Deep Learning Under the Clouds: Weather Forecasting Using Keras Model with SimpleRNN for Precision Metrological Insights

Shivam Kumar[1], Adarsha Kumar[2]
[1,2] Dept. of Computer Science and Engineering Kalasalingam Academy of Research and Education Krishnankovil,TamilNadu,India

Abstract—Weather forecasting is the software of technology and era to expect the country of the environment for a given region. Ancient climate forecasting techniques usually depend on determined patterns of occasions, additionally termed pattern recognition. Agriculture, transportations, and energy are sectors that depend on heigh resolution weather models, which typically consumes many hours of large Heigh Performance Computing (HPC) systems to deliver timely results. Many users cannot afford to run the desired resolution and are forced to use low resolution output. In this paper we proposed to evaluate a strategy based on a deep learning neural network to learn a heigh-resolution representation from low- resolution predictions using weather forecasting as a practical use case. We take deep learning approach using SimpleRNN method to obtain our output. The device will expect weather based on parameters with temperature, pressor and humidiy.

Keywords—Deep Learning, Recurrent Neural Network, Adam optimizer, Keras Model, SimpleRNN, RMSprop Optimizer.

I. INTRODUCTION

Weather simply refers to the state of air on Earth at a given place and time. It is a continues, data-intensive, multidimensional, dynamic and chaotic process. These characteristics make weather forecasting a huge challenge. Forecasting is the process of estimating unknown unknown situation form historical data. Weather forecasting is one of the most scientifically and technologically challenging problems worldwide. Indeed, making an accurate forecasting is one of the main challenging facing meteorologists around the world. Since ancient times, weather forecasting has been one of the most interesting fields. Scientists have tried to predict meteorological characteristics using a number of methods, some of which are more accurate than others. Knowledge of meteorology forms the basis of the science of weather forecasting, which revolves around predicting the state of the atmosphere for a given location. Weather forecasting is often made by collecting quantitative data about the current state of atmosphere and using scientific knowledge atmospheric process to predict how the atmosphere will evolve in the future.

II. LITERATURE REVIEW

Forecasting the Future: A Deep Learning Approach for Accurate Weather Predection. In this the author has discusssed about the Weather forecasting with the implementation of Naïve Bayes Algorithm of [1].

DeepDownscale: a deep learning strategy for high-resolution weather forecast. In this article the authors have proposed a Convolutional Neural Network that takes as input resolution weather data and interpolates it into a high-resolution output. The features are the output of a set of global models for we crop a region of interest. These levels are observations in high-resolution. The benefits of the proposed neural networks, the authors made predictions and compared them to standard procedures of downscale [2]. Analog Forecasting of Extreme-Causing Weather Patterns Using Deep Learning. The authors have implemented the prediction models with a data-driven framework that is based on analog forecasting (prediction using past similar patterns) and employes a novel deep learning pattern-recognition technique (capsule neural networks, CapsNets) and an impact-based auto labeling strategy. The result shows the promises of multivariate data-driven frameworks for accurate and fast extreme weather predictions, which can potentially augment numerical weather predictions efforts in providing early warnings [3]. Deep Learning-Based Effective Fine-grained weather forecasting model. The authors have used the well-known numerical weather prediction (NWP) models to solve complex equations to obtain a forecast based on current weather conditions. In this article long short-term memory (LSTM) and temporal convolutional networks (TCN) have used. The experiments shows that the proposed lightweight model produces better results compared to the well-known and complex

