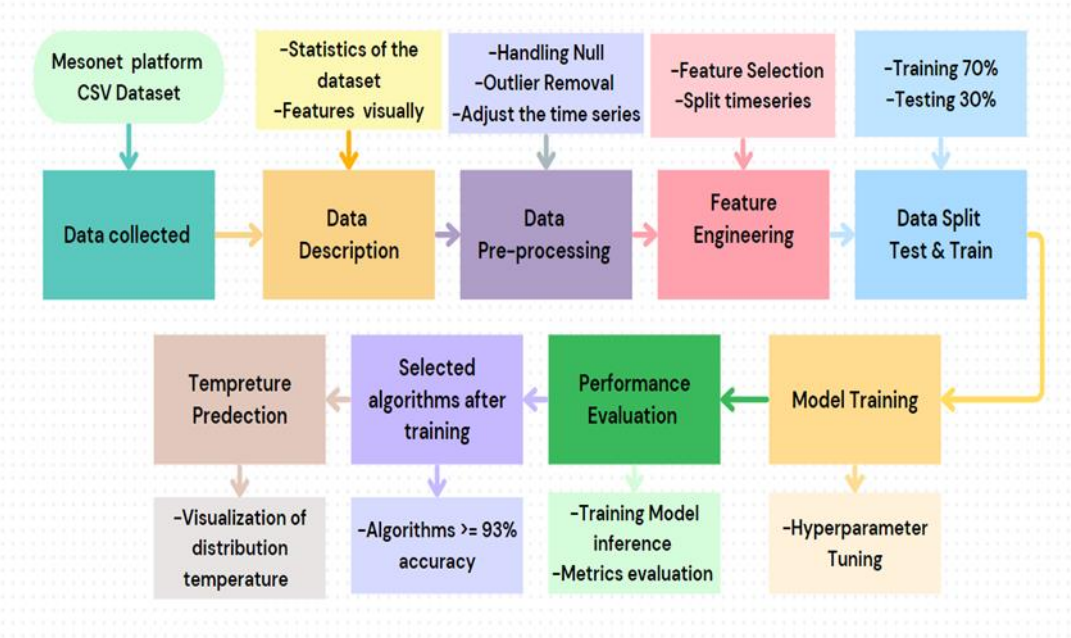

Figure 1 diagram for the temperature prediction model.

1. Dataset

This dataset contains observations from November 4, 2010, to November 4, 2024, captured from Basra Airport at an hourly rate Taken from a trusted platform called Mesonet (https://mesonet.agron.iastate.edu/request/download.phtml?network=IQ__ASOS)Iowa Environmental Mesonet (IEM) collects environmental data from cooperating members with observing networks[10]. The data are stored and made available on this website., and consists of 3 columns. The essence of the current study is to analyze and predict the temperature in Basra city in order to benefit from the continuity of solar radiation in order to exploit it in generating

electricity from clean resources, including temperature, humidity, and wind speed as auxiliary factors in order to predict the temperature behavior

The dataset can be classified into two main groups. The primary category includes atmospheric factors that directly affect temperature, such as humidity and wind speed. The second category includes details about the time and date of collection of each data sample, including hourly, weekly, monthly and annual time series for a period of 14 years, with a total of 122,734 readings.

2. Data Description.

This section includes a detailed description of each selected atmospheric factor, and it is also described statistically after analyzing the data obtained in the initial form in Table 1 below. Addition to Figure 2 represents a visual representation of the state of the selected initial parameters before reprocessing during the time series.

Table 1 Descriptive statistics of the dataset.

	Temperature	**Humidity**	**windspeed**
Count	122734.0	122734.0	122734.0
Mean	26.7268564537938	38.82610653934509	8.65200506787183
Std	11.074772658946578	25.137834811797905	8.476202978741734
Min	-2.0	0.23	0.0
25%	18.0	18.06	5.0
50%	26.66666667	32.43	8.0
75%	35.0	57.26	12.0
Max	62.0	100.0	1360.69

Date and time: Represent the date and time for each sample of the data set, with each sample changing every hour. The data was collected for 14 years, resulting in approximately 122,734 rows.

Temperature (Temp): is a measure of the average kinetic energy of particles in a substance or system. It quantifies how hot or cold an object or environment is and serves as a fundamental parameter influencing various physical, chemical, and biological processes. In meteorology and environmental studies, temperature is crucial for understanding weather patterns, atmospheric stability, and phenomena like evaporation, condensation, and precipitation. In chemistry, it governs reaction rates and equilibrium states.

