AI in Solar Pond

Project report submitted to
Visvesvaraya National Institute of Technology, Nagpur
in partial fulfillment of the requirements for the award of
the degree

Bachelor of Technology
In
Mechanical Engineering
by

Atharva Kokate BT18MEC069

Bhavin Yardi BT18MEC096

Rohit R.Menon BT18MEC133

under the guidance of

Dr.Trushar B.Gohil

Department of Mechanical Engineering
Visvesvaraya National Institute of Technology
Nagpur 440 010 (India)

April 2022

AI in Solar Pond

Project report submitted to
Visvesvaraya National Institute of Technology, Nagpur
in partial fulfillment of the requirements for the award of
the degree

Bachelor of Technology
In
Mechanical Engineering

by

Atharva Kokate BT18MEC069

Bhavin Yardi BT18MEC096

Rohit R.Menon BT18MEC133

under the guidance of

Dr.Trushar B.Gohil

Department of Mechanical Engineering
Visvesvaraya National Institute of Technology
Nagpur 440 010 (India)
April 2022

Department of Mechanical Engineering
Visvesvaraya National Institute of Technology, Nagpur

Declaration

We, **Atharva Kokate (BT18MEC069)**, **Bhavin Yardi (BT18MEC096)** and **Rohit R.Menon (BT18MEC133)**, hereby declare that this project work titled “AI in Solar Pond” is carried out by us in the Department of Mechanical Engineering of Visvesvaraya National Institute of Technology, Nagpur. The work is original and has not been submitted earlier in whole or in part for the award of any degree/diploma at this or any other Institution / University.

Enrollment No	Name	Signature
BT18MEC069	Atharva Kokate	
BT18MEC096	Bhavin Yardi	
BT18MEC133	Rohit R. Menon	

Date:

Certificate

This is to certify that the project titled "AI in Solar Pond", submitted by **Atharva Kokate (BT18MEC069)**, **Bhavin Yardi (BT18MEC096)** and **Rohit R.Menon (BT18MEC133)** in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology in Mechanical Engineering**, VNIT Nagpur. The work is comprehensive, complete, and fit for final evaluation.

Dr.Trushar B.Gohil

Assistant Professor,
Department of Mechanical Engineering,
VNIT, Nagpur

Head, Department of Mechanical Engineering
VNIT, Nagpur
Date:

ACKNOWLEDGEMENT

Any significant task ever achieved is the cumulation of one’s efforts enhanced through their knowledge, experiences, and efforts. This dissertation is as such.

We express our sincere gratitude to our project guide Dr.Trushar B.Gohil, Assistant Professor, Department of Mechanical Engineering, VNIT, Nagpur for his invaluable guidance throughout the duration of our project. It is due to his support, close supervision, suggestions, and continuous encouragement that we were able to cover the various aspects of this dissertation. We would like to express our sincere thanks to our Head of the Department, Mechanical Engineering, VNIT Nagpur, for his comprehensive support and cooperation during the whole research work.

This dissertation would not have been possible without the information that was made available to us by various researchers through their publications, which acted as the foundation for our dissertation. We are grateful to everyone who made this dissertation possible.

Atharva Kokate BT18MEC069
Bhavin Yardi BT18MEC096
Rohit R.Menon BT18MEC133

ABSTRACT

Solar Ponds are reservoirs that provide a method to capture the solar energy and use it for useful work ranging from power generation to hot water supply in buildings. The solar pond is characterized by three zones - Upper Convective Zone (UCZ), Non-Convective Zone(NCZ) & Lower Convective Zone (LCZ). The salinity gradient in the pond prevents convective losses and traps the heat in the LCZ. The purpose of the study is to predict the temperatures of LCZ and UCZ of a salt gradient solar pond using Random Forest Regression and Deep Neural Network (DNN) algorithms, over a period of three years, using meteorological data.

The input parameters included 10 different meteorological and design parameters of the solar pond and the output parameters obtained are the temperatures of LCZ and UCZ for 3 years. This study gives a reliable and convenient method to determine the temperature values of LCZ and UCZ saving computational time.

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

UCZ - Upper Convective Zone

NCZ - Non-Convective Zone

LCZ - Lower Convective Zone

ML - Machine Learning

DNN-Deep Neural Network

ANN - Artificial Neural Network is a collection of nodes or neurons that facilitates the transfer of data and processes it.

MRE - Mean Relative Error is the mean of the absolute difference between experimental and predicted values.

R-squared - A metric that shows how suitable an algorithm is for a particular dataset.

Pamb - Partial pressure of water vapour at ambient temperature, mm of Hg

Patm - atmospheric pressure, mm of Hg

Pucz - vapor pressure of water at temperature of UCZ, mm of Hg

Qcond - rate of conduction heat transfer, W

Qconv - rate of convection heat transfer, W

Qevap - rate of evaporation heat loss from the surface of solar pond, W

Qg - rate of ground heat loss from LCZ, W

Qload - rate of heat extraction from LCZ, W

Qlosses - rate of heat loss, W

Qrad - rate of radiation heat loss from the surface of solar pond, W

Qs - rate of solar radiation absorbed in a layer, W

Subscripts

out-output

in-input

inst-instantaneous

LCZ-lower convective zone

NCZ- non-convective zone

UCZ - upper-convective zone

INDEX

CHAPTERS **PAGE NO.**

CHAPTER I

INTRODUCTION

1.1 Overview

1.1.1 Solar Pond

Salt Gradient Solar Ponds or simply solar ponds are a reservoir of salt water that has the capacity to store thermal energy. The water is made salty in such a way that the salinity gradient increases with an increase in depth, and below a certain depth, the pond has a uniform high salt concentration. The thermal energy could be stored for up to two days and utilized for a variety of purposes such as power generation, water desalination, refrigeration, etc.

Upper Convective Zone (UCZ), Non-Convective Zone (NCZ), and Lower Convective Zone (LCZ) are the three zones that make up a solar pond (LCZ). The UCZ is the solar pond's highest and thinnest layer, containing practically all freshwater. With respect to the depth of the solar pond, the NCZ has an increase in salt concentration and density. It serves as a barrier between the UCZ and the LCZ. The LCZ stores thermal energy and has the highest salt concentration and density of the solar pond.

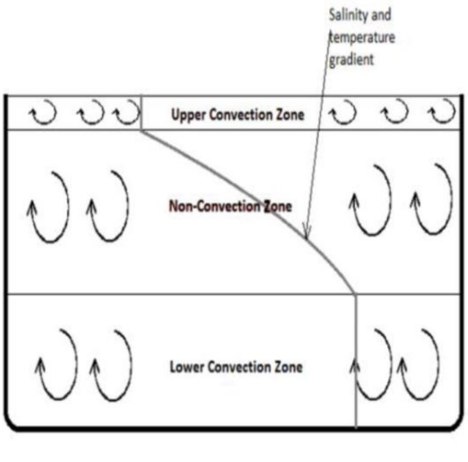


Figure - Layers of a Solar Pond. Reprinted from Singh, Baljit & Saoud, Altenaijy & Remeli, Muhammad & Ding, Lai Chet & Date, Abhijit & Akbarzadeh, Aliakbar. (2015).

1.1.2 AI

Artificial Intelligence (AI) involves the simulation of human intelligence by machines. An AI system feeds on the dataset that is fed to it and looks for patterns and trends and develops the ability to make predictions. The growth of AI in the last decade has been monumental and has ensured it is incorporated in almost every field. The fact that an AI model can produce near-perfect results with highly reduced computational time gives it the cutting edge over conventional methods. AI technology has already become an indispensable part of Autonomous Vehicles, AI-powered Assistants, Personalized Assistants, and so on.

The dataset fed to the AI model is to be distinguished into training and test datasets. The training is where the AI learns the patterns and features of the data. Then the test data is to provide a real-world check of how the algorithm will perform on unseen data. Every time an AI system finishes a round of data processing, it measures its own performance and reinforces its own expertise.

The two commonly associated terms with AI are Machine Learning (ML) and Deep Learning. ML is a subset of AI where systems learn from data using algorithms to perform a task without being explicitly programmed. The force that drives ML is statistics and learning is based on patterns and interference. Deep Learning is a subset of ML and deploys a brain-like logical structure of algorithms to facilitate the learning process. Deep learning is highly sophisticated and mathematically convoluted and has been recently gaining immense success in a variety of fields. It consists of the brain-like layered structure of algorithms called neural networks.

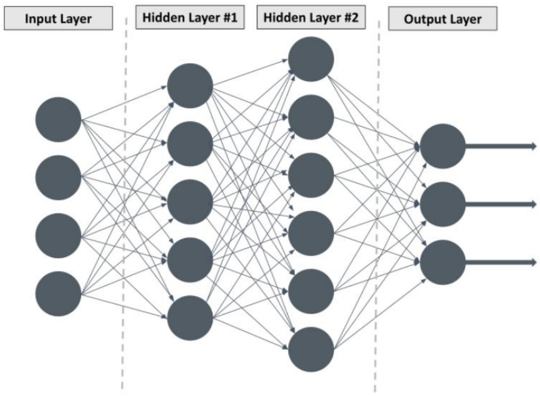


Figure-2 A Simple Neural Network.Reprinted from Baek, Jieun & Choi, Yosoon. (2020).

The learning of an AI model takes place through one of the three- supervised, unsupervised, or reinforcement learning. In supervised learning, the AI model is fed with labeled data and the algorithm finds a cause-effect relationship between variables and establishes a connection between input and output. Unsupervised learning is where the AI model is fed unlabeled data reducing human effort into the dataset. Unlike supervised learning, unsupervised learning algorithms adapt to dynamically changing hidden structures, and relationships between data are perceived in an abstract manner. Reinforcement learning in simple words trial-and-error. It features an algorithm that improves and learns from its own mistakes and ‘reinforces’ the learning curve.

1.2 Aim & Objectives

Solar Ponds remain an untapped section in the list of methods that harness solar energy. Thermal energy has been generated in this manner in many different parts of the world for many decades now, but none sustaining. The main cause for this is the low conversion efficiency of the solar ponds and the effort required to maintain the composition of the pond. Some of the solar ponds across the globe include

Table 1 Solar Ponds across the World

Location	**Latitude**	**Longitude**
El Paso, Texas	31.7619° N	106.4850° W
Atacama Desert, Chile	23.8634° S	69.1328° W
Royal Melbourne University, Australia	37.8136° S	144.9631° E
Bhuj, Gujarat	23.2420° N	69.6669° E
Beit HaArava, Israel	31.8078°N	35.4764° E
Kuwait City	29.3759° N	47.9774° E

Granada, Spain	37.1773° N,	3.5986° W
Makkah, Saudi Arabia	21.3891° N,	39.8579° E
Margherita di Savoia, Italy	41.399° N	16.0593° E

The aim of this dissertation is to develop an AI solver that can predict the temperatures of the LCZ and UCZ of a solar pond across the globe, for a period of three years. This dissertation makes use of the Random Forest Regressor Algorithm and Deep Neural Network (DNN) to train the dataset.

The objectives of this study are as follows:

- Generate a dataset using the meteorological and geometric data at different locations to feed the AI model.
- Identify suitable AI algorithms that can give good accuracy for the generated dataset.
- Train the AI model and tune the model to give improved accuracy and loss metrics.
- Plot graphs comparing the prediction and the measured data at each location to validate the accuracies of the AI models.

CHAPTER II
LITERATURE SURVEY

2.1 ANN and Solar Pond

2007 Kurt Artificial Neural Network Solar Pond [1] uses ANN to determine the density and temperature profiles of a SGSP. The study considered factors such as solar pond depth, ambient temperature, the radiation absorption coefficient of salty solution in the pond, initial density values of the pond, and time of day as the input parameters for the ANN model and generated the necessary profiles as the output. However, the dataset used consisted of 168 data points, far too less in comparison with the dataset used in this dissertation. The study obtained negligible MRE and the R-squared value obtained for the profiles was close to 1.

2.2 Predicting Solar Pond Parameters

A variety of studies have been conducted to develop mathematical models that predict various parameters of a Solar Pond. However, it is to be noted that studies that tap into AI for the prediction of solar pond parameters are scarce.