WRF model, demonstrating its potential for efficient and accurate forecasting up to 12h [4]. A real-time weather forecasting and analysis, the authors have used ARIMA model that gives better results for time-series data is used to predicting the values for forthcoming [5]. Weather forecasting using deep learning techniques. In this article the authors had compared prediction performance of Recurrence neural Network (RNN), Conditional Restricted Boltzmann machine (CRBM), and Convolutional Network (CN) models. Those models are tested using weather data set provided by BMKG (Indonesian Agency for Meteorology, Climatology, and Geophysics). Forecasting accuracy of each model is evaluated using Frobenius norm [6]. Short-term local weather forecast using dense weather station by deep neural network. The authors propose a very-sort-term, i.e., less than 1-hour weather forecast method. This paper outperforms the existing state-of-the-art methods such as XGBoost and support vector machines using a large real observed data [7]. Daily weather Forecasting based on Deep Learning Model: A case study of Shenzhen City, China. The authors used a long short-term memory (LSTM) neural network for the Shenzhen daily weather forecast. The experimental results shows that their design of the EMD-LSTM model has higher forecasting precision and efficiency than traditional models, which provides new ideas for he weathers forecast [8]. A hybrid firefly algorithm with particle swarm optimization for energy efficient optimal cluster head selection in wireless sensor networks. In this article the authors have dealed with Wireless Sensors Networks (WSN), Clustering techniques, and a Hybried approach of Firefly Algorithm with Particle Swarm Optimization (HFAPSO) [9].

III. METHODOLOGY

Our proposed methodology is to made neural network with one input layer with three neurons, two hidden layers and one output layer with three neurons consisting of temperature, pressor and humidity (refer *figure 1).* The weight and biased will be automatically update by activation function (*shown in Table1*).

Formula for activation function (sigmoid/logistic)
$$f(x) = \frac{1}{1 + e^{-x}}$$

Table 1.

A. Algorithms

For forecasting the weather of the given place, we are using Recurrent Neural Network (RNN) is a layer used in deep learning models, particularly for sequence data analysis.

Here, we are especially applying SimpleRNN() used for predicting the output.

B. Methods

- Keras Models, is a neural network architecture for deep learning. It simplices building, training, and deploying models. It consists of layers that define the model's structure, specifying how dta is processed to make predictions or classifications.

- SimpleRNN(), In Keras models the 'SimpleRNN' layer is part of recurrent module and can be easily added to a neural network model. The SimpleRNN layer excels at processing sequential data making it particularly suited for time-series forecasting tasks such as weather prediction. By understanding the intricate patterns and relationships within historical weather data, our Keras model aims to provide more accurate and granular insights into future meteorological conditions.

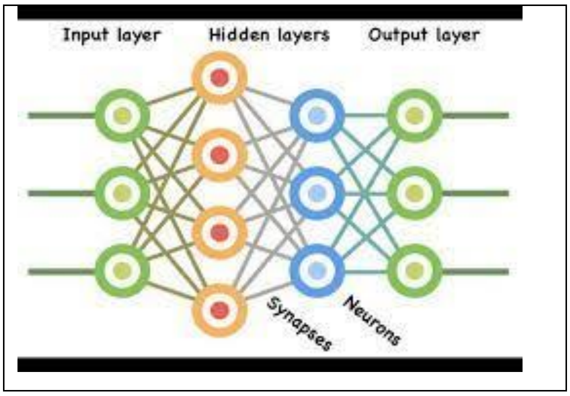


Figure 1. (source- google)

The above Fig. 1 show the architecture of neural network which is the basic concept of building RNN model for weather forecasting. This neural network is implemented with three input temperature, pressure and humidity and output as predicted temperature, pressure and humidity

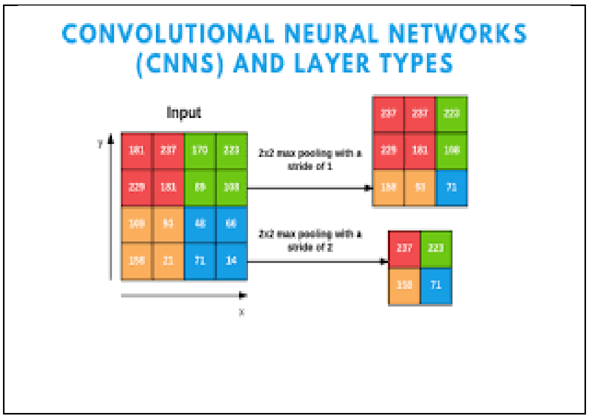


Figure 2. (source- google)