1. Kelvin (K): The Kelvin scale is the SI unit for temperature. It starts at absolute zero, the point where all molecular motion theoretically ceases.

$$T_K = T_C + 273.15 \quad (1)$$

2. Celsius (°C): The Celsius scale is commonly used in most parts of the world and is defined relative to the freezing and boiling points of water at standard atmospheric pressure:

- Freezing point: 0◦
- Boiling point: 100°C

3. Fahrenheit (°F): The Fahrenheit scale is widely used in the United States. It relates to Celsius as:

$$T_c = 5/9 \times (T_f - 32) \quad (2)$$

To convert from Fahrenheit to Celsius:

$$T_f = (9/5) \times T_C + 32 \quad (3)$$

Relative Humidity (RH): is a measure of how saturated the air is with water vapor (moisture in its gaseous state). It is expressed as a percentage and reflects the ratio between the current amount of water vapor in the air and the maximum amount the air can hold at a given temperature if fully saturated.

$$RH = \frac{e}{e_s} X100 \quad (4)$$

Where:

- e: Partial pressure of water vapor in the air (actual water vapor present).
- e_s: Saturation vapor pressure at the same temperature (the maximum amount of water vapors the air can hold).

Wind speed: refers to the rate at which air moves horizontally through the Earth's atmosphere. It plays a significant role in weather patterns, climate systems, and environmental processes. It is typically measured in units like meters per second (m/s), kilometers per hour (km/h), or miles per hour (mph).

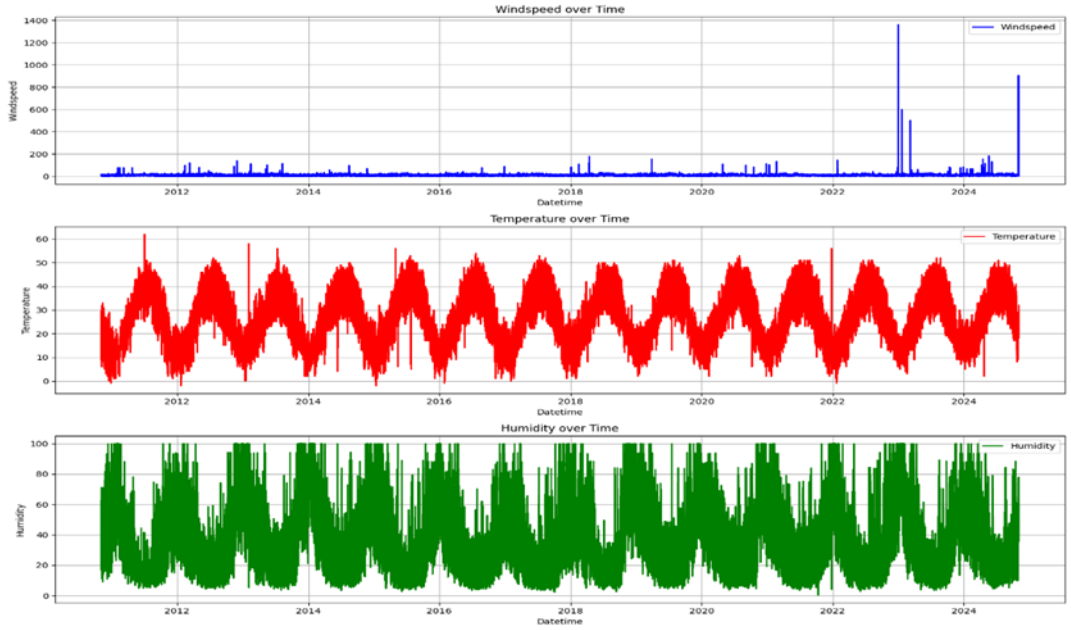

Figure 2 Visualize the features before cleaning during the time series.

3. Data preprocessing.

Data quality is an essential prerequisite for enhancing prediction models used. The process of data preprocessing has a crucial role in generalization capabilities, reducing noise, and thereby enhancing the speed and performance of AI algorithms, particularly when handling huge timeseries datasets. The techniques employed in data extraction and monitoring include feature selection, the removal of outliers, and the imputation of null values.

We used the Interquartile Range Rule (IQR) to remove outliers. This method is based on quartile values and is used to identify and remove values that are considered outliers based on the spread of the data. Outliers can mislead the learning process, resulting in a model that struggles to provide reliable and

accurate temperature index predictions, affecting the overall quality and reliability of the predictions. Figure 3 shows a box plot to show the outliers for each feature for the location. It is clear that most of the outliers come from the wind speed feature, around 2524 from 122734 values, and hence we removed outliers based on these features after removing the outlier that was not in the range of Q1 and Q3 represented by the features in the box plot given for the location.