2021 Platikanov Spain [2] focuses on analyses of the operational parameters of the solar pond. The study focuses on determining the salt concentration and temperature changes during the operational season and checking stability. 2*018 Kumar Inverse Prediction and Optimization Analysis* [3] deals with a solar pond coupled with Thermoelectric Generator. The study determines the thickness and temperature of each zone of the solar pond and the power output from a thermo-electric generator.

2019 Verma Wall Profile [4] generates a model to evaluate the pond wall profiles. It delves into developing a mathematical model for analyzing various wall profiles of the fixed thickness of each layer and calculating the corresponding efficiency—optimization analysis of a given pond volume and area. The study concludes that the vertical wall profile provides the best performance.

2017 Abdullah [5] focuses on a mathematical model to determine the UCZ wall thickness. For a particular LCZ operating temperature, solar ponds with thicker UCZs have shorter recovery times (and so are more stable). A salt-gradient solar pond is least stable during the winter months, according to the proposed operating protocol, which may appear counter-intuitive at first.

2.3 Heat Extraction and Storage

2005 Andrew Heat Extraction [6] provides an alternate heat extraction method. The research covers Extracting Heat from the Gradient Zone(NCZ) and the Lower Convective Zone increases the overall energy efficiency of collecting solar radiation, storing heat, and delivering this heat to an application. *2017 Aramesh Transient* [7] predicts the thermal behavior of a solar pond during extraction using FEM. Temperature profiles were discovered inside the LCZ and the tube. For the months of August, September, and November, heat extraction rates were estimated. In August, the temperatures, stored energy, and extracted energy were at their peak, and in November, they were at their lowest. In November, though, the steady-state was easily achieved.

2006 Jaefarazadeh, Heat Extraction [8] contrasts summer and winter heat extraction. The temperature fluctuations of the storage zones, surface zone, ambient, inlet, and output of the internal heat exchanger were studied on an hourly and daily basis. The study concludes that in the transitional stage, the pond can provide heat with a relatively high thermal efficiency for a limited time. This approach can also be used for long periods of time at a low efficiency.

2.4 Predictions using AI Algorithms

This dissertation deals with using Random Forest Regression and the DNN algorithm to make predictions of the solar pond parameters. So studies using the RF and DNN algorithm to make predictions were analyzed during reviewing the literature.

Khaki S and Wang L (2019) [9] make use of the DNN algorithm to make predictions of crop yield. The root-mean-square-error on the dataset is 12% of the average yield and the study clearly underlines how DNN computationally outperforms other popular methods. Finally, the study concludes how environmental factors have a bigger effect on crop yield than genotype. *Shuhui Li (2003)* [10] used multilayer perceptron (DNN) to make wind power predictions which are on similar grounds to solar pond predictions. The accuracy obtained on a dataset on similar grounds indicated the effectiveness of the DNN algorithm. *Iwendi C, Bashir AK, Peshkar A, Sujatha R, Chatterjee JM, Pasupuleti S, Mishra R, Pillai S and Jo O (2020)* [11] uses the Random Forest Algorithm to predict the severity and recovery of a Covid Patient based on his/her geographical details.

2.5 Modeling the Dataset

S.G. Chakrabarty, et al. [12] analyzes the variation of the thickness of zones, ground conditions, and surface losses on the temperature of various zones of Solar Pond. The energy balance equations of various zones of the solar pond were obtained from this study and used for generating a Matlab model which in turn provided the dataset required for the AI models.This study presents the influence of LCZ temperature on the thickness of UCZ and establishes that evaporative losses are more predominant than convection and radiation losses. The results generated by the mathematical models validate the experimental results from other literature. The study also provides various equations necessary to generate a dataset using geometric and meteorological data.

2.6 Miscellaneous

2020 Feizabadi spiral coil [13] analyzes a wavy structure at six different locations of a solar pond. A spiral coil is preferred over a straight tube for extracting heat due to the absence of centrifugal forces. Wavy structure on the walls of the coil increases surface area and influences flow pattern. It is found a wavy structure on the upper wall has maximum enhancement and the lower wall has minimum. Also increasing the wall amplitude significantly increases the efficiencies. *2019 Verma Decay of Radiative Intensity* [14] deals with improving the estimation of radiative intensities in order to optimise the thicknesses of NCZ and LCZ. An equivalent absorption coefficient derived using Beer-equation Lambert's is presented as a better description of the radiative intensity decrease in solar ponds. Based on the increased expression, a transient thermal model involving side and bottom heat losses for a solar collector-type pond with a moving storage zone is constructed. The effects of several factors on the steady and transient states are identified.

2018 el Mansouri Lattice [15] develops a 2-dimensional numerical model for convection in solar ponds. The numerical code accounts for double-diffusive convection in the pond as well as solar radiation absorption via the horizontal fluid layers, allowing for an accurate handling of boundary conditions at the water-free surface. According to the findings, the mean temperature in the heat storage zone (HSZ) rises at a pace of roughly 5°C every day. The daily quantity of energy accumulated in both the heat storage and the heat storage is computed in the absence of heat extraction..

CHAPTER III

THEORY

3.1 Working of Solar Pond

The three different layers of the solar pond have three different densities due to the salinity gradient. The uppermost layer UCZ (Upper convective zone) has the least salinity and density whereas the lowest layer LCZ (Lower convective zone) has the highest, and the NCZ (Non-convective zone) stands in between these two. As the name of the second layer suggests the NCZ does not allow the convection to occur in the water, thus by restricting these convective currents we trap the heat that is absorbed to remain and accumulate in the lowest layer of LCZ.

This energy can later be used for many applications as previously mentioned. In order to use it for any application, we need to first find what is the output we could extract out of it. For this purpose, we need to perform an analysis of all the losses, temperatures of all the three layers, and also geometrical and meteorological parameters affecting the prior two.

The losses include:

1. Reflected radiation loss
2. Heat losses to the environment
3. Heat losses to the ground

Some of the advantages of this type of working of the Solar Pond are -

- Heat remains stored in the collector and its extraction can be done at a very low cost.
- Cleaning of large collector surfaces which tends to be expensive can be easily avoided.
- Large-scale energy generation is made possible by constructing large surfaces.

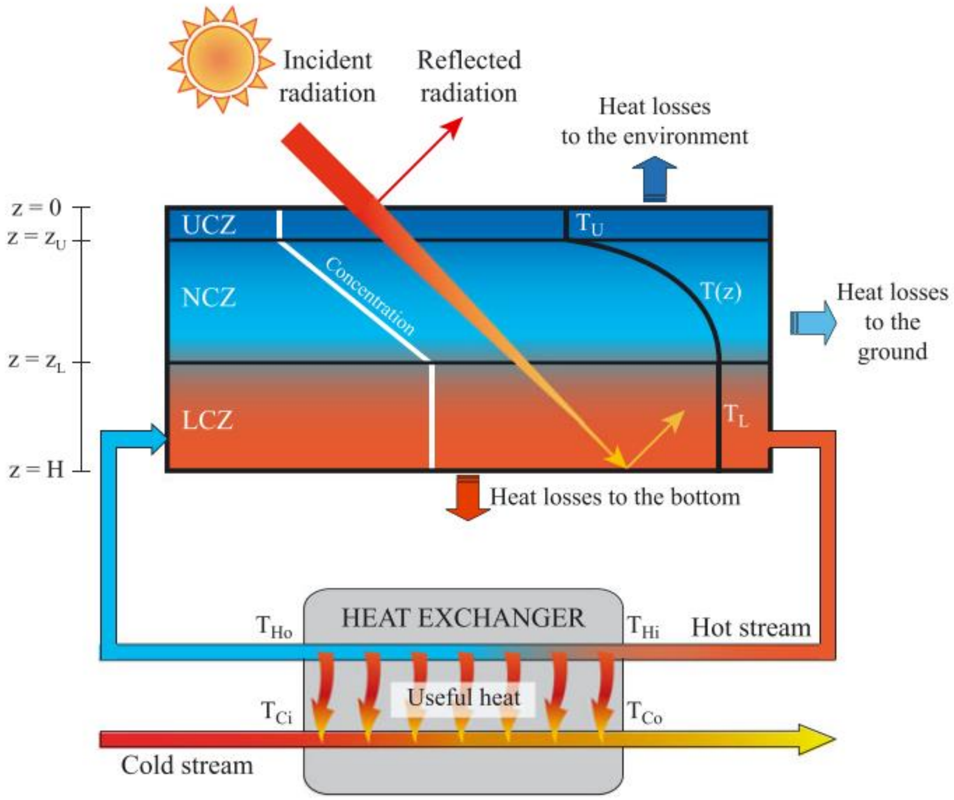


Figure-3 Working of Solar Pond. Adapted from https://blog.nus.edu.sg/oversteppingourmak/2016/10/07/solar-pond/

3.2 AI Algorithms

There are numerous algorithms that can be used for the training of the AI model, but all the algorithms have different objectives of use. It is the task of the model designer to choose the algorithm by studying the needs of his model and also by analyzing the dataset that is going to be used for training the AI model. In our model, we have mainly focused on two algorithms, namely, Random Forest Regressor and Deep Neural Network (DNN).

3.2.1 Random Forest Regressor

This algorithm has its basic element known as a “Decision Tree”, a decision tree itself is a decision support tool having a tree-like model popularly used in machine learning. Decision trees usually have a problem of over fitting to the training dataset.
Random Forest is a machine learning method or algorithm which can be used for both classifications as well as regression. It is basically a multitude of decision trees, for the

output of a regression task it takes the average or means of the output of all the individual decision trees, whereas in classification the output is the one class selected by the most number of trees.

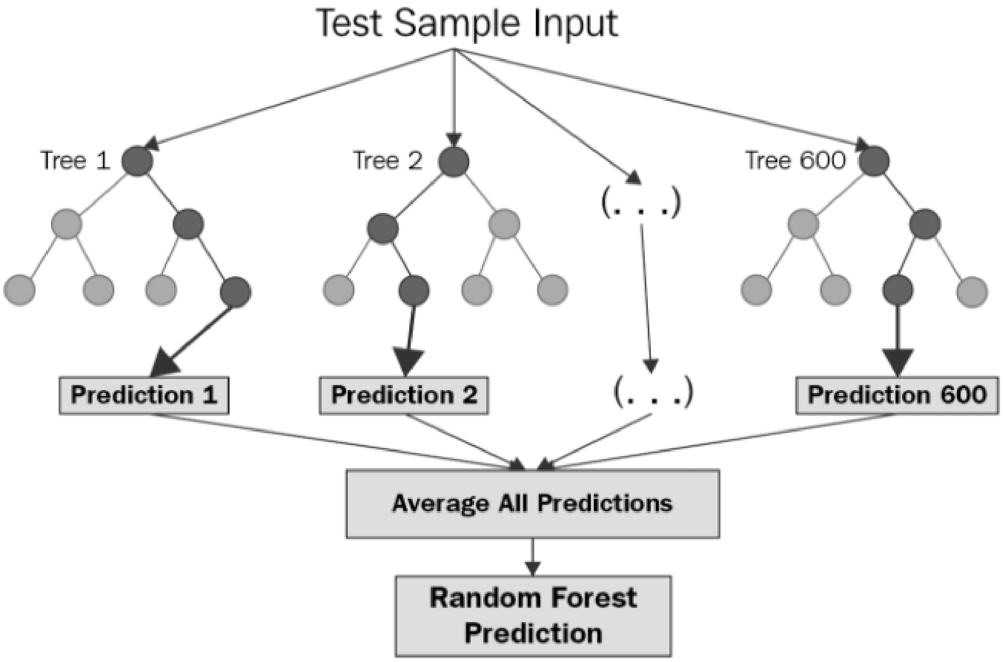


Figure-4 Visualization of Random Forest Algorithm. Adapted from https://levelup.gitconnected.com/random-forest-regression-209c0f354c84 by Chaya Bakshi.

3.2.2 Deep Neural Network (DNN)

The DNN comes under the umbrella of "Artificial Neural Network (ANN)". There are many types of Neural Networks and the reason why they are called so is that they try to function similar to the human brain which itself is Neural Network.

Though there are many different types of Neural networks, they all have some common components like neurons, weights, biases, synapses, and functions. Neural Networks have an input layer to take in the input from the dataset, a hidden layer where the weights and biases are assigned to the input, and an output layer to get the desired output.