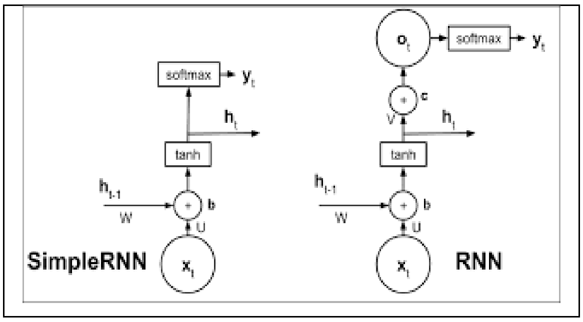
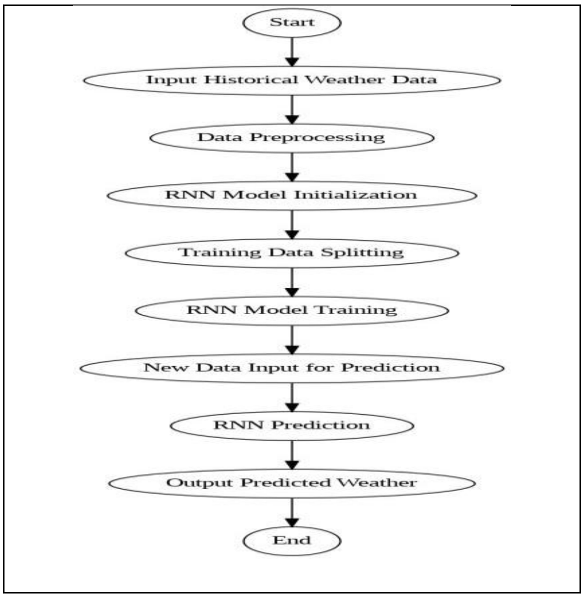


Figure 3. (source- google)

The above *figure 1* shows the neural model of our project having input as temperature, pressure and humidity and output with predicted temperature, pressure and humidity and *figure 3* represents the architecture of SimpleRNN.

IV. IMPLEMENTATION

We mentioned below step-by-step procedure of implementation through a flowchart.



Flow chart

A. Data Sets

Our data set contains three different types data sets temperature, pressure and humidity. Each data set consisting of more than fifteen attributes at the same time. We took temperature of different cities at the same time likely for pressure and humidity.

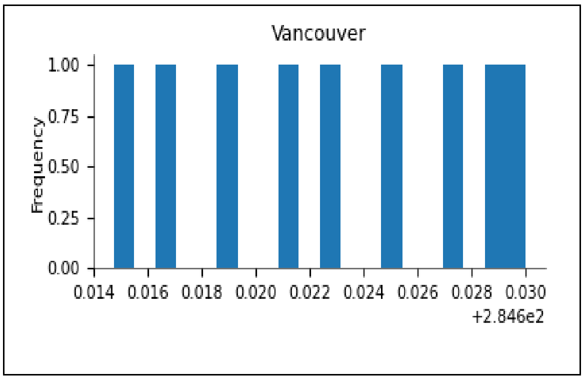


Figure 4. (temperatures in Vancouver city)

The above Fig. 4 shows the frequency of temperature in the Vancour city that means the how many times same temperature is present in our temperature data set.

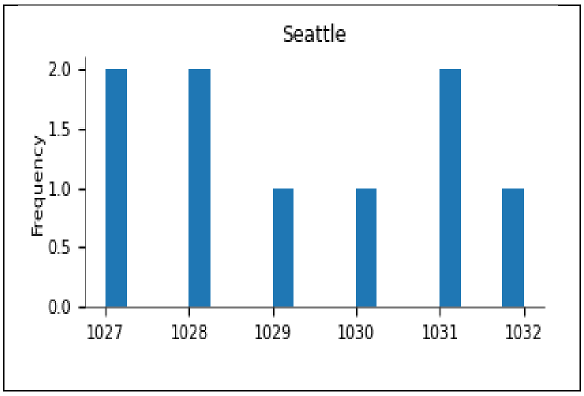


Figure 5. (pressure in Seattle city)

The above Fig. 5 shows the frequency of pressure in the Seattle place of the first 10 rows of temperature data set.

