iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 12 **Issue:** I **Month of publication:** January 2024

DOI: https://doi.org/10.22214/ijraset.2024.57959

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 12 Issue I Jan 2024- Available at www.ijraset.com

An Approach to Weather Forecasting using Additive and Deep Learning Algorithms

Shivam Balikondwar[1], Sunil Kanobaji Moon[2]
SCTR's Pune Institute of Computer Technology

Abstract: ***Weather is an important factor that directly affects farming activities. The temperature and humidity of a region play a crucial role in the type of crop to be cultivated. Thus, It becomes important for farmers to know the future trends in the weather to plan farming activities. There are several classical weather prediction services but all of them depend on complex modelling and are operated by government authorities. In this case, time series becomes an effective way of forecasting these temperature trends as it not only requires minimal computational resources but only the past weather data to achieve results. In this paper, we study the process of implementing both additive and regression-based models and compare their performance to decide which approach is the best for weather prediction.***
Keywords: AR, MA, ARIMA, LSTM, RMSE, MAE, MSE

I. INTRODUCTION

Time series data refers to that type of data containing observations collected over subsequent timestamps. The core of time series analysis is using past observations to predict future values without considering any additional factors. The research and applications of time series have become a hotspot in recent years due to its various applications from stock market index predictions to real estate price predictions.

One of the applications of time series data is in weather prediction. Weather conditions are rapidly changing around the world [1] and forecasting these trends is a very exhaustive task. The current weather prediction approach relies heavily on complex physical models and the forecasts from these models require tremendous data extracted from various factors like satellite weather reports, precipitation reports [2] and other factors. The type of equipment needed for this is also not readily available to the public, thus forecasts are mostly done by government or non-profit authorities. Prediction using time series is not a new problem but with the advent of deep learning models and complex statistical models like ARIMA, they can be a cheaper yet viable alternative to predict future trends in weather data. Moreover, these predictions can be combined with regular forecast data to achieve rolling forecasts of time series data which are the most accurate form of predictions.

The weather is a seasonal quantity which is stationary as it repeats itself after a certain period in time. The deep learning models and statistical time series models like ARIMA and Prophet exploit these seasonal trends in the weather data to learn complex seasonal patterns and accurately predict the weather parameters.

During writing this article we studied various research methods and selected three models namely Prophet, ARIMA and LSTM for weather prediction. The first two techniques are additive models while the last one is a memory-based deep learning model. This study helps compare the performance of both categories of ML models. The forecasted values versus actual values will help us decide if time series approaches are significantly cheaper and more reliable to develop and would replace the classical weather forecast approaches.

The paper is structured as follows: Section II includes information on the methodology of data processing. Section III describes the different models used, Section IV discusses the methodology to train the models for the weather forecast system. Then, Section V provides the results of our model and finally, Section VI summarizes the conclusion.

II. DATA COLLECTION AND PREPROCESSING

For our research problem, we selected the prediction of temperature and humidity weather parameters since they are the most basic yet crucial weather parameters. The weather data of the Delhi region was considered as it is the capital city and has quite a wide variation in temperature trends throughout the year.

The weather data is collected from The Indian Meteorological Department which indexes the daily weather data for different Indian cities.

©IJRASET: All Rights are Reserved

A. Data Preprocessing

We fetched the temperature and humidity data for the past 20+ years (1996-2018) which consisted of hourly data readings as seen in Fig. 1. The hourly data readings are quite nuanced and are difficult to train models for weather prediction over long periods in future. Hence, we replace the hourly data with the mean average temperature and humidity of the data readings. We also observed that the temperature and humidity readings were missing for a few days. These missing values have been replaced with the weekly mean value of the weather data as shown in Fig. 2.

datetime_utc	humidity	temperature
1996-11-01 11:00:00	27.0	30.0
1996-11-01 12:00:00	32.0	28.0
1996-11-01 13:00:00	44.0	24.0
1996-11-01 14:00:00	41.0	24.0
1996-11-01 16:00:00	47.0	23.0

Fig. 1 Hourly weather data

datetime_utc	humidity	temperature
1996-11-01	52.9	22.3
1996-11-02	48.6	22.9
1996-11-03	56.0	21.8
1996-11-04	48.1	22.7
1996-11-05	29.4	27.8

Fig. 2 Daily weather data replaced by hourly mean

We split the data into two sets for training and testing with the time series from the 1996-2018 year being used for training the models and the time series from the 2018-2020 year used to verify our prediction results achieved from the models.