DNN is a Neural Network with multiple hidden layers between input and output layers. It can map complex non-linear relationships between input and output. It is a feed forward network, in which at first the data passes from input to output layers through the hidden layers. The initial values of weights (assigned numerical values for a connection between neurons) are set at random. Then in the back propagation, the weights are adjusted to increase the accuracy or decrease the loss.

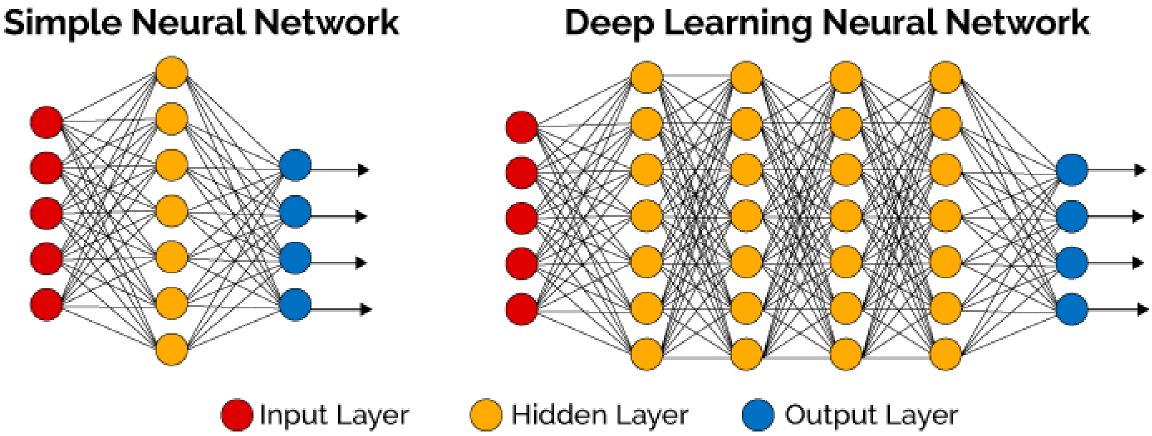


Figure-5 Comparison of Simple Neural Network and Deep Neural Network. Reprinted from Williams, Vinay & Argyriou, Vasileios & Shaw, Peter & Montag, Christoph & Herdrich, Georg & Knoll, Aaron & Moertl, Maximilian. (2019).

CHAPTER IV
METHODOLOGY

4.1 Modeling the Dataset

The modeling of the dataset is done on MATLAB. The Research Paper titled **Investigation of temperature development in salinity gradient solar ponds using a transient model of heat transfer** was used as the base for modeling the model. The MATLAB model is based on the following assumptions -

- Both UCZ and LCZ are assumed to be perfectly mixed.
- The side walls are correctly insulated.
- The solar radiation reaching the LCZ is wholly absorbed in this layer.
- Heat storage in the NCZ is not considered for the estimation of temperatures of LCZ and UCZ.
- Conduction heat transfer as per Fourier's law is considered in NCZ.
- Mass transfer is not considered in this study.

The modeling was done based on the following equations -

$$\rho_{UCZ} C_{p,UCZ} t_{UCZ} A_{UCZ} \left(\frac{dT_{UCZ}}{d\tau} \right) = Q_{s,UCZ} + Q_{cond} - Q_{losses,UCZ}$$

$$Q_{cond} = k_{NCZ} A_{NCZ} \left(\frac{dT_{NCZ,m}}{dt} \right) + Q_{s,NCZ,m} - Q_{losses,NCZ,m}$$

$$\rho_{LCZ} C_{p,LCZ} t_{LCZ} A_{LCZ} \left(\frac{dT_{LCZ}}{d\tau} \right) = Q_{s,LCZ} - Q_{cond} - Q_{load} - Q_{losses,LCZ}$$

Figure-6 Energy Balance Equations of a Solar Pond

4.2 Making the Dataset

The dissertation aims to reduce the computational time required for the characterization of a solar pond. This study has identified 10 input parameters and 2 output parameters which will form the structure of the AI models generated. The 10 input parameters include

- Solar Radiation
- Ambient Temperature
- Sky Temperature
- Relative Humidity
- Wind Velocity
- Area of Solar Pond
- Thickness of NCZ
- Thickness of LCZ
- Depth of Ground
- Thermal Conductivity of Ground

The output parameters predicted by the AI include:

- Thickness of LCZ
- Thickness of UCZ

The dissertation used meteorological and geometric data in order to generate the dataset for 6 locations, each dataset containing 18720 variations. The table consisting of the 6 Locations taken under consideration for making the dataset is given below –

Table 2 Locations considered for making the dataset

Location	**Latitude**	**Longitude**
Bengaluru	12.9716° N	77.5946° E
Bhavnagar	21.7645° N	72.1519° E
Chandigarh	30.7333° N	76.7794° E

Gandhinagar	23.2156° N	72.6369° E
Kolkata	22.5726° N	88.3639° E

The Methodology for making the final dataset of 12000 rows from the 6 different locations is given below -

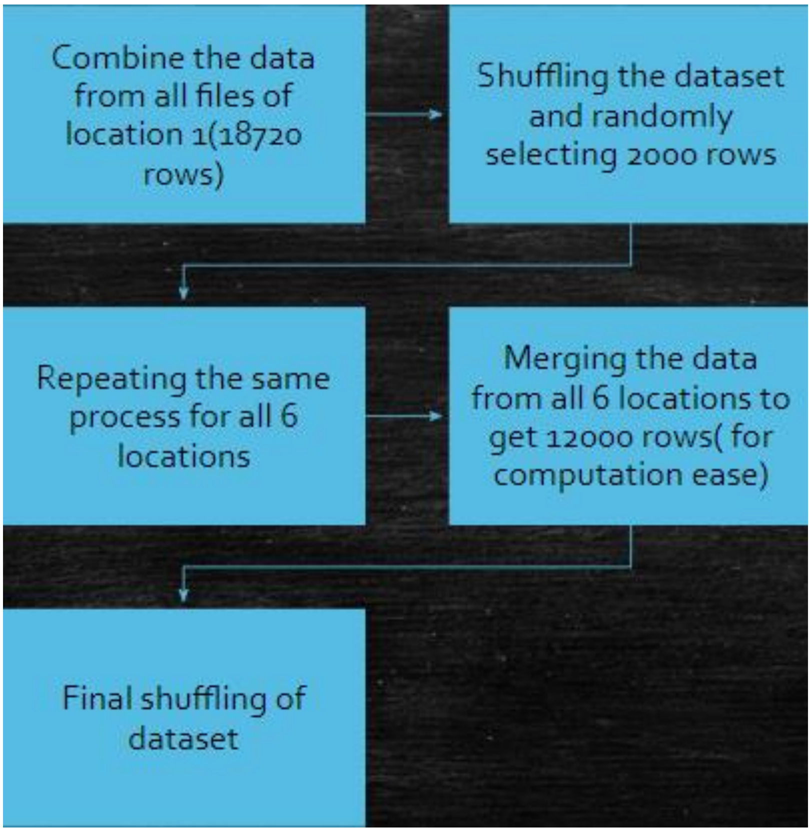
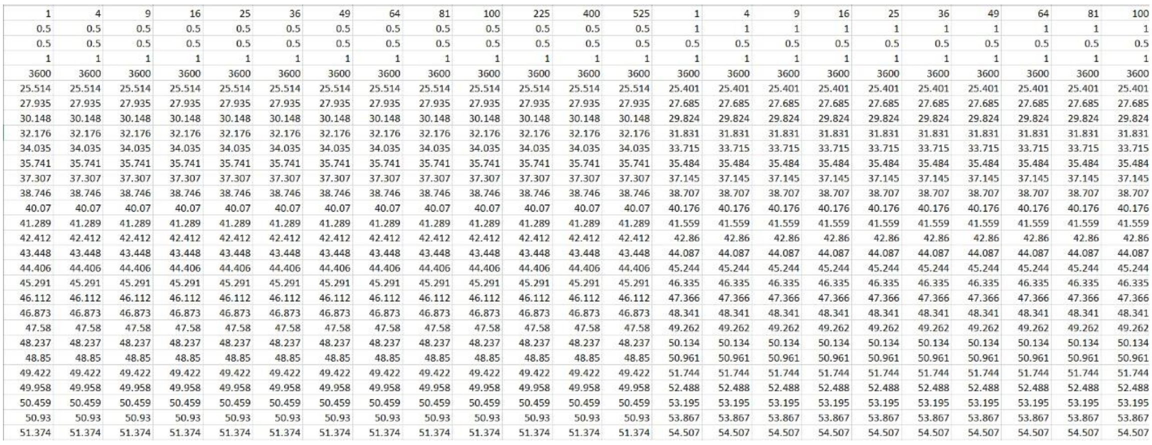

Combine the data from all files of location 1(18720 rows)
Shuffling the dataset and randomly selecting 2000 rows
Repeating the same process for all 6 locations
Merging the data from all 6 locations to get 12000 rows(for computation ease)
Final shuffling of dataset

Figure-7 Flow Chart Depicting the Making of Dataset

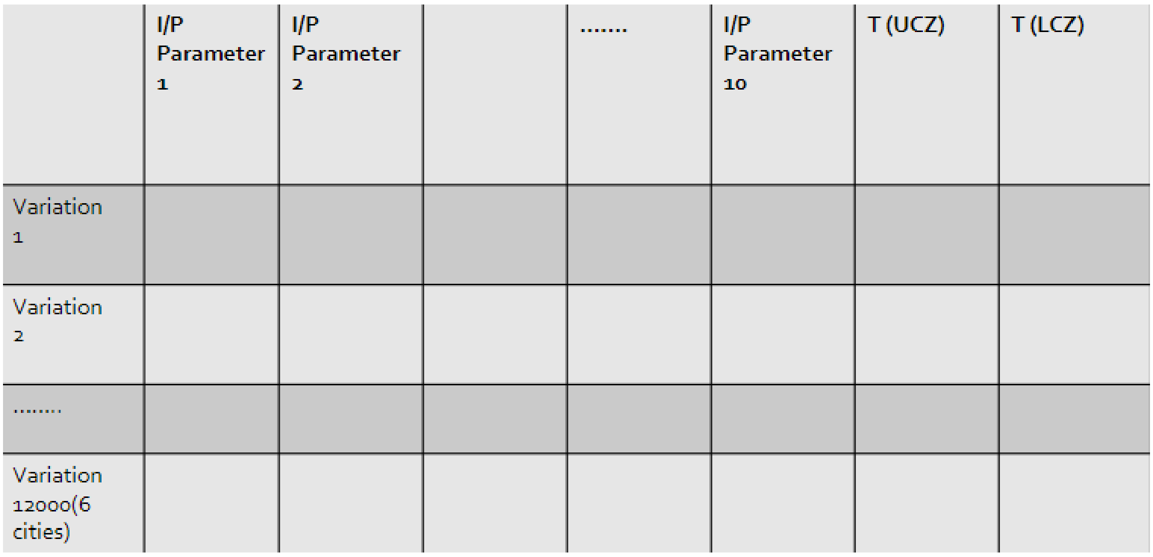
The Matlab model mentioned in the previous section is fed the meteorological data for 6 different locations. The Matlab model generates an excel file consisting of 18720 variations distributed along the rows of the file for a particular location.