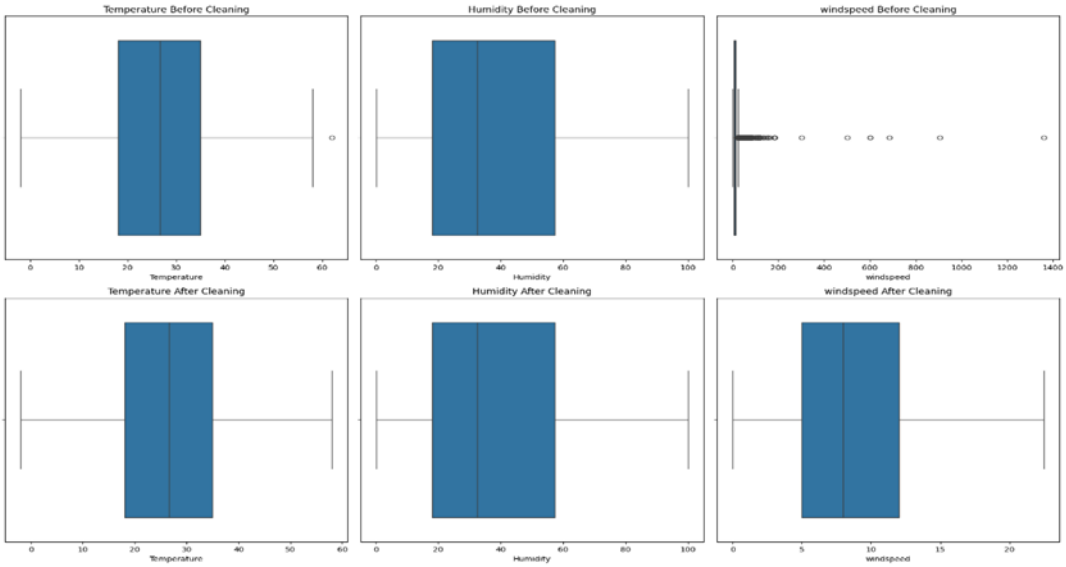

Figure 3 Boxplot before after Outlier Removal

We also filled in the missing (null) values or data after removing outliers by using the (linear interpolation) method, which is a mathematical method used to replace missing values within the range of known data points, assuming that the change between two adjacent points is linear. This method is widely applied in time series analysis.

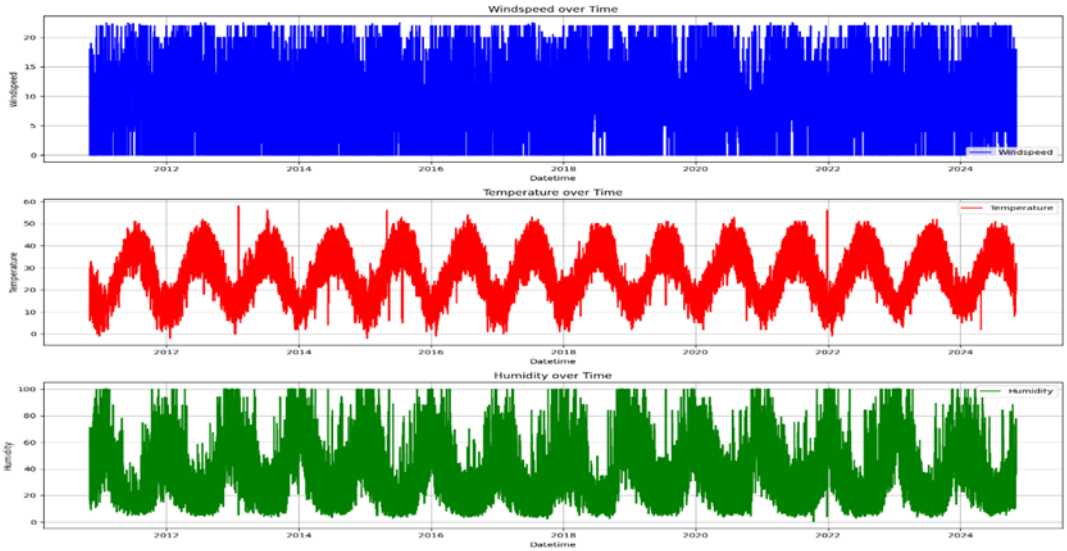

Figure 4 Visualize the features after cleaning during the time series.

Figure 4 above provides a view of the features after reprocessing the data by removing outliers and replacing them with linear interpolation. This preserves the time series and does not leave gaps, obtaining higher accuracy when training the model.

4. Feature Selection and Engineering.

Wind speed and humidity are important environmental factors that directly influence the perceived temperature. Humidity determines how we feel heat or cold, as higher humidity levels increase our perception of heat in hot weather, while in cold weather, it can make us feel colder. On the other hand, wind speed affects heat transfer; high wind speeds can enhance the feeling of cold in cooler weather through the "wind chill" effect, while in hot weather, wind can help reduce the perception of heat. Therefore, analyzing the relationship between these factors is crucial for improving temperature prediction accuracy.