III.MODELS

There are various methods for time series forecasting both regression and deep learning models. According to the survey paper by Zhenyu Liu and his team [3] ARIMA is the most widely used method for time series forecasting. Hence, we selected ARIMA as our baseline model to compare against our independent suggestions. Among the forecasting approaches, there are ANNs, SVM, Fuzzy time series for forecasting and RNNs. A major challenge is the prediction of weather parameters over a long duration for which LSTMs [4] have proven to be ideal. The Prophet model is also an additive model like ARIMA and provides automatic hyper-tuning parameter selection. Thus, we collectively implement ARIMA, Prophet and LSTM approaches and compare their performance.

A. Autoregressive Integrated Moving Average (ARIMA)

ARIMA is a set of statistical analysis models [5] that forecasts the time series results by examining the past values on the three different parameters namely Autoregression (AR), Moving Average (MA) and order of differencing (I).

1) Autoregression (AR)

The auto-regression component is the model parameter whose predictions depend on the number of past values in the time series. The AR equation is as follows in Equation 1,

$$Yt = \beta\square * y\text{-}\square + \beta\square * y\square\text{-}\square + \beta\square * y\square\text{-}\square + \ldots + \beta\square * y\square\text{-}\square$$

—---- (1)

In the above equation, β represents the coefficient of the AR model and the value of time series at 't' is based on various slots t-1, t-2,...,t-k of the past.

2) Moving Average (MA)

The time series at any given time might be impacted by errors in various past time slots. The moving average is the model which calculates these residuals of errors in past time series and depending on that it calculates the future values.

The MA equation is as follows in Equation 2,

$Yt = \alpha\square * \text{Ɛ}\square\text{-}\square + \alpha\square * \text{Ɛ}\square\text{-}\square + \alpha\square * \text{Ɛ}\square\text{-}\square + \ldots + \alpha\square * \text{Ɛ}\square\text{-}\square$

—---- **(2)**

In the above equation, α represents the coefficient of the MA model and the error residuals at each past time instance 't' is denoted by Ɛ of the respective past instance.

3) Order of Differencing (I)

To apply the time series models we must first ensure the dataset is stationary. The term stationary refers to the property of the data to repeat itself in trend after periodic intervals in time.

If the dataset is not stationary, it can be made stationary by subtracting the past 't-1' values for any 't' value. The number of times this differencing is needed is the term order of differencing.

Thus, considering the above components the equation for ARIMA can be written as follows in Equation 3,

$Yt = \beta\square * y\square\text{-}\square + \alpha\square * \text{Ɛ}\square\text{-}\square + \beta\square * y\square\text{-}\square + \alpha\square * \text{Ɛ}\square\text{-}\square + \beta\square * y\square\text{-}\square + \alpha\square * \text{Ɛ}\square\text{-}\square + \ldots\ldots\ldots\ldots + \beta\square * y\square\text{-}\square + \alpha\square * \text{Ɛ}\square\text{-}\square$

—---- **(3)**

B. Prophet

It is a time series forecasting method developed by the Data Science team at Meta [6] in 2017. It is an additive model whose predictions depend collectively on namely three components: trend, seasonality and holidays as shown in below Equation 4.

$Y_t = G_t + H_t + S_t + \text{Ɛ}\square$ —---- **(4)**

1) *Trend (G):* The trend term deals with the change in trends of the seasonal data required to fit a proper curve from the training set.
2) *Holidays (H):* The holiday term refers to any missing values (holidays) in the data. If there are no missing terms then its value is 0.
3) *Seasonality (S):* The seasonality term deals with the periodic nature of the data. Unlike ARIMA it is not the order of differencing but simply the 'repetition in trend' that might appear. E.g.: Seasons repeat every 365 days in the dataset.
4) *Error term (Ɛ):* The error term is the parameter calculated by the model while fitting the model with training data.