1	4	9	16	25	36	49	64	81	100	225	400	525	1	4	9	16	25	36	49	64	81	100
0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	1	1	1	1	1	1	1	1	1	1
0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
3600	3600	3600	3600	3600	3600	3600	3600	3600	3600	3600	3600	3600	3600	3600	3600	3600	3600	3600	3600	3600	3600	3600
25.514	25.514	25.514	25.514	25.514	25.514	25.514	25.514	25.514	25.514	25.514	25.514	25.514	25.401	25.401	25.401	25.401	25.401	25.401	25.401	25.401	25.401	25.401
27.935	27.935	27.935	27.935	27.935	27.935	27.935	27.935	27.935	27.935	27.935	27.935	27.935	27.685	27.685	27.685	27.685	27.685	27.685	27.685	27.685	27.685	27.685
30.148	30.148	30.148	30.148	30.148	30.148	30.148	30.148	30.148	30.148	30.148	30.148	30.148	29.824	29.824	29.824	29.824	29.824	29.824	29.824	29.824	29.824	29.824
32.176	32.176	32.176	32.176	32.176	32.176	32.176	32.176	32.176	32.176	32.176	32.176	32.176	31.831	31.831	31.831	31.831	31.831	31.831	31.831	31.831	31.831	31.831
34.035	34.035	34.035	34.035	34.035	34.035	34.035	34.035	34.035	34.035	34.035	34.035	34.035	33.715	33.715	33.715	33.715	33.715	33.715	33.715	33.715	33.715	33.715
35.741	35.741	35.741	35.741	35.741	35.741	35.741	35.741	35.741	35.741	35.741	35.741	35.741	35.484	35.484	35.484	35.484	35.484	35.484	35.484	35.484	35.484	35.484
37.307	37.307	37.307	37.307	37.307	37.307	37.307	37.307	37.307	37.307	37.307	37.307	37.307	37.145	37.145	37.145	37.145	37.145	37.145	37.145	37.145	37.145	37.145
38.746	38.746	38.746	38.746	38.746	38.746	38.746	38.746	38.746	38.746	38.746	38.746	38.746	38.707	38.707	38.707	38.707	38.707	38.707	38.707	38.707	38.707	38.707
40.07	40.07	40.07	40.07	40.07	40.07	40.07	40.07	40.07	40.07	40.07	40.07	40.07	40.176	40.176	40.176	40.176	40.176	40.176	40.176	40.176	40.176	40.176
41.289	41.289	41.289	41.289	41.289	41.289	41.289	41.289	41.289	41.289	41.289	41.289	41.289	41.559	41.559	41.559	41.559	41.559	41.559	41.559	41.559	41.559	41.559
42.412	42.412	42.412	42.412	42.412	42.412	42.412	42.412	42.412	42.412	42.412	42.412	42.412	42.86	42.86	42.86	42.86	42.86	42.86	42.86	42.86	42.86	42.86
43.448	43.448	43.448	43.448	43.448	43.448	43.448	43.448	43.448	43.448	43.448	43.448	43.448	44.087	44.087	44.087	44.087	44.087	44.087	44.087	44.087	44.087	44.087
44.406	44.406	44.406	44.406	44.406	44.406	44.406	44.406	44.406	44.406	44.406	44.406	44.406	45.244	45.244	45.244	45.244	45.244	45.244	45.244	45.244	45.244	45.244
45.291	45.291	45.291	45.291	45.291	45.291	45.291	45.291	45.291	45.291	45.291	45.291	45.291	46.335	46.335	46.335	46.335	46.335	46.335	46.335	46.335	46.335	46.335
46.112	46.112	46.112	46.112	46.112	46.112	46.112	46.112	46.112	46.112	46.112	46.112	46.112	47.366	47.366	47.366	47.366	47.366	47.366	47.366	47.366	47.366	47.366
46.873	46.873	46.873	46.873	46.873	46.873	46.873	46.873	46.873	46.873	46.873	46.873	46.873	48.341	48.341	48.341	48.341	48.341	48.341	48.341	48.341	48.341	48.341
47.58	47.58	47.58	47.58	47.58	47.58	47.58	47.58	47.58	47.58	47.58	47.58	47.58	49.262	49.262	49.262	49.262	49.262	49.262	49.262	49.262	49.262	49.262
48.237	48.237	48.237	48.237	48.237	48.237	48.237	48.237	48.237	48.237	48.237	48.237	48.237	50.134	50.134	50.134	50.134	50.134	50.134	50.134	50.134	50.134	50.134
48.85	48.85	48.85	48.85	48.85	48.85	48.85	48.85	48.85	48.85	48.85	48.85	48.85	50.961	50.961	50.961	50.961	50.961	50.961	50.961	50.961	50.961	50.961
49.422	49.422	49.422	49.422	49.422	49.422	49.422	49.422	49.422	49.422	49.422	49.422	49.422	51.744	51.744	51.744	51.744	51.744	51.744	51.744	51.744	51.744	51.744
49.958	49.958	49.958	49.958	49.958	49.958	49.958	49.958	49.958	49.958	49.958	49.958	49.958	52.488	52.488	52.488	52.488	52.488	52.488	52.488	52.488	52.488	52.488
50.459	50.459	50.459	50.459	50.459	50.459	50.459	50.459	50.459	50.459	50.459	50.459	50.459	53.195	53.195	53.195	53.195	53.195	53.195	53.195	53.195	53.195	53.195
50.93	50.93	50.93	50.93	50.93	50.93	50.93	50.93	50.93	50.93	50.93	50.93	50.93	53.867	53.867	53.867	53.867	53.867	53.867	53.867	53.867	53.867	53.867
51.374	51.374	51.374	51.374	51.374	51.374	51.374	51.374	51.374	51.374	51.374	51.374	51.374	54.507	54.507	54.507	54.507	54.507	54.507	54.507	54.507	54.507	54.507

Figure-8 Dataset for a Location

This dataset is shuffled and 2000 rows are randomly selected to comprise the final dataset. A similar process is repeated with the other five locations and the data for all the six locations are merged to give a final dataset of 12000 rows and 7670 columns, that depict the various input and output parameters identified above over a period of three years. This dataset is given one final shuffle in order to randomize the data fed. The final structure of the dataset is as follows

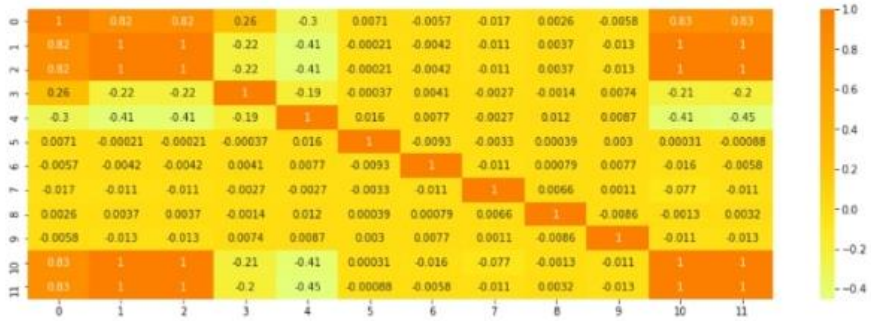


	I/P Parameter 1	I/P Parameter 2		…….	I/P Parameter 10	T (UCZ)	T (LCZ)
Variation 1							
Variation 2							
……..							
Variation 12000(6 cities)							

Figure-9 Structure of the Final Dataset

4.2.1 Data Preprocessing

The data before being fed to the AI model is pre-processed to identify any loopholes that could affect the accuracy of the prediction. This dissertation has made of the correlation matrix to find out the parameters that have a lesser influence on the prediction output. These parameters were omitted from being fed to the AI model and resulted in improved accuracies. This way correlation matrix serves as a clinical tool to identify the less influential parameters and clean the dataset to provide the best possible efficiency.



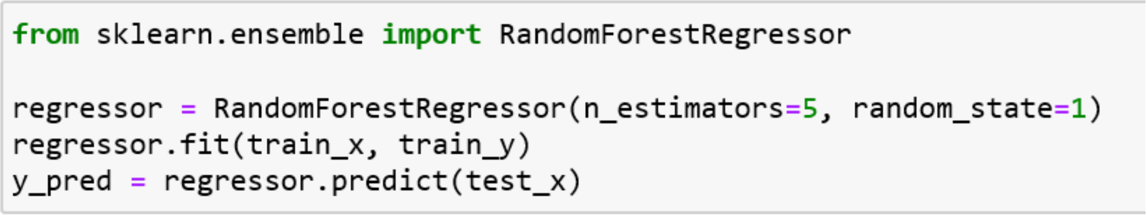
	0	1	2	3	4	5	6	7	8	9	10	11
0	1	0.82	0.82	0.26	-0.3	0.0071	-0.0057	-0.017	0.0026	-0.0058	0.83	0.83
1	0.82	1	1	-0.22	-0.41	-0.00021	-0.0042	-0.011	0.0037	-0.013	1	1
2	0.82	1	1	-0.22	-0.41	-0.00021	-0.0042	-0.011	0.0037	-0.013	1	1
3	0.26	-0.22	-0.22	1	-0.19	-0.00037	0.0041	-0.0027	-0.0014	0.0074	-0.21	-0.2
4	-0.3	-0.41	-0.41	-0.19	1	0.016	0.0077	-0.0027	0.012	0.0087	-0.41	-0.45
5	0.0071	-0.00021	-0.00021	-0.00037	0.016	1	-0.0093	-0.0033	0.00039	0.003	0.00031	-0.00088
6	-0.0057	-0.0042	-0.0042	0.0041	0.0077	-0.0093	1	-0.011	0.00079	0.0077	-0.016	-0.0058
7	-0.017	-0.011	-0.011	-0.0027	-0.0027	-0.0033	-0.011	1	0.0066	0.0011	-0.077	-0.011
8	0.0026	0.0037	0.0037	-0.0014	0.012	0.00039	0.00079	0.0066	1	-0.0086	-0.0013	0.0032
9	-0.0058	-0.013	-0.013	0.0074	0.0087	0.003	0.0077	0.0011	-0.0086	1	-0.011	-0.013
10	0.83	1	1	-0.21	-0.41	0.00031	-0.016	-0.077	-0.0013	-0.011	1	1
11	0.83	1	1	-0.2	-0.45	-0.00088	-0.0058	-0.011	0.0032	-0.013	1	1

Figure-10 Correlation Matrix

The 12 parameters (10 input and 2 output) are tagged from 0 to 11. As seen in the figure above parameters 5 to 9 have a negligible or negative effect on the output parameters (10 & 11). In this manner, the correlation matrix serves as an important gateway to obtaining a ‘clean’ dataset before training.

4.3 AI Model Architecture

4.3.1 Random Forest Regressor



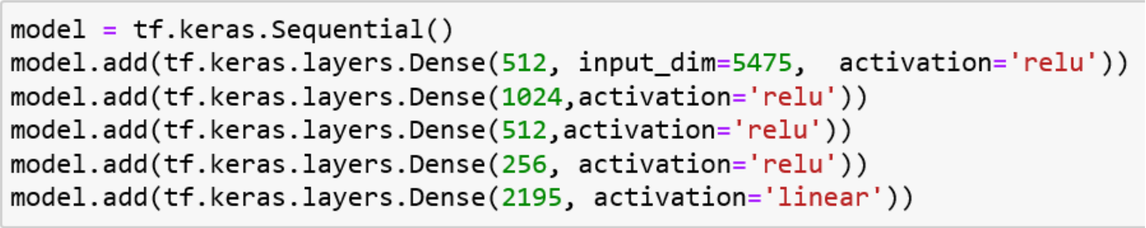
```
from sklearn.ensemble import RandomForestRegressor

regressor = RandomForestRegressor(n_estimators=5, random_state=1)
regressor.fit(train_x, train_y)
y_pred = regressor.predict(test_x)
```

Figure-11 Codes for RandomForestRegressor Algorithm

In the RandomForestRegressor, the **n_estimators** are passed as a hyper parameter to the model. As its value increases, the model tends to be accurate but its computational time also increases, so, it becomes essential to find its optimized value by trial & error method.

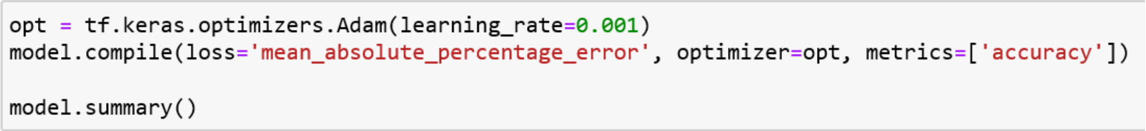
There are various other hyperparameters that need to be tuned along with **n_estimators**. But owing to the high accuracy achieved through this architecture, we decided to keep the rest of the hyperparameters at their default value.