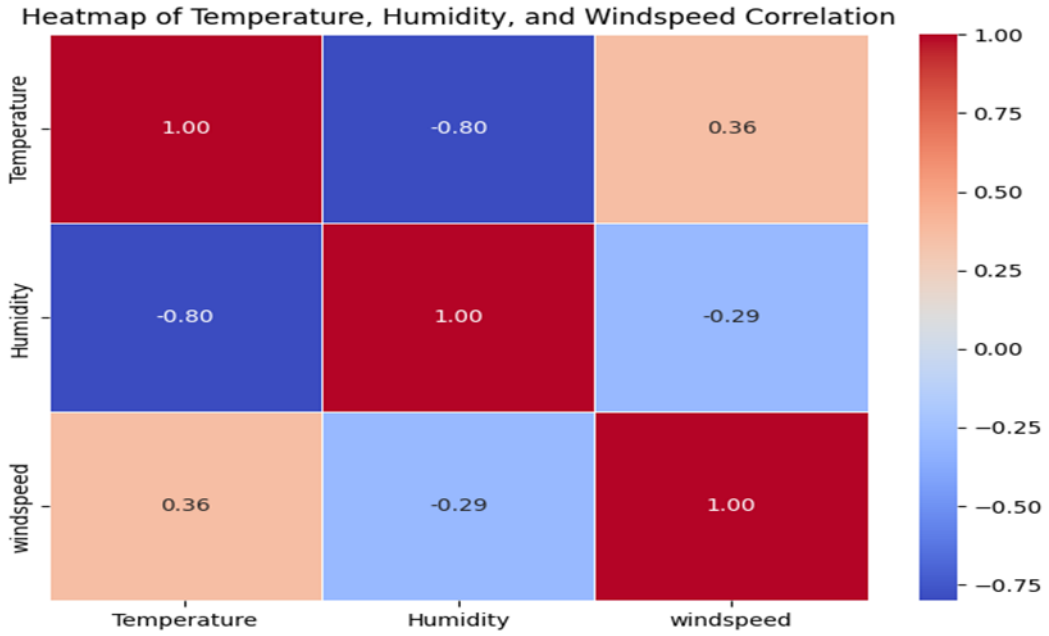

Figure 5 correlation between temperature, humidity, and wind speed.

The heatmap shows in figure 5 the correlation between temperature, humidity, and wind speed, which expresses the degree of correlation between each pair of variables. The relationship between temperature and humidity was (-0.80), indicating a strong inverse relationship; humidity levels decrease significantly with increasing temperatures. The relationship between temperature and wind speed was (0.36), reflecting a weak direct correlation; wind speed increases slightly with increasing temperatures. As for the relationship between humidity and wind speed, it was (-0.29), indicating a weak inverse correlation; humidity tends to decrease with increasing wind speed. These results indicate a clear effect of temperature on humidity, while its effect on wind speed is less clear.

Splitting temporal data into periods (hourly, daily, weekly, monthly, yearly) enhances understanding of seasonal and cyclical patterns of temperature over time, helping to improve the accuracy of predictive models. Daily changes, such as daytime peaks, weekly changes associated with human activity behaviors, and seasonal and annual patterns due to climate can be detected. This splitting allows models to deal with nonlinear relationships between time and temperature, capture long-term trends such as the impact of climate change, and provide more accurate short- and long-term forecasts.

5. Prediction and model training.

This part focuses on enhancing a predictive model to forecast temperature levels depending on several measured features. We aim to implement multiple machine learning algorithms to predict the temperature derived from specific features concentration ranges and time series.

To predict the temperature distribution, we employed supervised regression algorithms, including random forests, XGBoost, K-nearest neighbors, linear regression, and artificial neural networks. When developing a machine learning model, various architectural design options are typically presented.

The dataset is divided into training and testing sets to evaluate the model's performance. We used a 30–70 split, where 70% of the data are used for training the model and the remaining 30% are used for testing. This division ensures that the model is trained on a substantial portion of the data while being tested on an unseen subset to validate its generalization capability.

Determining the optimal model structure for a given model is not always easy, necessitating experimentation with different sub-parameters to explore a range of options. In our study, we focus on exploring and selecting the optimal model structure through sub-parameter tuning. This process involves tuning the parameters that make up the model structure to improve performance. We performed sub-parameter tuning using scikit-learn's Grid Search CV, iterating over a specific set of sub-parameters to identify those that give the best accuracy. The sub-parameters used after these tuning iterations, along with their values, are shown in Table 2.

Table 2. Hyperparameters used for each model.