C. Long Short-Term Memory (LSTM)

LSTM is a recurrent neural network (RNN) architecture-based deep learning model [7]. It excels at capturing long-term dependencies in the data thus making it ideal for predicting time series data which contains exhaustive historical time series data.

The different layers present in our model are:

1) *Convolution Layers:* These layers are responsible for extracting hidden features of our data.
2) *Max Pooling and Flatten Layers:* These layers are responsible for performing dimensionality reduction of our data so that the model can be trained efficiently.
3) *Bi-directional LSTM Layers:* These layers are responsible for knowing the forward and backward features in the sequence. It results in more accurate predictions in time series rather than normal LSTM which only focuses on past results.
4) *Dense Layers:* This is the end stage of LSTM and takes the output of the past LSTM layers and converts it to a higher dimension vector suitable for predicting the results.

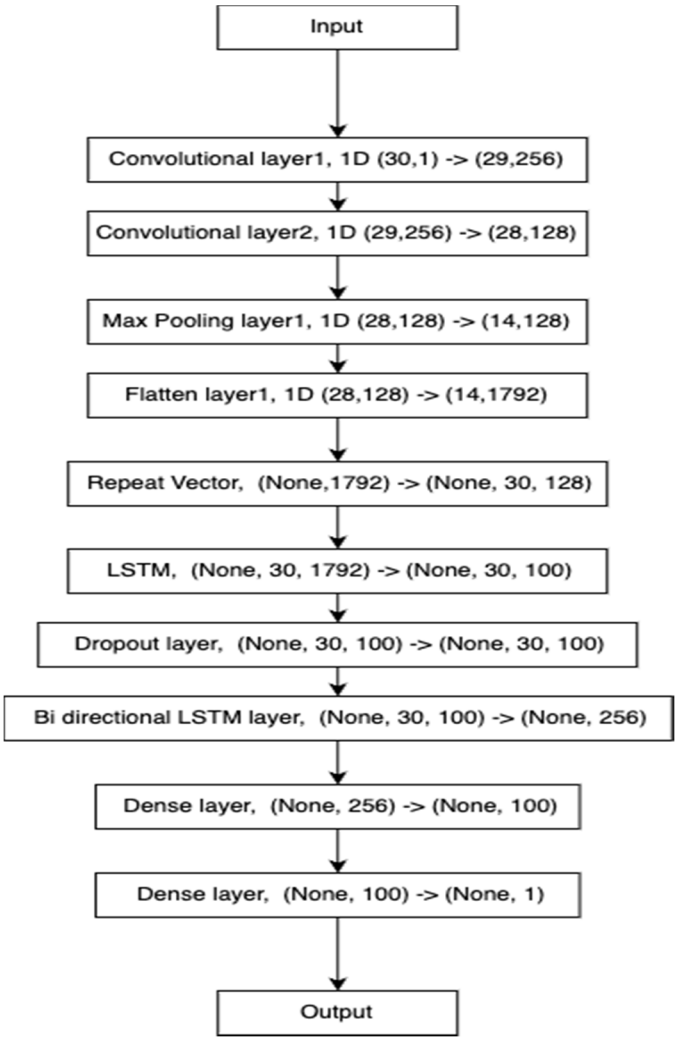


Diagram. 1 Layers of Bi-directional LSTM

We constructed an LSTM model consisting of 10 layers with a combination of the different layers as demonstrated in the Diagram. 1.

IV. METHODOLOGY

To train different models we study different parameters and steps necessary to train each model.

A. ARIMA

To apply ARIMA we need to first ensure that our time series data is stationary as mentioned in the study by ZhiQiang Li and his team [8]. This can be done by plotting the rolling mean and standard deviation of the yearly data.