4.3.2 Deep Neural Network



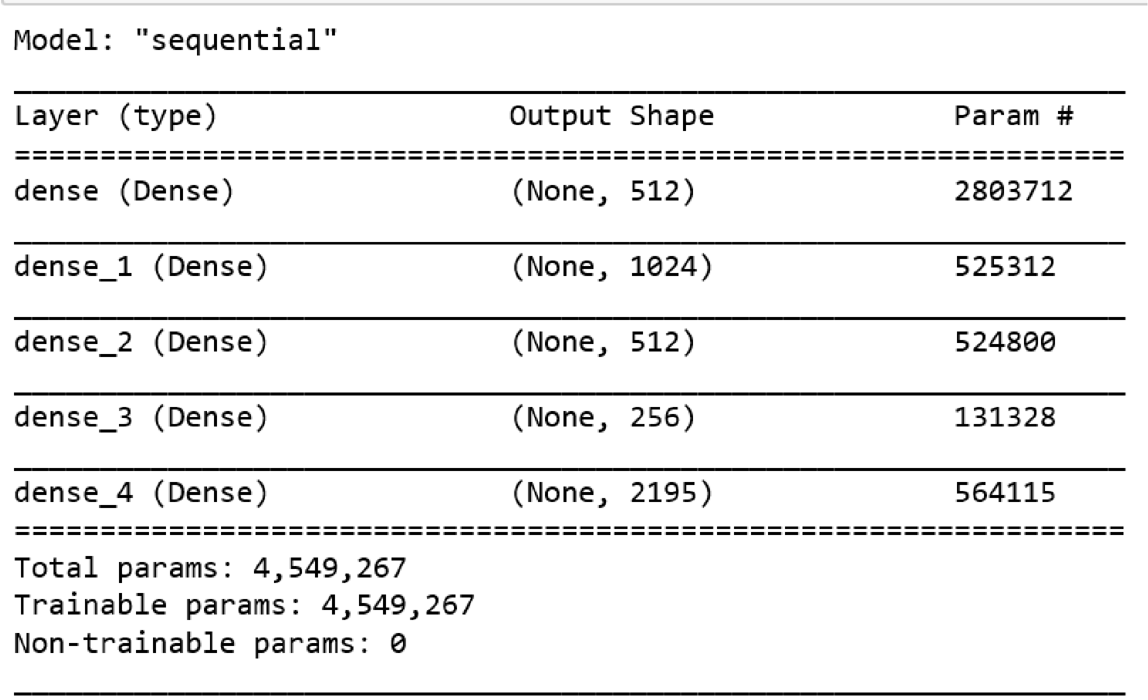
```
model = tf.keras.Sequential()
model.add(tf.keras.layers.Dense(512, input_dim=5475,  activation='relu'))
model.add(tf.keras.layers.Dense(1024,activation='relu'))
model.add(tf.keras.layers.Dense(512,activation='relu'))
model.add(tf.keras.layers.Dense(256, activation='relu'))
model.add(tf.keras.layers.Dense(2195, activation='linear'))
```

Figure-12 Layers & Neurons in the DNN



```
opt = tf.keras.optimizers.Adam(learning_rate=0.001)
model.compile(loss='mean_absolute_percentage_error', optimizer=opt, metrics=['accuracy'])

model.summary()
```

Figure-13 Optimizer & Metrics used in the DNN



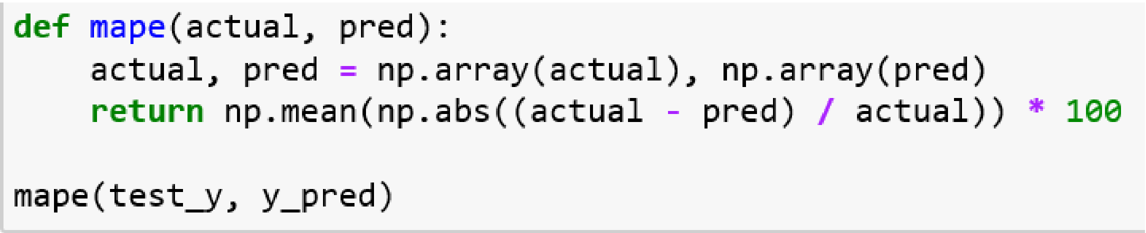
```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense (Dense)                (None, 512)               2803712
_________________________________________________________________
dense_1 (Dense)              (None, 1024)              525312
_________________________________________________________________
dense_2 (Dense)              (None, 512)               524800
_________________________________________________________________
dense_3 (Dense)              (None, 256)               131328
_________________________________________________________________
dense_4 (Dense)              (None, 2195)              564115
=================================================================
Total params: 4,549,267
Trainable params: 4,549,267
Non-trainable params: 0
_________________________________________________________________
```

Figure-14 Final Architecture of DNN Mode

In the DNN, the number of layers, & a number of neurons in each layer can be varied to generate various different architectures. By varying them, we vary the number of trainable parameters, which in turn affects the accuracy of the model. Along with this, we also change the Optimizer & the Learning Rate. The choice of Optimizer determines the formulae based on which the parameters will update themselves & the Learning Rate determines how fast or slow the parameters will approach their optimum values.

4.4 Calculating Losses

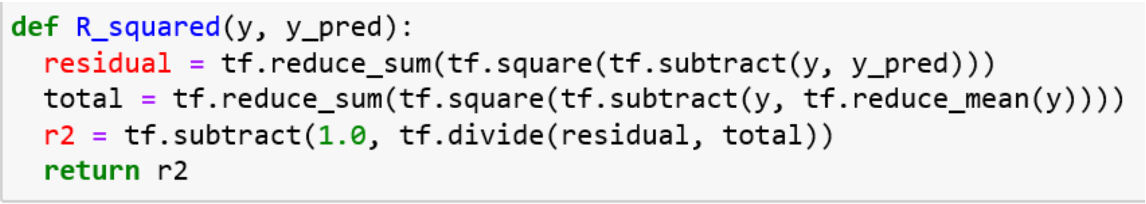
4.4.1 Mean Average Percentage Error (MAPE)



```
def mape(actual, pred):
    actual, pred = np.array(actual), np.array(pred)
    return np.mean(np.abs((actual - pred) / actual)) * 100

mape(test_y, y_pred)
```

Figure-15 Code for Mean Average Percentage Error

4.4.2 R-squared Error



```
def R_squared(y, y_pred):
  residual = tf.reduce_sum(tf.square(tf.subtract(y, y_pred)))
  total = tf.reduce_sum(tf.square(tf.subtract(y, tf.reduce_mean(y))))
  r2 = tf.subtract(1.0, tf.divide(residual, total))
  return r2
```

Figure-16 Code for R-Squared Error

For calculating both the losses, we pass the Actual Test Output Values & the Predicted Output Values as arguments to the respective functions. Based on the results obtained from the metrics, we can decide how accurate the model is & which model to use for the given dataset.

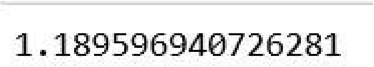
CHAPTER V

RESULTS

5.1 Random Forest Regressor

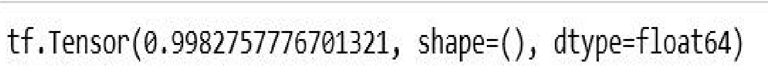
Metrics used for evaluation of Models -

MAPE -



```
1.189596940726281
```

R-Squared Error -



```
tf.Tensor(0.9982757776701321, shape=(), dtype=float64)
```

The average of all values from all columns was taken from the final mixed dataset and its predicted output was plotted against the actual output values. The same procedure was repeated for all the different 6 locations, but only for RandomForestRegressor since it gave sufficiently high accuracies than the Deep Neural Network.