Model	Hyperparameters
XGradientBoosting	{'colsample_bytree': 1, 'learning_rate': 0.2, 'max_depth': 7, 'n_estimators': 200, 'subsample': 0.8}
Random Forest	{'max_depth': 30, 'min_samples_split': 2, 'n_estimators': 200}
K-Nearest Neighbors	{'metric': 'manhattan', 'n_neighbors': 3, 'weights': 'distance'}
Linear Regression	{'fit_intercept': True}
ANN	{'hidden_layer_sizes'=256,128,64, 'activation'input: tanh, 'activation'output: linear,'Dropout':0.3,'optimizer'=adam,'Alpha':0.001,'batch_size':64,'epochs: 100, validation_split=0.2}

1-Linear regression is a widely used statistical method for modeling the relationship between a dependent variable (outcome) and one or more independent variables (predictors). It works by fitting a line (or plane in the case of multiple predictors) that minimizes the difference between predicted and actual values[11].

Linear regression predicts the dependent variable Y based on the independent variable(s) X using the equation:

$$Y=\beta 0+\beta 1X+\epsilon \quad (5)$$

2-Random Forest Regressor is an ensemble learning technique based on decision trees, designed for regression tasks. It operates by constructing multiple decision trees during training and averaging their outputs to improve predictive accuracy and control overfitting. This approach enhances the model's robustness by leveraging the "wisdom of the crowd," where the collective output of multiple weak models (decision trees) produces stronger and more reliable predictions[12]. The prediction for a Random Forest Regressor is given by:

$$y^{\wedge} = \frac{1}{N}\sum_{i=1}^{N} T_i(x) \quad (6)$$

Where:

N: Number of trees in the forest.

T_i (x): Prediction of the i-th tree for input x.

3-K-Nearest Neighbors (KNN) is a simple yet effective algorithm used for classification and regression tasks. It works by identifying the k closest data points to a given query point and makes predictions based on the majority class (for classification) or the average of their values (for regression). KNN is a non-parametric method, meaning it makes no assumptions about the underlying data distribution[13].

For regression, the prediction is made by averaging the target values of the k nearest neighbors:

$$y^{\wedge}(x) = \frac{1}{k}\sum_{i=1}^{k} y_i \quad (7)$$

4-XGBoost (Extreme Gradient Boosting) is a powerful and efficient implementation of gradient boosting that has gained significant popularity in machine learning, particularly for structured/tabular data. It builds an ensemble of decision trees sequentially, where each new tree tries to correct the errors made by the previous trees. XGBoost uses an advanced form of gradient boosting that includes regularization to reduce overfitting, making it highly effective for various tasks, including classification, regression, and ranking[14].

5-Artificial Neural Networks (ANN) are computational models inspired by the structure and functioning of biological neural networks in the human brain.

ANNs consist of interconnected nodes (neurons) organized into layers: an input layer, one or more hidden layers, and an output layer. Each connection between neurons has a weight, and each neuron applies an activation function to the weighted sum of its inputs to determine its output. These networks are capable of learning complex patterns and relationships from data, making them powerful tools for tasks like image recognition, natural language processing, and predictive analytics[15].

The output of a neuron is calculated as:

$$yi = f\left(\sum_{j=1}^{m} w\,ij\,x_j + b_i\right) \quad (8)$$

Where:

W i : Weights of the inputs.

x i: Inputs to the neuron.

b: Bias term.

f: Activation function (e.g., ReLU, Sigmoid).

Performance Evaluation Metrics.

In this section, we evaluated the model using three metrics, which are shown below.

Mean Squared Error (MSE) is a cost function that calculates the average of the squares of the errors, the average squared difference between the estimated and actual values. is a commonly used metric to measure the accuracy of a model. It evaluates how close a model's predictions are to the true data points. Lower MSE values indicate a model that predicts values closer to actual observations[16].

Mathematical Formula:

$$\mathbf{MSE} = \sum_{i=1}^{n}(\mathbf{actual} - \mathbf{forcat})\text{^}\mathbf{2} \quad (9)$$

Mean Absolute Error (MAE) is a metric that calculates the average magnitude of the absolute errors between the predicted and actual values. characterizes the alteration among the original and predictable values and is mined as the dataset's total alteration mean[17].

$$\mathbf{MAE} = \frac{1}{n}\sum_{i=1}^{n}|\mathbf{actual} - \mathbf{forcat}| \quad (10)$$

R-squared (R^2) is a statistical metric used to assess how well a regression model explains the variability of the target variable. It is also called the coefficient of determination and ranges between 0 and 1[18].