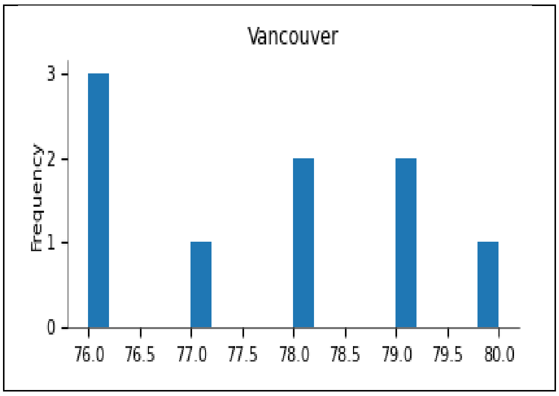


Figure 6. (humidity in Vancouver city)

Likely, the Fig. 6 shows the frequency of humidity in the Vancouver city of the first 10 rows of our humidity data set.

B. Training and Testing of Data Points

We choose Tp = 7000 here which means we will train the RNN with only first 7000 data points and then let it predict the long-term trend (for the next > 35000 data points or so). That is not a lot of training data compared to the number of test points.

Plotting the training of humidity, temperature and pressure are shown in figure *7, figure 8* and *figure 9* respectively.

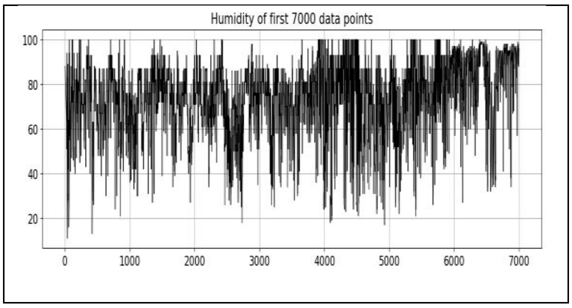


Figure 7. (plotting of humidity data points)

The above Fig. 7 represents the graph of training of RNN over the first 7000 data points of humidity data set.

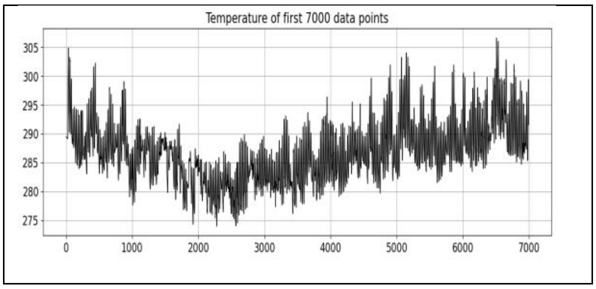


Figure 8. (plotting of temperature data points)

The above Fig. 8 represents the training of RNN model over the first 7000 data points of temperature data set.

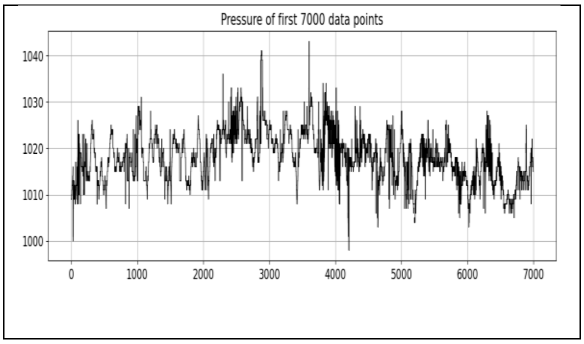


Figure 9. (plotting of pressure data points)

Likely, the Fig. 9 represents the graph of training of RNN model over the first 7000 data points pressure data sets

Training over 7000 and testing over 38252 sets of data points and plotting the graph together. Blue color represents training and yellow color represents testing portion as shown in *figure 10.*