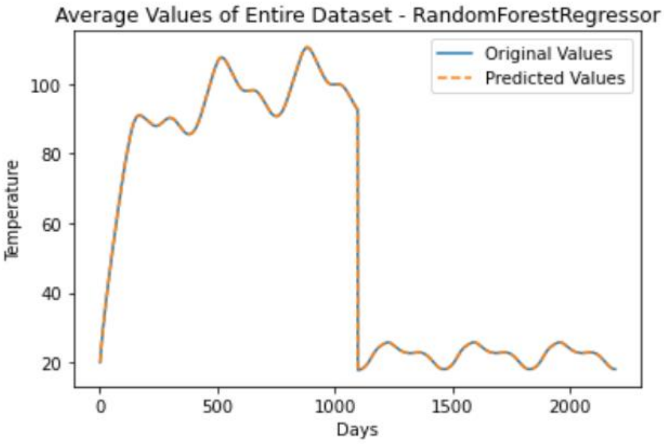


Figure-17 Average Values of Entire Dataset– Original vs. Predicted

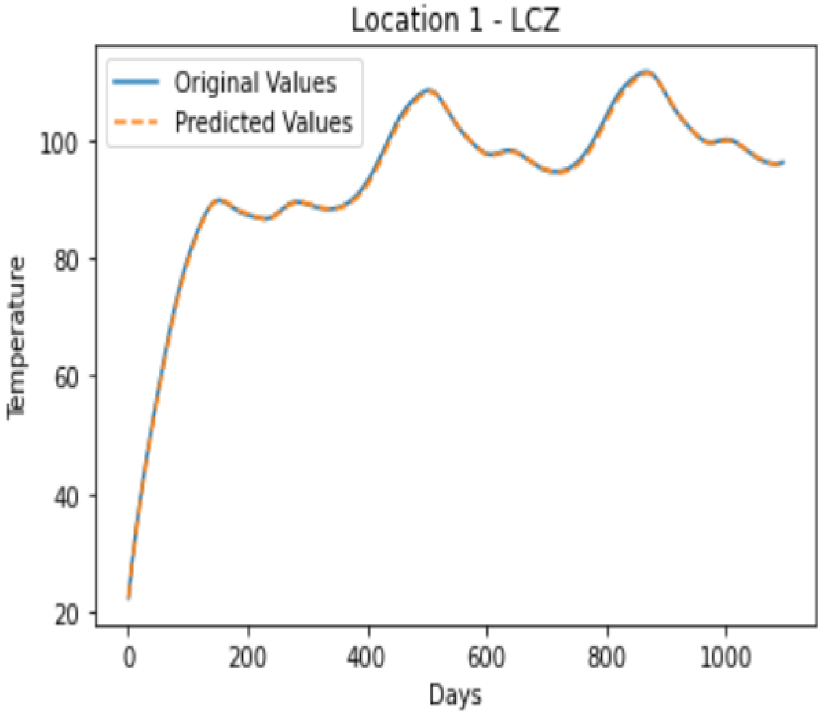


Figure-18 Average Values of Location 1 for LCZ– Original vs. Predicted

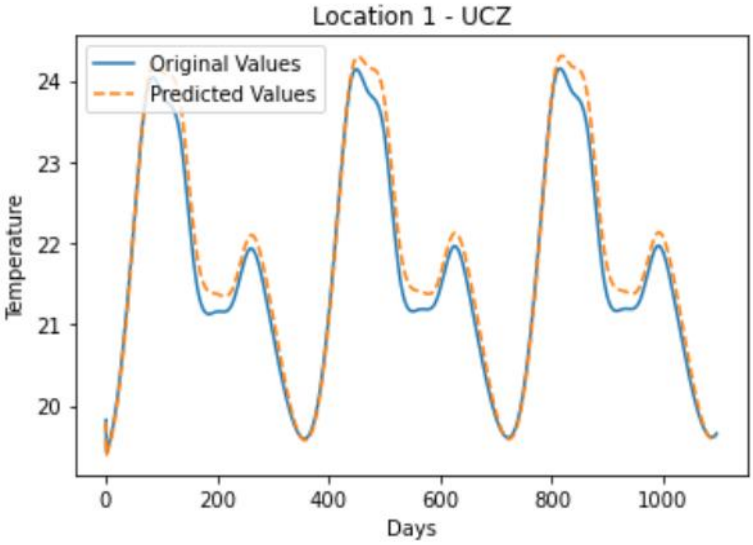


Figure-19 Average Values of Location 1 for UCZ– Original vs. Predicted

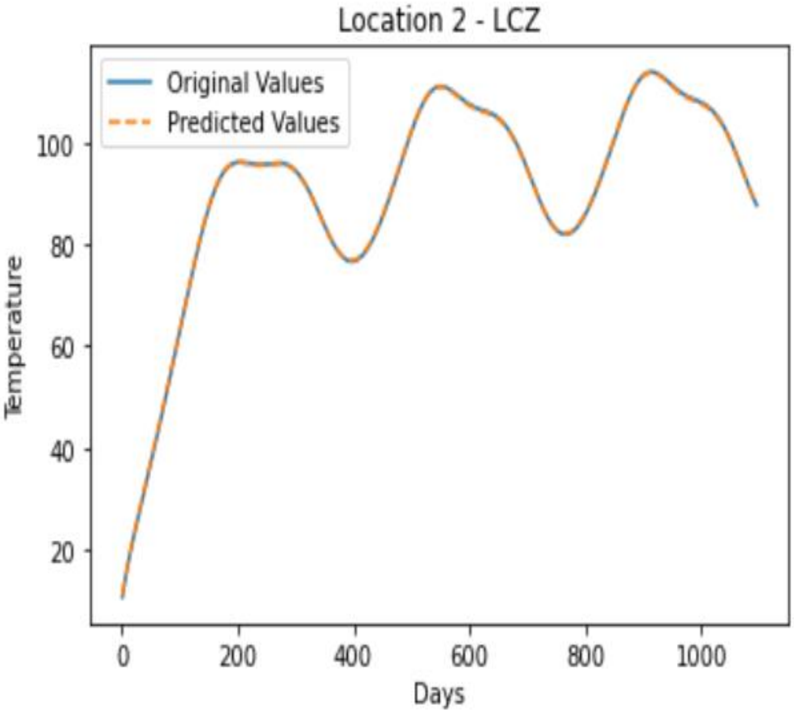


Figure-20 Average Values of Location 2 for LCZ– Original vs. Predicted

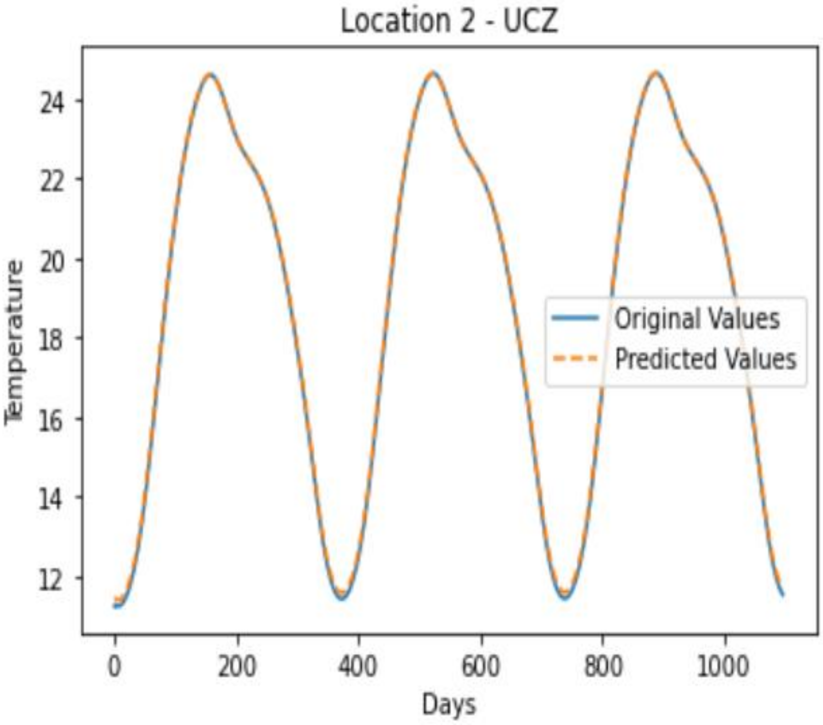


Figure-21 Average Values of Location 2 for UCZ– Original vs. Predicted

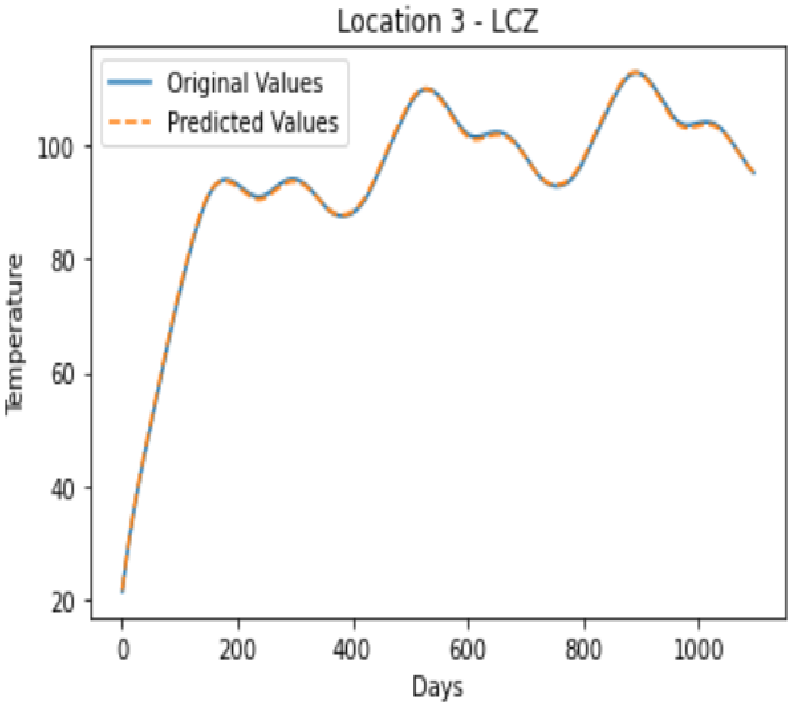


Figure-22 Average Values of Location 3 for LCZ– Original vs. Predicted

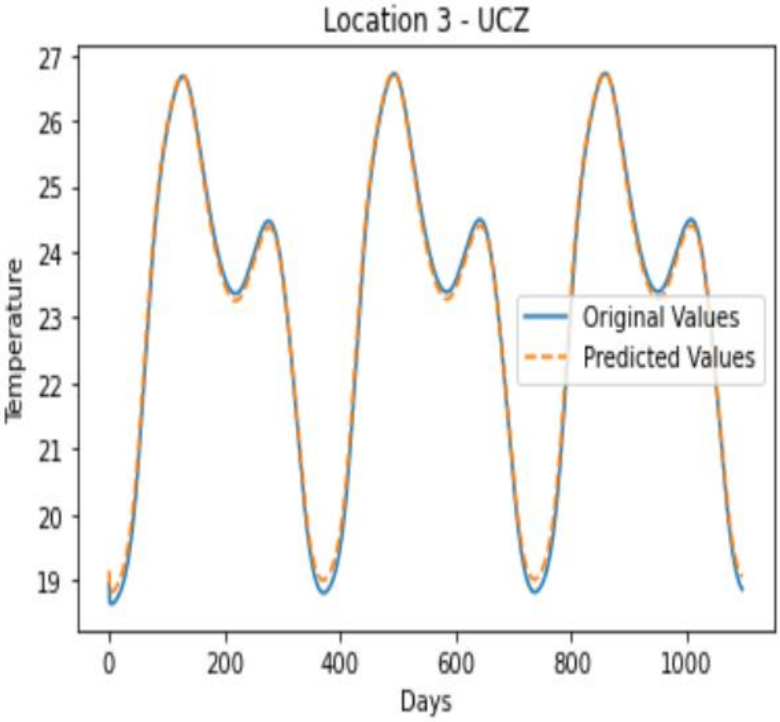


Figure-23 Average Values of Location 3 for UCZ– Original vs. Predicted

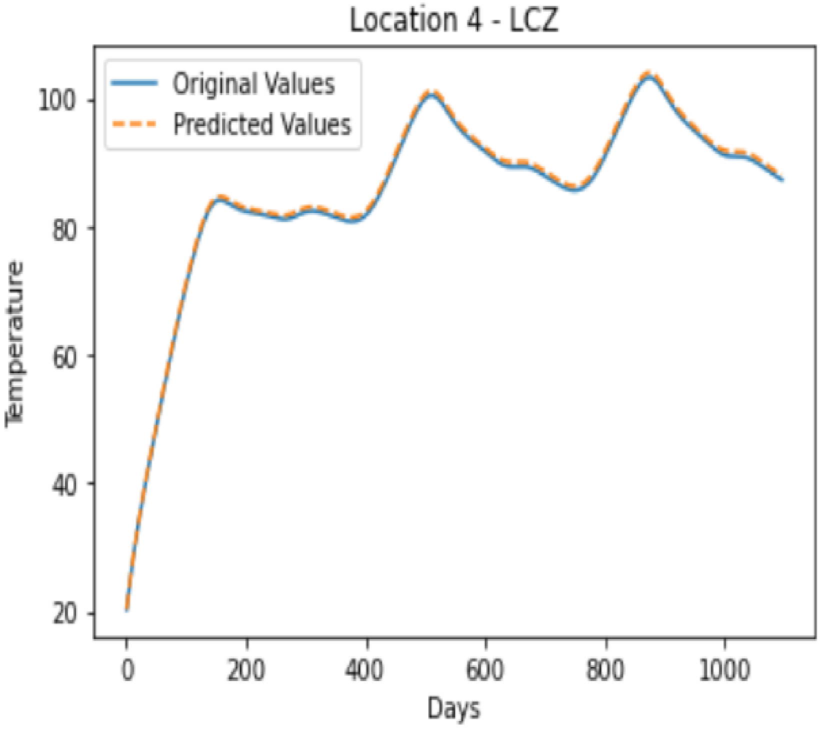


Figure-24 Average Values of Location 4 for LCZ– Original vs. Predicted

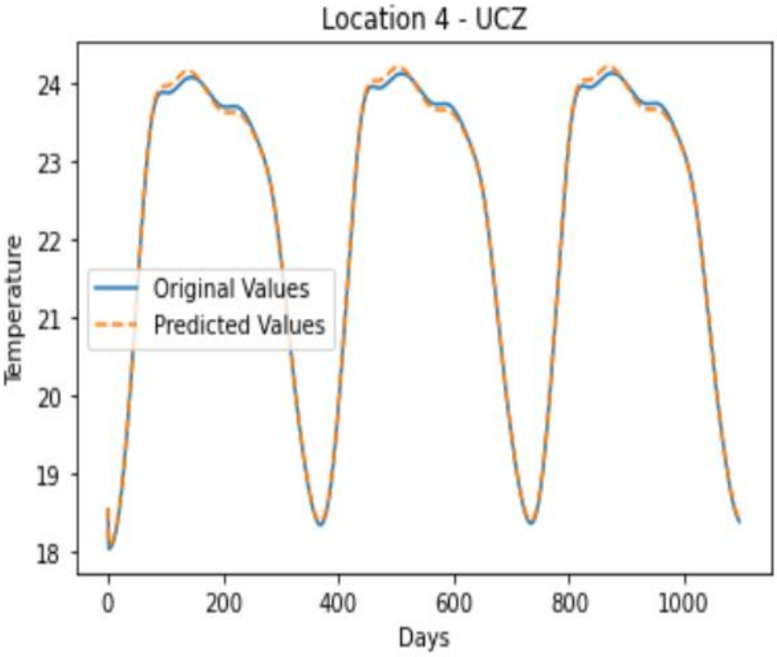