The formula for calculating R2 is:

$$\mathbf{R2} = 1 - \frac{\mathrm{SStot}}{\mathrm{SSres}} \quad (11)$$

RESULTS AND DISCUSSION

The results of this study demonstrate the performance of several machine learning models in predicting temperature, as evaluated through the metrics of

R^2, MSE, and MAE. Models such as XGBRegressor, Random Forest Regressor, KNeighborsRegressor and Artificial Neural Network all achieved high accuracy, surpassing the required R^2 threshold of 93%.XGBRegressor was the top performer, yielding an R^2 value of 0.973, an MSE value of 3.22, and an MAE value of 1.33.The Random Forest model showed accurate performance, scoring high on various evaluation metrics, achieving an MSE value of 3.35, an MAE of 1.28, and an R^2 of 0.972, placing it second in terms of prediction accuracy among the models used. K Neighbors Regressor also demonstrated strong performance with an R^2 of 0.934, MSE of 8.19, and MAE of 1.869, which is still within the acceptable range. As for the artificial neural network model, it exceeded the acceptance threshold by a critical percentage, as it recorded values for the evaluation metrics of 8.49 for MSE and 2.26 for MAE, while the (R^2) reached 0.931.In contrast, linear regression did not meet the desired accuracy, producing a significantly lower R^2 of 0.684, MSE = 40.14, and MAE = 4.92. These findings suggest tree-based models, especially XGBRegressor and Random Forest, are most suitable for the temperature forecasting tasks in this study, while ANNs require more data to understand the forecast pattern. The data used were classified within the small to medium size range, while the ANN model requires medium to large size data, which is observed by the increasing accuracy of the model as the data size increases. On the other hand, it is recommended to exclude the use of a linear regression model when dealing with this type of data to ensure achieving the required accuracy for reliable forecasts. A summary of the results is presented in Table 3 below

Table 3 Performance Metrics Results for Each Model

Model	**MSE**	**MAE**	**R^2**	**Acceptance**
XGBRegressor	3.22	1.33	0.973	Accepted
RandomForest Regressor	3.35	1.28	0.972	Accepted
KNeighborsRegressor	8.12	1.869	0.934	Accepted
ArtificialNeural Network (ANN)	8.49	2.26	0.931	Accepted
Linear Regression	38.88	4.92	0.684	Not Accepted

These results emphasize that tree-based models are more effective for temperature prediction, aligning with the study's objectives of achieving high-accuracy predictions for climate modeling.

In this study, we used data from a real-time platform, which means that optimization techniques such as artificial oversampling or data augmentation or methods such as SMOTE for resampling cannot be applied. Therefore, we must rely on algorithms that handle real-world datasets with the highest accuracy.

The algorithm below illustrates the process of predicting temperature distribution using different machine-learning models. It starts by dividing the dataset into training (70%) and test (30%) sets. Five models: linear regression, eXtreme Gradient Boosting (XGBoost), K-Nearest Neighbor (KNN), Random Forest Regressor, and artificial neural network (ANN). For each model,

hyperparameters are tuned, followed by training on the training data. Predictions are then made on the test data. The algorithm evaluates each model by calculating an MSE, MAE, and R^2 to select algorithms that achieve accuracy higher than 93% to display their results graphically.

Algorithm 1: Algorithm for prediction of AQI_Class.

Require: Training and testing datasets with a ratio of 70:30
Ensure: MSE, MAE and R^2 for each model
1:Models←[Linear Regression(),K-Nearest Neighbor (),Random Forest(),XGBoost (), ANN]
2: for $i \leftarrow 0$ to len(Models) **do**
3: model ← Models[i]
4: HYPERPARAMETERTUNING(model)
5: model.fit(training_data)
6: predictions ← model.predict(testing_data)
7: MSE, MAE and R^2 ← EVALUATE(model, predictions)
8: **Output** → DISPLAY(MSE, MAE and R^2,predect)
9: end for
10: **function** HYPERPARAMETERTUNING(model)
11: Perform hyperparameter tuning for the given model
12: **end function**
13: **function** EVALUATE(model, predictions, evaluation)
14: Compute MSE, MAE and R^2
15: **Return** MSE, MAE and R^2
16**: end function**
17: **function** DISPLAY(MSE, MAE and R^2)
18: Display the evaluation metrics
19: **end function**

20**: function** DISPLAY(Comparison of Actual and Predicted**)**
21**:** Display the Predicted
19: **end function**

The (https://mesonet.agron.iastate.edu/) platform, as the database, was used to predict the distribution values of temperatures based on the concentration of different auxiliary meteorological parameters during the time series at a specific location. The dataset, consisting of over 122,734 data points collected over 14 years each hour, was first divided into training (70%) and testing (30%) subsets to facilitate model evaluation. The main libraries utilized in this study include Pandas for data manipulation, Seaborn and Matplotlib for visualizations, Date Time for time-series management, and Scikit-learn and TensorFlow for model development and training. The Google Colab environment provided the resources needed to train the model and evaluate performance, including the use of cloud-based GPU capabilities.