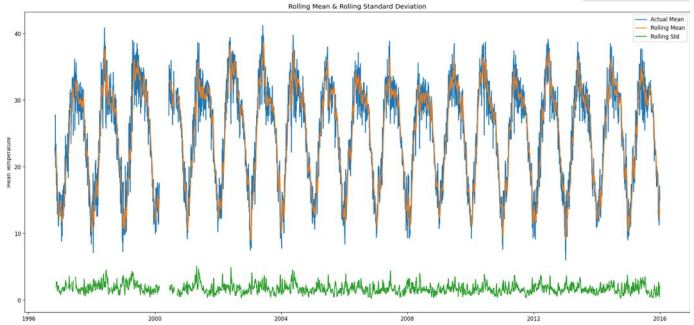


Fig. 3 Rolling Mean and Standard Deviation graph of yearly data.

As seen from our plotted graph it can be seen that the 'green' line which means the standard deviation remains constant with time thus confirming that the series is stationary.

On performing the parameter evaluation using 'pmdarima' library we tried fitting the ARIMA model with AR and MA values within the 0-3 range to find the least Akaike information criterion (AIC) score. AIC score defines how well the model fits with the training data. The AR term of the 3rd order, the MA term of the 1st order and the order of differencing is 0 providing us the least AIC.

TABLE I

Sr. no	ARIMA(p, d, q) parameters	AIC score
1	ARIMA(0,0,0)	72490.515
2	ARIMA(1,0,0)	30217.378
3	ARIMA(0,0,1)	44770.518
4	ARIMA(2,0,0)	30012.657
5	ARIMA(3,0,0)	29769.329
6	ARIMA(2,0,1)	29773.484
7	ARIMA(3,0,1)	29776.145
8	ARIMA(2,0,2)	29769.202
9	ARIMA(3,0,1)	29765.762

B. Prophet

Prophet has an inbuilt framework that provides the most suitable parameters and forecasts with parameters provided out of the box. For our data, we provide a holiday parameter whose value is 0 as in our data we don't have any missing data entries and the value of the period parameter is 365 days as we have yearly time series data. The trend and error term parameters are specified by the model internally.

C. LSTM

Our LSTM comprises of 10-layer deep learning model. For preparing data for the model, the first preprocessing that we do is normalizing the range of features in the data by using the Min-Max scalar. This normalizes the temperature and humidity time series data values within the range from -1 to +1. The second phase of data preparation that we do is converting the given normalized data into a sequence of past time series. We use the step size as past 30-day data points to predict the next data in the time series. Thus, to predict daily weather data, we prepare a list of time series sequences for each day entry containing the past 30-day time series values.

In our LSTM, the Dense layers are used with 'Relu' as the activation function. For our loss function we use the Mean Square Error function and the optimizer used to fit the models is the 'Adam' optimizer.

The three regression parameters, namely Mean Absolute Error, Root Mean Squared Error and Mean Squared Error are used to evaluate the performance of models.

1) *Mean Absolute Error (MAE):* MAE depicts and calculates the difference between the variables.
2) *Error of Root Mean Squared (RMSE):* RMSE calculates differences between the predicted values and the actual values. The difference is the prediction error incurred during the process in any model
3) *Error of Mean Squared (MSE):* MSE calculates the total square of the differences between the predicted values and the actual values. The difference is the prediction error incurred during the process in any model

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

V. RESULTS

Based on our findings, we used the training data to fit models and achieved the predicted vs actual plots for each model approach for both the temperature and humidity weather data.

The 'Blue' color line in plots represents the 'actual' values whereas the 'Red' color line in plots represents the 'predicted' values.