Figure-25 Average Values of Location 4 for UCZ– Original vs. Predicted

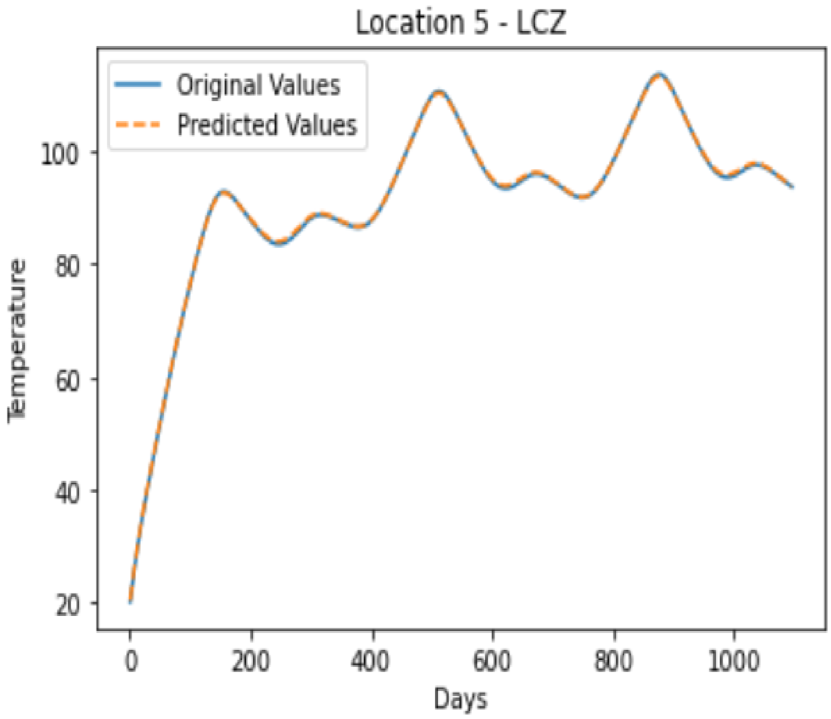


Figure-26 Average Values of Location 5 for LCZ– Original vs. Predicted

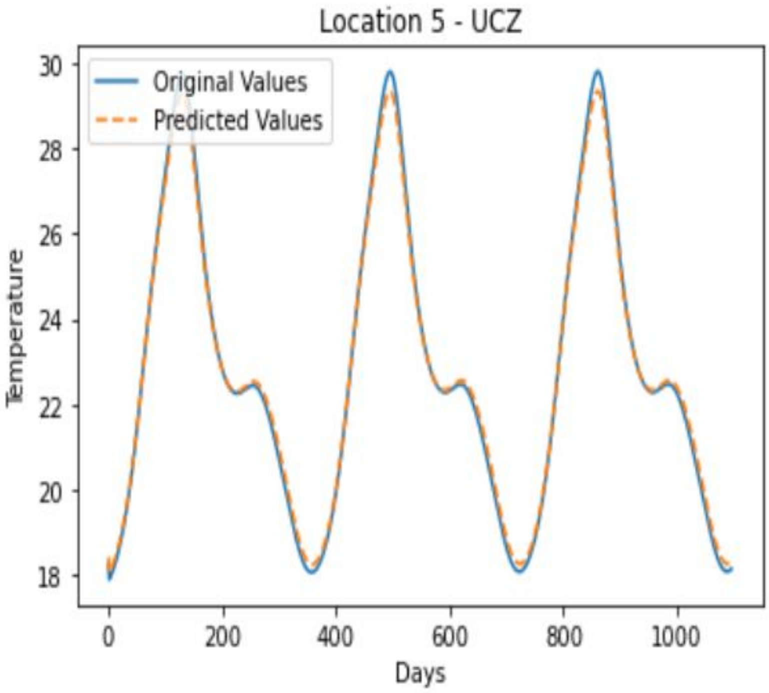


Figure-27 Average Values of Location 5 for UCZ– Original vs. Predicted

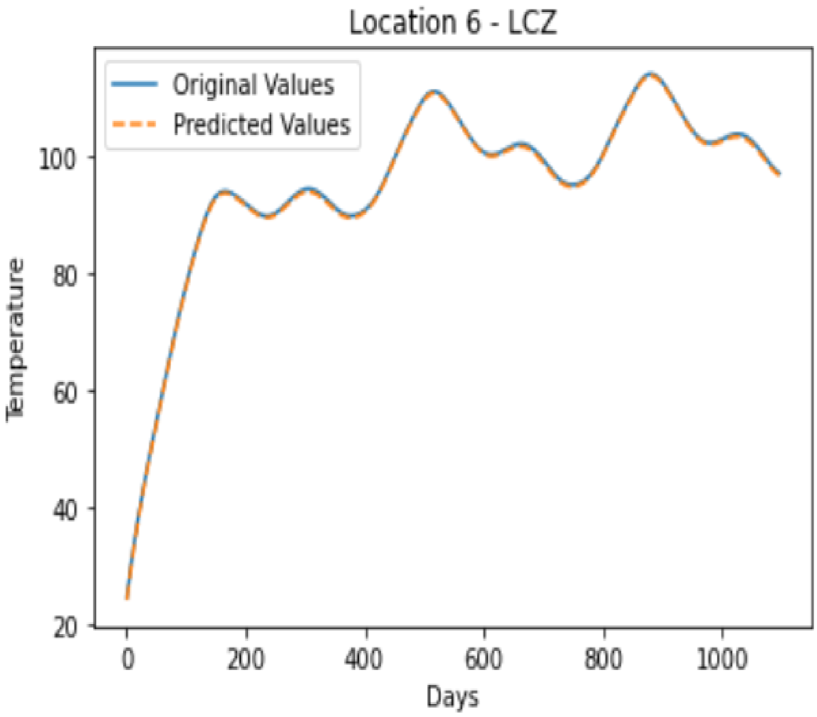


Figure-28 Average Values of Location 6 for UCZ– Original vs. Predicted

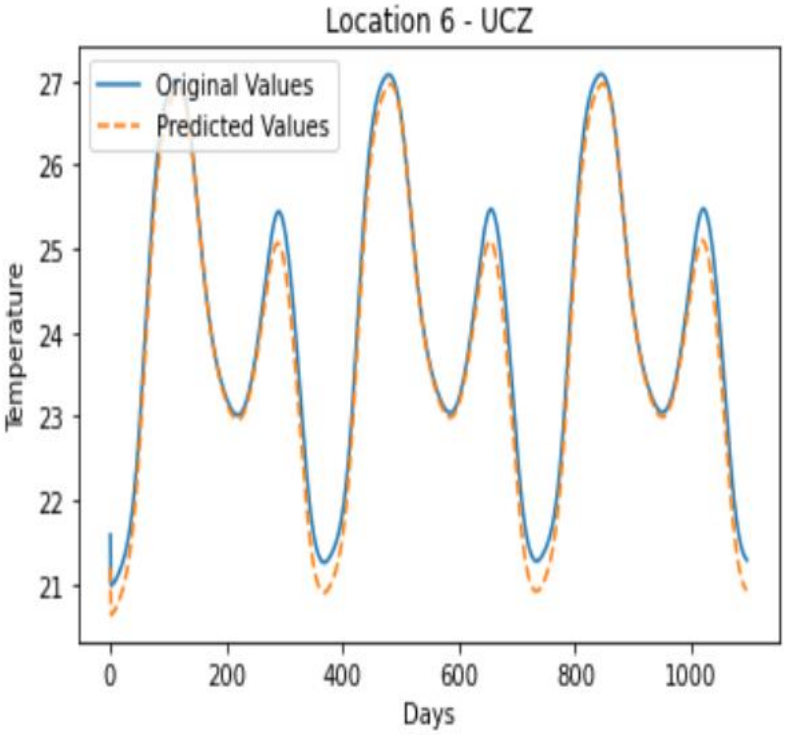
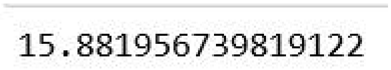


Figure-29 Average Values of Location 6 for UCZ– Original vs. Predicted

5.2 DNN

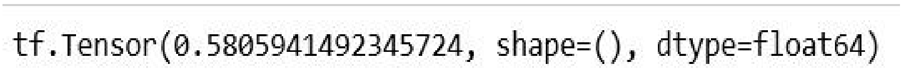
The metrics used for evaluation of the model include -

MAPE -



```
15.881956739819122
```

R-Squared Error -



```
tf.Tensor(0.5805941492345724, shape=(), dtype=float64)
```

The Graph of Average Values of the Actual Outputs & the Predicted Outputs is shown below -

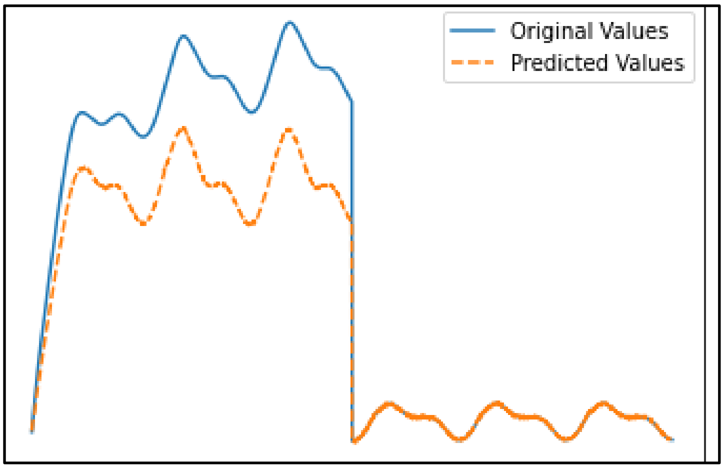
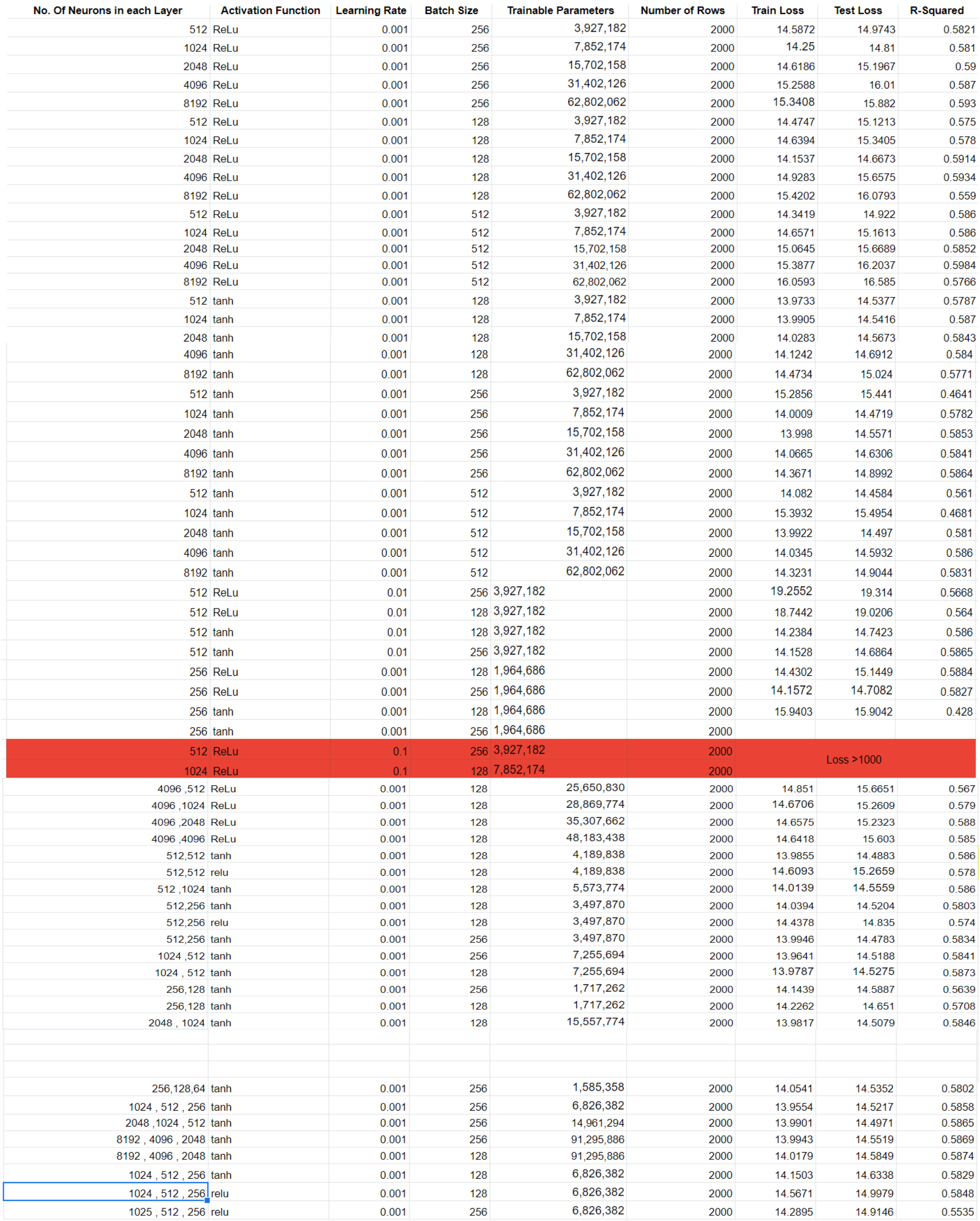