The graphs in Figure. 6 show a comparison of actual and predicted temperature values over time using five models: The results show that both Random Forest and XGBoost achieved a strong match between actual and predicted values, indicating their high accuracy in predicting seasonal values

across years. The K-nearest neighbors model showed an acceptable match but was relatively less accurate, with some discrepancies at certain points in time. The artificial neural network (ANN) model performed less accurately than its peers above, with a suboptimal match between actual and predicted values. In contrast, the linear regression model performed poorly, with large discrepancies between actual and predicted values, reflecting its limitations in capturing nonlinear and cyclical patterns. These results reflect the ability of the models to handle cyclical patterns, with performance varying between different models.

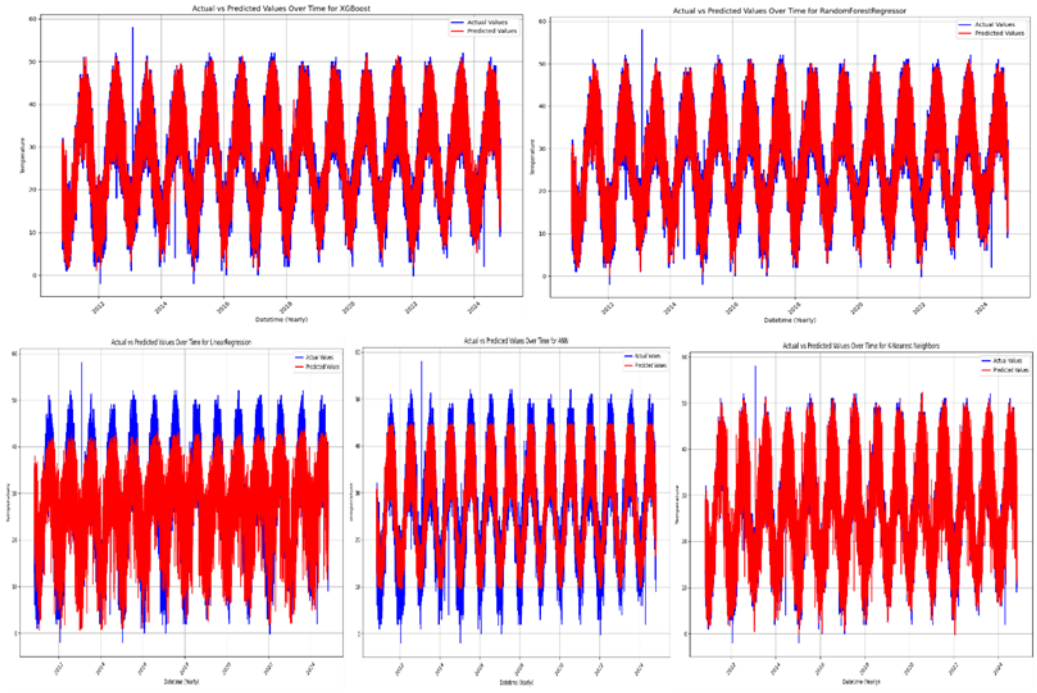

Figure 6 Comparison of Actual and Predicted Temperature Values Over Time

CONCLUSION

Temperature prediction is one of the most important fields due to its potential application in real-world problems, one of the most important of which is its use in renewable energy production applications, the most important of which is the production of electrical energy using solar cells. The proposed approach is based on a time-series dataset of temperature and using additional factors such as humidity and wind speed, which were collected from Basra Airport in Basra Governorate, southern Iraq, during the period from 2010 to 2024. In the proposed research on temperature distribution prediction, we evaluated five different models: XGBRegressor, RandomForest Regressor, KNeighborsRegressor, Artificial Neural Network (ANN), and Linear Regression, which we list in order of their efficiency, respectively. Among these models, XGBRegressor achieved the best performance, closely followed by Random

Forest Regressor.Both models were accepted based on their superior performance metrics as required by our study, which exceeded 93% accuracy.The KNeighborsRegressor algorithm and the artificial neural network showed critical acceptance in the study, reflecting the need to improve their performance to achieve more accurate prediction results.Linear regression yielded a higher error rate and R^2 scores of less than 0.674, leading to their rejection.Considering the error metrics and R^2 values, XGBRegressor emerged as the most accurate model for predicting the temperature distribution in this study.

RECOMMENDATIONS FOR FUTURE WORK

Firstly, we recommend using time series algorithms like ARIMA or LSTM in RNN to generalize over the longest possible period by increasing the data to millions of data instead of hundreds of thousands because it requires data that is classified as medium to huge to increase the accuracy of the prediction.

-Secondly, We also recommend increasing the time series to reach tens of years instead of the 14 years whose data we relied on to give space for deep learning algorithms to understand the data pattern.