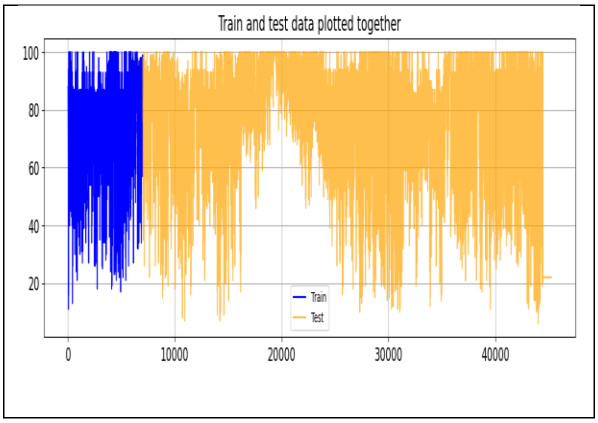


Figure 10.

The above Fig. 10 shows the graphical representation of training and testing of RNN model.

C. Building Models

Keras model with SimpleRNN layer, we build a simple function to define the RNN model. It uses a single neuron for the output layer because we are predicting a real-valued number here. As activation, it uses the ReLu function, following arguments are supported.

- Neurons in the RNN layer.
- Embedding length (i.e., the step length we chose).
- Neurons in the densely connected layer.
- Learning rate.

After creating the model function, we created a simple Callback class to print progress of the training at regular epoch interval. We are training our model for 1000s epochs, so loss metric available in the history attribute of the model is MSE loss and taking the square-root of MSE to compute the RMSE. The above steps will be repeat for our all-inputs temperature, pressure and humidity. So, we are plotting RMSE for each model of each inputs temp, pressure and humidity shown in *figure 11, 12 and 13.*

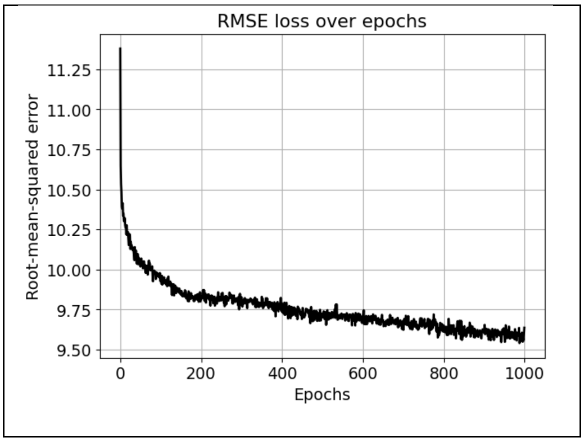


Figure 11. (RMSE loss for humidity models)

The above Fig. 11 shows the 2-dimensional representation of RMSE loss over epochs while training our humidity models.

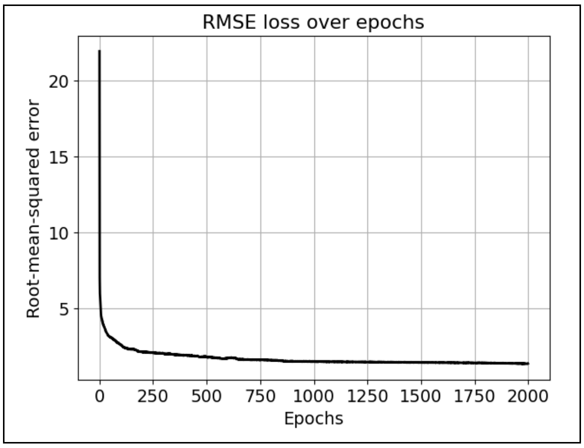


Figure 12. (RMSE loss for temperature models)

The above Fig. 12 shows the 2-dimensional representation of RMSE loss over epochs while training our temperature models.