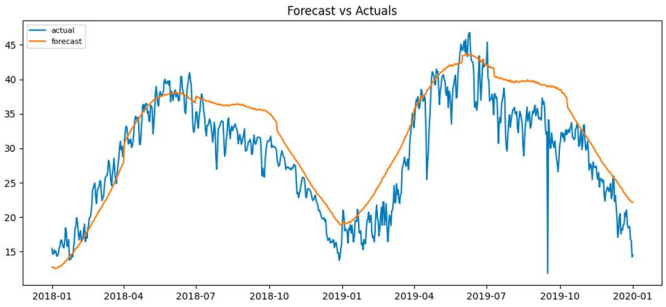


Fig. 4 Forecast vs Actual plot for temperature data using ARIMA

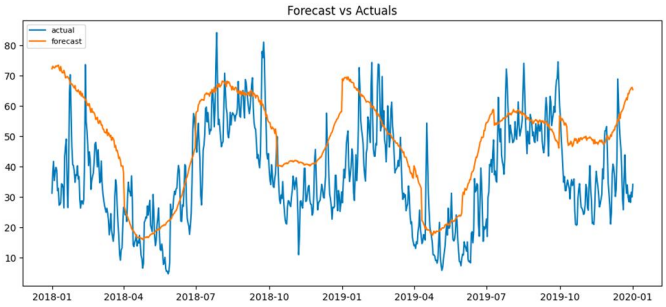


Fig. 5 Forecast vs Actual plot for humidity data using ARIMA

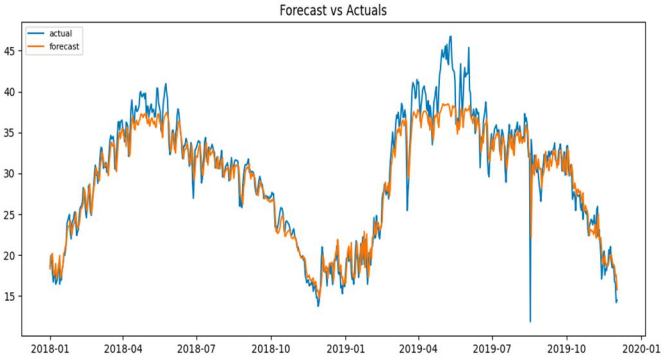


Fig. 6 Forecast vs Actual plot for temperature data using LSTM

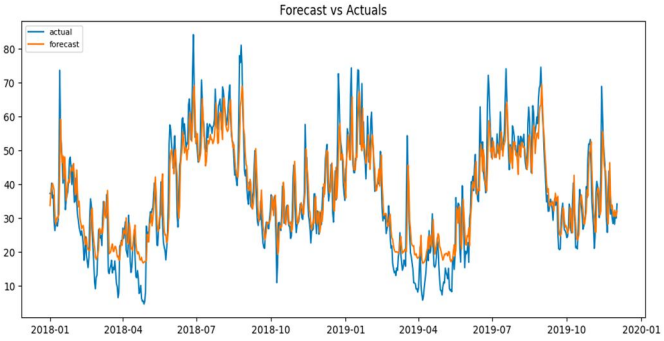


Fig. 7 Forecast vs Actual plot for humidity data using LSTM

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 12 Issue I Jan 2024- Available at www.ijraset.com

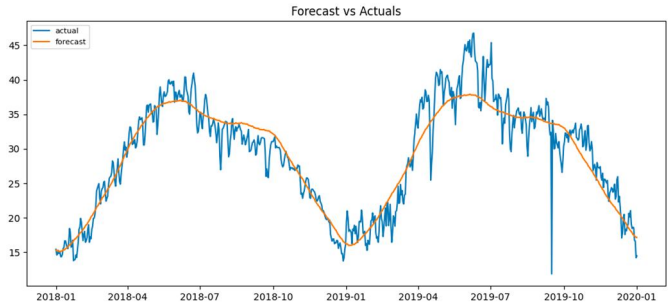


Fig. 8 Forecast vs Actual plot for temperature data using Prophet

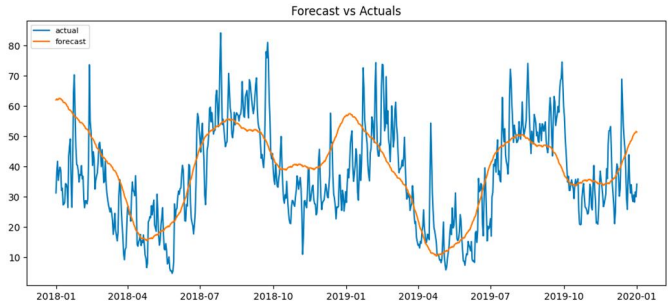


Fig. 9 Forecast vs Actual plot for humidity data using Prophet

TABLE II. Study of model performance metrics.