-Finally, we recommend increasing the comprehensiveness of the areas targeted in the study to include cold, moderate, and hot areas, to expand coverage of the topic of sustainability and alternative energy.

ACKNOWLEDGEMENT

The authors expose their thanks for Basrah University for the support and encouragement to accomplish this research.

REFERENCES

1. P. Suanpang and P. Jamjuntr, “Machine Learning Models for Solar Power Generation Forecasting in Microgrid Application Implications for Smart Cities,” Sustainability (Switzerland), vol. 16, no. 14, Jul. 2024, doi: 10.3390/su16146087.
2. Mohd. R. S. Shaikh, “A Review Paper on Electricity Generation from Solar Energy,” Int J Res Appl Sci Eng Technol, vol. V, no. IX, 2017, doi: 10.22214/ijraset.2017.9272.
3. M. A. Goralski and T. K. Tan, “Artificial intelligence and sustainable development,” International Journal of Management Education, vol. 18, no. 1, 2020, doi: 10.1016/j.ijme.2019.100330.
4. J. Del Ser et al., “Randomization-based machine learning in renewable energy prediction problems: Critical literature review, new results and perspectives,” Appl Soft Comput, vol. 118, p. 108526, Mar. 2022, doi: 10.1016/J.ASOC.2022.108526.
5. V. Tran, F. Septier, D. Murakami, and T. Matsui, “Spatial–Temporal Temperature Forecasting Using Deep-Neural-Network-Based Domain Adaptation,” Atmosphere (Basel), vol. 15, no. 1, 2024, doi: 10.3390/atmos15010090.

6. D. Fister, J. Pérez-Aracil, C. Peláez-Rodríguez, J. Del Ser, and S. Salcedo-Sanz, “Accurate long-term air temperature prediction with Machine Learning models and data reduction techniques,” Appl Soft Comput, vol. 136, 2023, doi: 10.1016/j.asoc.2023.110118.
7. M. A. Rahman, O. Nafiz Akbar, and M. Assaduzzaman, “Applied Weather Forecasting using Machine Learning Approach,” in 2023 26th International Conference on Computer and Information Technology, ICCIT 2023, 2023. doi: 10.1109/ICCIT60459.2023.10441392.
8. [8] Y. Yuan et al., “Research and Application of Intelligent Weather Push Model Based on Travel Forecast and 5G Message,” Atmosphere (Basel), vol. 14, no. 11, 2023, doi: 10.3390/atmos14111658.
9. P. Malini and B. Qureshi, “A Deep Learning Framework for Temperature Forecasting,” in Proceedings - 2022 7th International Conference on Data Science and Machine Learning Applications, CDMA 2022, 2022. doi: 10.1109/CDMA54072.2022.00016.
10. “IEM :: Download ASOS/AWOS/METAR Data.” Accessed: Nov. 25, 2024. [Online]. Available: https://mesonet.agron.iastate.edu/request/download.phtml?network=IQ__ASOS
11. “Linear Regression with Python Implementation - Analytics Vidhya.” Accessed: Dec. 08, 2024. [Online]. Available: https://www.analyticsvidhya.com/blog/2022/02/linear-regression-with-python-implementation/
12. “RandomForestRegressor — scikit-learn 1.5.2 documentation.” Accessed: Dec. 08, 2024. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html
13. “1.6. Nearest Neighbors — scikit-learn 1.5.2 documentation.” Accessed: Dec. 08, 2024. [Online]. Available: https://scikit-learn.org/1.5/modules/neighbors.html
14. “XGBoost Documentation — xgboost 2.1.1 documentation.” Accessed: Dec. 08, 2024. [Online]. Available: https://xgboost.readthedocs.io/en/stable/
15. [Y. Wu, “Introduction to Neural Network Algorithm.”
16. T. O. Hodson, T. M. Over, and S. S. Foks, “Mean Squared Error, Deconstructed,” J Adv Model Earth Syst, vol. 13, no. 12, 2021, doi: 10.1029/2021MS002681.
17. A. S. Rajawat, O. Mohammed, R. N. Shaw, and A. Ghosh, “Renewable energy system for industrial internet of things model using fusion-AI,” Applications of AI and IOT in Renewable Energy, pp. 107–128, Jan. 2022, doi: 10.1016/B978-0-323-91699-8.00006-1.
18. J. Gao, “ R-Squared (R 2) – How much variation is explained? ,” Research Methods in Medicine & Health Sciences, vol. 5, no. 4, pp. 104–109, Sep. 2024, doi: 10.1177/26320843231186398.
19. Waleed Saleh Hamed1 , Dr. Karim Q. Hussein , “Using Deep Learning Techniques to Predict Wind Speed “ , International Journal of Computer Science and Mobile Applications , Vol.10 Issue. 10, October- 2022, pg. 1-15 ISSN: 2321-8363