Figure-30 Average Values of Entire Dataset – Original vs. Predicted

Table-3 Different Architectures for DNN



No. Of Neurons in each Layer	Activation Function	Learning Rate	Batch Size	Trainable Parameters	Number of Rows	Train Loss	Test Loss	R-Squared
512	ReLu	0.001	256	3,927,182	2000	14.5872	14.9743	0.5821
1024	ReLu	0.001	256	7,852,174	2000	14.25	14.81	0.581
2048	ReLu	0.001	256	15,702,158	2000	14.6186	15.1967	0.59
4096	ReLu	0.001	256	31,402,126	2000	15.2588	16.01	0.587
8192	ReLu	0.001	256	62,802,062	2000	15.3408	15.882	0.593
512	ReLu	0.001	128	3,927,182	2000	14.4747	15.1213	0.575
1024	ReLu	0.001	128	7,852,174	2000	14.6394	15.3405	0.578
2048	ReLu	0.001	128	15,702,158	2000	14.1537	14.6673	0.5914
4096	ReLu	0.001	128	31,402,126	2000	14.9283	15.6575	0.5934
8192	ReLu	0.001	128	62,802,062	2000	15.4202	16.0793	0.559
512	ReLu	0.001	512	3,927,182	2000	14.3419	14.922	0.586
1024	ReLu	0.001	512	7,852,174	2000	14.6571	15.1613	0.586
2048	ReLu	0.001	512	15,702,158	2000	15.0645	15.6689	0.5852
4096	ReLu	0.001	512	31,402,126	2000	15.3877	16.2037	0.5984
8192	ReLu	0.001	512	62,802,062	2000	16.0593	16.585	0.5766
512	tanh	0.001	128	3,927,182	2000	13.9733	14.5377	0.5787
1024	tanh	0.001	128	7,852,174	2000	13.9905	14.5416	0.587
2048	tanh	0.001	128	15,702,158	2000	14.0283	14.5673	0.5843
4096	tanh	0.001	128	31,402,126	2000	14.1242	14.6912	0.584
8192	tanh	0.001	128	62,802,062	2000	14.4734	15.024	0.5771
512	tanh	0.001	256	3,927,182	2000	15.2856	15.441	0.4641
1024	tanh	0.001	256	7,852,174	2000	14.0009	14.4719	0.5782
2048	tanh	0.001	256	15,702,158	2000	13.998	14.5571	0.5853
4096	tanh	0.001	256	31,402,126	2000	14.0665	14.6306	0.5841
8192	tanh	0.001	256	62,802,062	2000	14.3671	14.8992	0.5864
512	tanh	0.001	512	3,927,182	2000	14.082	14.4584	0.561
1024	tanh	0.001	512	7,852,174	2000	15.3932	15.4954	0.4681
2048	tanh	0.001	512	15,702,158	2000	13.9922	14.497	0.581
4096	tanh	0.001	512	31,402,126	2000	14.0345	14.5932	0.586
8192	tanh	0.001	512	62,802,062	2000	14.3231	14.9044	0.5831
512	ReLu	0.01	256	3,927,182	2000	19.2552	19.314	0.5668
512	ReLu	0.01	128	3,927,182	2000	18.7442	19.0206	0.564
512	tanh	0.01	128	3,927,182	2000	14.2384	14.7423	0.586
512	tanh	0.01	256	3,927,182	2000	14.1528	14.6864	0.5865
256	ReLu	0.001	128	1,964,686	2000	14.4302	15.1449	0.5884
256	ReLu	0.001	256	1,964,686	2000	14.1572	14.7082	0.5827
256	tanh	0.001	128	1,964,686	2000	15.9403	15.9042	0.428
256	tanh	0.001	256	1,964,686	2000			
512	ReLu	0.1	256	3,927,182	2000	Loss >1000		
1024	ReLu	0.1	128	7,852,174	2000			
4096 ,512	ReLu	0.001	128	25,650,830	2000	14.851	15.6651	0.567
4096 ,1024	ReLu	0.001	128	28,869,774	2000	14.6706	15.2609	0.579
4096 ,2048	ReLu	0.001	128	35,307,662	2000	14.6575	15.2323	0.588
4096 ,4096	ReLu	0.001	128	48,183,438	2000	14.6418	15.603	0.585
512,512	tanh	0.001	128	4,189,838	2000	13.9855	14.4883	0.586
512,512	relu	0.001	128	4,189,838	2000	14.6093	15.2659	0.578
512 ,1024	tanh	0.001	128	5,573,774	2000	14.0139	14.5559	0.586
512,256	tanh	0.001	128	3,497,870	2000	14.0394	14.5204	0.5803
512,256	relu	0.001	128	3,497,870	2000	14.4378	14.835	0.574
512,256	tanh	0.001	256	3,497,870	2000	13.9946	14.4783	0.5834
1024 ,512	tanh	0.001	256	7,255,694	2000	13.9641	14.5188	0.5841
1024 , 512	tanh	0.001	128	7,255,694	2000	13.9787	14.5275	0.5873
256,128	tanh	0.001	256	1,717,262	2000	14.1439	14.5887	0.5639
256,128	tanh	0.001	128	1,717,262	2000	14.2262	14.651	0.5708
2048 , 1024	tanh	0.001	128	15,557,774	2000	13.9817	14.5079	0.5846
256,128,64	tanh	0.001	256	1,585,358	2000	14.0541	14.5352	0.5802
1024 , 512 , 256	tanh	0.001	256	6,826,382	2000	13.9554	14.5217	0.5858
2048 ,1024 , 512	tanh	0.001	256	14,961,294	2000	13.9901	14.4971	0.5865
8192 , 4096 , 2048	tanh	0.001	256	91,295,886	2000	13.9943	14.5519	0.5869
8192 , 4096 , 2048	tanh	0.001	128	91,295,886	2000	14.0179	14.5849	0.5874
1024 , 512 , 256	tanh	0.001	128	6,826,382	2000	14.1503	14.6338	0.5829
1024 , 512 , 256	relu	0.001	128	6,826,382	2000	14.5671	14.9979	0.5848
1025 , 512 , 256	relu	0.001	256	6,826,382	2000	14.2895	14.9146	0.5535

The DNN model was run with a variety of different architecture. The number of hidden layers in the neural network varied from 1 to 3. Similarly, the number of neurons was altered in each hidden layer in order to find the best possible accuracy. Changes were also made in the activation function, batch size, and learning rate before feeding the dataset to the AI model. The results for various iterations of architecture are provided in Table-2 above.

After numerous runs, it was found that tanh activation function provides better results than ReLu. Also, the learning rate of 0.1 was found to be too high for the model as it gave huge losses. The runs also showed that the number of neurons beyond 512 in each layer increases the loss function.

It was seen from the runs above that the loss stagnated at around 14%, with the best loss coming out at 13.9641% on the training set for a 2 hidden layer neural network using tanh function. The R-squared error for all the runs also loitered around 0.58. This is in accordance with the loss shown in various runs.

CHAPTER VI
CONCLUSION

As Solar Ponds will play an integral role in the surge of a non-conventional energy-driven world, this study poses a convenient approach for the characterization of a solar pond. This dissertation was successful in scouting an algorithm that best suits the dataset.

The MAPE and R-squared metrics underline that the RandomForestRegression algorithm gives near-perfect results with the solar pond dataset. The DNN algorithm although gives perfect predictions for the temperature of UCZ, it cannot map similar accurate predictions for the LCZ temperatures. This presents the fact that the weights assigned to different parameters are accurate but goes wrong in the bias terms. This grey area requires further in depth study in the future.Even though the final dataset fed to the AI consists of 2000 shuffled rows from each location, the AI model gives slightly better results for certain locations than the others. Care is to be taken in creating the dataset and pre-processing it using suitable filters in order to obtain the best possible accuracy.

The study is proof of the fact that AI in Mechanical Engineering and Renewable Energy has potential for immense growth owing to its ability to massively reduce computational times and provide near-accurate results and predictions.

CHAPTER VII
FUTURE SCOPE

Future Developments that can be applied to the current project include:-

- Identifying the problems related to the low accuracy provided by the DNN architecture in predicting the values.
- Finding out various other regression algorithms that can be used to solve the particular problem & analyzing the results to find out the Best Algorithm and its corresponding best Architecture.
- The current AI Model can be extended further to find out the maximum amount of heat that can be extracted from the LCZ. An additional dataset needs to be provided through which the Model can adapt to the formulae involved in finding out the maximum heat that can be extracted.
- Developing a FrontEnd Platform wherein the user enters the most basic inputs such as the Dimensions of the Pond, & the dates for which the values need to be predicted. The platform must find the Meteorological Data from the NASA website and feed it into the AI model as inputs with the output being an Excel file consisting of the predicted parameters.
- Similar AI models can be extended to solve various other Mechanical Engineering problems which will result in a decrease in computational time by a huge amount.

REFERENCES

1. Kurt, Huseyin & Atik, Kemal & Ozkaymak, M. & Binark, A.. (2007). Artificial neural network approach for evaluation of temperature and density profiles of salt gradient solar pond. Journal of the Energy Institute. 80. 46-51. 10.1179/174602207X171570.
2. Platikanov, Stefan & Tauler, Roma & Cortina, Jose & Valderrama, César. (2021). Multivariate analysis of the operational parameters and environmental factors of an industrial solar pond. Solar Energy. 223. 113-124. 10.1016/j.solener.2021.05.045.
3. Abhishek Kumar, Kuljeet Singh, Sunirmit Verma, Ranjan Das, Inverse prediction and optimization analysis of a solar pond powering a thermoelectric generator, Solar Energy, Volume 169,2018, Pages 658-672, ISSN 0038-092X, https://doi.org/10.1016/j.solener.2018.05.035.
4. Verma, Sunirmit & Das, Ranjan. (2019). Wall profile optimization of a salt gradient solar pond using a generalized model. Solar Energy. 184. 356-371. 10.1016/j.solener.2019.04.003.
5. Abdullah, Abdullah & Lindsay, Kenneth. (2017). Assessing the Maximum Stability of the Nonconvective Zone in a Salinity-Gradient Solar Pond. Journal of Solar Energy Engineering. 139. 10.1115/1.4036773.
6. Andrews, J. & Akbarzadeh, Aliakbar. (2005). Enhancing the thermal efficiency of solar ponds by extracting heat from the gradient layer. Solar Energy - SOLAR ENERG. 78. 704-716. 10.1016/j.solener.2004.09.012.
7. Aramesh, Mohamad & Pourfayaz, Fathollah & Kasaeian, Alibakhsh. (2017). Transient heat extraction modeling method for a rectangular type salt gradient solar pond. Energy Conversion and Management. 132. 316-326. 10.1016/j.enconman.2016.11.036.
8. Jaefarzadeh, Reza. (2006). Heat extraction from a salinity-gradient solar pond using in pond heat exchanger. Applied Thermal Engineering - APPL THERM ENG. 26. 1858-1865. 10.1016/j.applthermaleng.2006.01.022.
9. Khaki, Saeed & Wang, Lizhi. (2019). Crop Yield Prediction Using Deep Neural Networks. Frontiers in Plant Science. 10. 10.3389/fpls.2019.00621.

10. Li, SH. (2003). Wind power prediction using recurrent multilayer Perceptron neural networks. 2330 Vol. 4. 10.1109/PES.2003.1270992.
11. Iwendi, Celestine & Bashir, Ali & Pasupuleti, Naga & Radha, Suja & Chatterjee, Jyotir & Peshkar, Atharva & Mishra, Rishita & Pillai, Sofia & Jo, Ohyun. (2020). COVID-19 Patient Health Prediction Using Boosted Random Forest Algorithm. Frontiers in Public Health. 8. 10.3389/fpubh.2020.00357.
12. Chakrabarty, Shyamal & Wankhede, Uday & Shelke, Rupesh & Gohil, Trushar. (2020). Investigation of temperature development in salinity gradient solar pond using a transient model of heat transfer. Solar Energy. 202. 32-44. 10.1016/j.solener.2020.03.052.
13. Khoshvaght-Aliabadi, Morteza & Feizabadi, Amir. (2020). Employing wavy structure to enhance thermal efficiency of spiral-coil utilized in solar ponds. Solar Energy. 199. 552-569. 10.1016/j.solener.2020.02.059.
14. Verma, Sunirmit & Das, Ranjan & Kumar, Abhishek. (2019). Transient thermal modelling and optimization of a solar collector-type pond considering an improved decay of radiative intensity. International Journal of Thermal Sciences. 139. 440-449. 10.1016/j.ijthermalsci.2019.02.008.
15. A. El Mansouri, M. Hasnaoui, A. Amahmid, R. Bennacer, Transient modeling of a salt gradient solar pond using a hybrid Finite-Volume and Cascaded Lattice-Boltzmann method: Thermal characteristics and stability analysis, Energy Conversion and Management, Volume 158,2018, Pages 416-429, ISSN 0196-8904, https://doi.org/10.1016/j.enconman.2017.12.085.