Model	Parameters	MAE	MSE	RMSE
ARIMA	temperature	3.346	17.283	4.157
	humidity	14.148	296.69	17.224
Prophet	temperature	2.021	7.441	2.728
	humidity	9.748	157.04	12.531
Bidirectional-LSTM	temperature	1.479	4.723	2.173
	humidity	5.126	55.578	6.676

VI. CONCLUSIONS

We implemented and compared both deep learning as well as regression models for forecasting temperature and humidity weather trends. On comparing the Fig. 4. and Fig. 5. plots with other figures we found that the ARIMA model showed the least accurate predictions. This was closely followed by Prophet which showed significant improvement as shown in Fig. 8 and Fig. 9. The Prophet model plots align well with the actual weather data values. Both of the regression models fail to observe granular details in the weather data that are only observed in deep learning models like LSTM. On observing the LSTM plots in both Fig. 6 and Fig. 7. we can safely conclude that deep learning models like LSTM are very advanced in the prediction of time series trends. The granular accuracy of LSTM is because it constantly uses past time sequence data to predict future trends acting as a rolling time prediction system which is generally considered the most accurate form of time series prediction.

The trend is further confirmed by comparing the performance error metrics of each model as shown in Table 2. Based on these metrics in comparison to our baseline ARIMA approach, LSTM exhibited the best performance with an MAE value of 1.479 degrees and 5.126 percentage relative humidity.