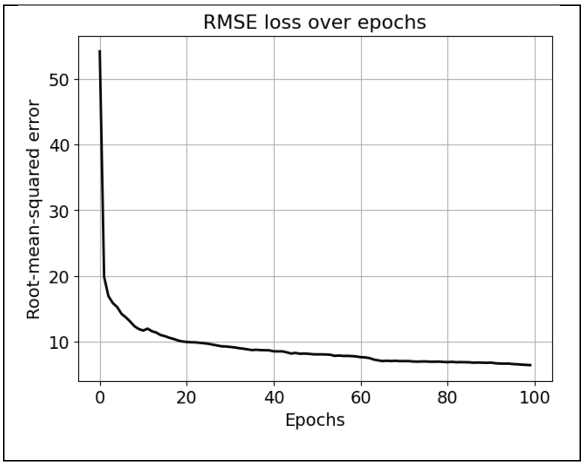


Figure 13. (RMSE loss for pressure model)

The above Fig. 13 shows the 2-dimensonal representation of RMSE loss over epochs while training our temperature models.

V. RESULT AND ANALYSIS

We are emphasizing and showing again what exactly the models see during the training. The models were fitted with trainX which is plotted below *(figure 14.)* and trainY which is just the 8-step shipped and shaped vector.

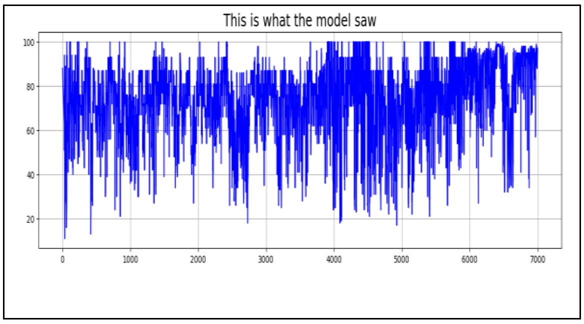


Figure 14. (trainX model)

The above Fig. 14 shows the actual weather at the ground. Now, we can generate predictions for the future by passing testX to the trained models.

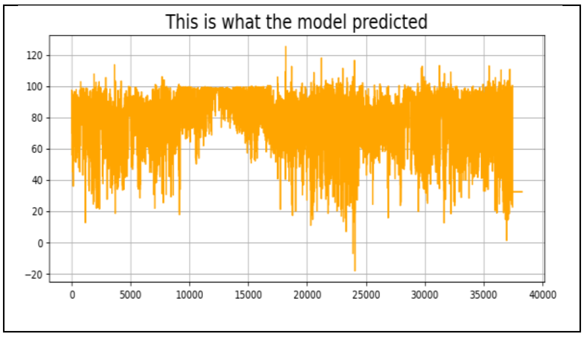


Figure 15. (predicted model for humidity)

The above *figures 14 &15* show the trained (ground truth) and predicted model for humidity respectively. Now we are plotting the ground truth and model predictions together below in *figure 16.*

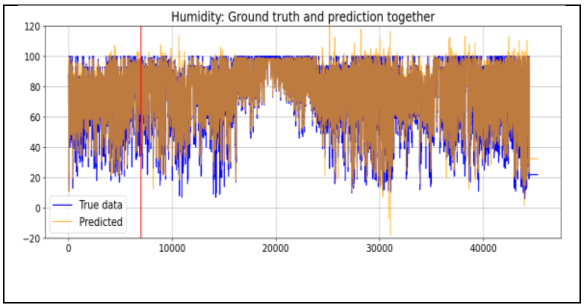


Figure 16. (humidity: ground truth and prediction together)

Modeling the other two data, since we have covered the modeling of humidity data step-by-step in detail, we will show the modeling with other two parameters- temperature and pressure. We already plotted RMSE for both. *(fig. 12 & 13)*

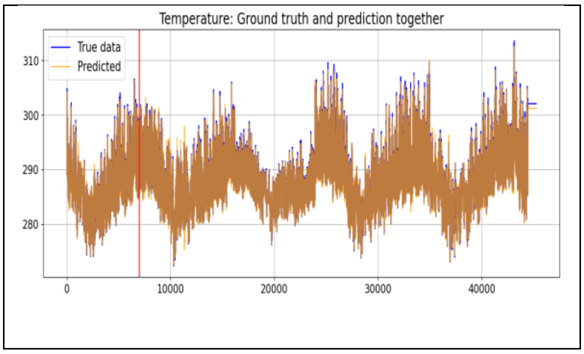


Figure 17. (temperature: ground truth predictions together)

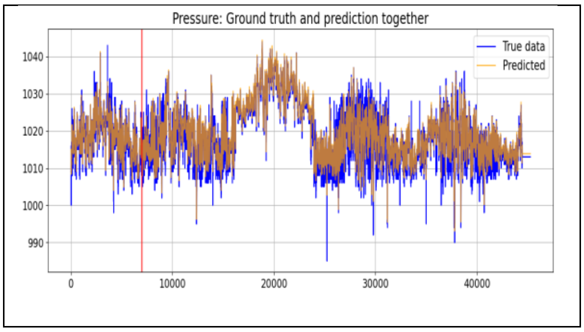


Figure 18. (pressure: ground truth and prediction together)

VI. CONCLUSION

We trained our all parameters using RNN method. We plotted the predicted vector; we see it matches closely the true values and that is amazing. Given how little training data was used and how far in the future it had to predict. Time-series techniques like ARIMA, Exponential smoothing, cannot predict very far into the future and their confidence interval quickly grows beyond being useful. We plotted the ground truth and model predictions together for all parameters temperature, pressure and humidity *(fig. 16, 17 &18)* to show that it follows the general trends in the ground truth pretty well. Considering less than 25% data was used for training, this short of amazing. The boundary between train and test splits is denoted by the vertical red line *(fig 16,17, & 18)*. Form the above figures, our models show the forecasting of future weather for three parameters- temperature, pressure and humidity at maximum accuracy.

REFERENCES

[1]. Venkata Ravi Kiran Kolla, "Forecasting the Future: A Deep Learning Approach for Accurate weather Prediction", vol. 6. USA, December 2018.

[2]. Eduardo R. Rodrigues, Igor Oliveira, Renato L. F. Cunha, and Macro A. S. Netto, "DeepDownscale: a deep learning stratergy for high-resolution weather forecast", IBM Research, Cornel university, August 2018.

[3]. Ashesh Chattopadhyay, Ebrahim Nabizadeh, and Perdram Hassanzadeh, "Analog Forecasting of Extreme-Causing Weather Pattrens Using Deep Learning", Depatrment of mechanical Engineering, Rice University, Tx, USA, Department of Earth, Environmental and Planetary Sciences, Rice University, TX, USA, Feberuary 2020.

[4]. Pradeep Hewage, Marcello Trovati, Ella Pereira, and Ardhendu Behera, "deep learning-based effective finr-grained weather forecasting model", Department of Computer Science, Edge Hill University, Ormskirk, Lancashire L39 4QP, UK, June 2020.

[5]. Sushmitha Kothapalli, and S. G. Totad, "A real-time weather forecasting and analysis", Department of Computer Science and Engineering, Hubballi, India, IEEE, June 2018.

[6]. Afan Galih Salman, Bayu Kanigoro, and Yaya Heryadi, "Weather Forecasting Using Deep Learning Techniques", School of Computer science, Bina Nasantara University, Jakarta, Indonesia, IEEE, October 2015.

[7]. Kazuo Yonekura, Hitoshi Hattori, and Taji Suzuki, "short-term local weather forecasting using dense weather station by deep neural network", Mathematical and Computational Engineering Department, IHI Corporation, Yokohama, Japan, IEEE, December 2018.

[8]. Guici Chen, sijia Liu, and Feng Jian, "Daily Weather Forescasting Based on deep Learning model: A Case study of Shenzen City, China", College of Science, Wuhan University of Science and Techonology, Wuhan $30081, China, vol. 13, MDPI, August 2022.

[9]. Pitchaimanickam and Murugaboopathi, "A hybrid firefly algorithm with particle swarm optmization for energy effficient optimal cluster head selection in wireless sensor networks", Department of Computer Science and Engineering, Kalasalingam Academy of Research and

Education, Krishnankoil, Virdhunagar, Tamil Nadu 626126, India, Scopus, Volume 32. , Issue 12. , June 2020.