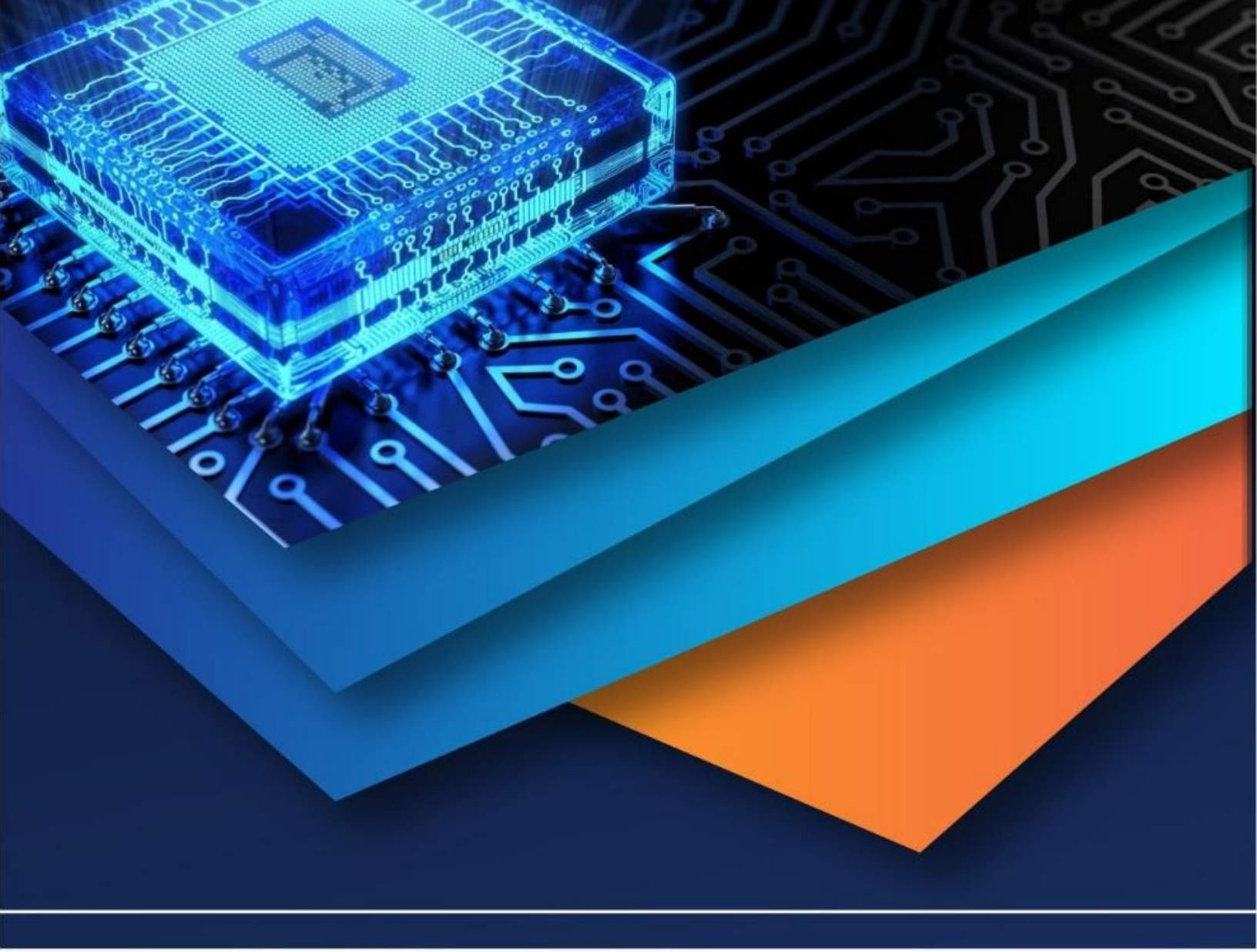
©IJRASET: All Rights are Reserved

Given the democratization of computing resources, deep learning models have become popular in recent years. The performance metrics exhibited by these models in the prediction of time series are impressive and can be considered as a viable and more practical approach to weather forecasting. These deep learning approaches can be further applied to various problems like stock value predictions [9] for finance sector industries.

REFERENCES

[1] F. V. Davenport and N. S. Diffenbaugh Using machine learning to analyze physical causes of climate change: A case study of U.S. Midwest extreme precipitation, Geophys. Res. Lett., vol. 48, no. 15, Aug. 2021 Art. no. e2021GL093787
[2] Krouma, M., Yiou, P., Déandreis, C., and Thao, S.: Assessment of stochastic weather forecast of precipitation near European cities, based on analogs of circulation, Geosci. Model Dev., 15, 4941–4958, https://doi.org/10.5194/gmd-15-4941-2022, 2022.
[3] Z. Liu, Z. Zhu, J. Gao and C. Xu, "Forecast Methods for Time Series Data: A Survey," in IEEE Access, vol. 9, pp. 91896-91912, 2021, doi: 10.1109/ACCESS.2021.3091162
[4] A. Srivastava and A. S, "Weather Prediction Using LSTM Neural Networks," 2022 IEEE 7th International conference for Convergence in Technology (I2CT), Mumbai, India, 2022, pp. 1-4, doi: 10.1109/I2CT54291.2022.9824268.
[5] J. Pant, R. K. Sharma, A. Juyal, D. Singh, H. Pant and P. Pant, "A Machine-Learning Approach to Time Series Forecasting of Temperature," 2022 6th International Conference on Electronics, Communication and Aerospace Technology, Coimbatore, India, 2022, pp. 1125-1129, doi: 10.1109/ICECA55336.2022.10009165.
[6] Taylor SJ, Letham B. 2017. Forecasting at scale. PeerJ Preprints 5:e3190v2. doi: https://doi.org/10.7287/peerj.preprints.3190v2
[7] Sepp Hochreiter, Jürgen Schmidhuber; Long Short-Term Memory. Neural Comput 1997; 9 (8): 1735–1780. doi: https://doi.org/10.1162/neco.1997.9.8.1735
[8] Z. Li, H. Zou and B. Qi, "Application of ARIMA and LSTM in Relative Humidity Prediction," 2019 IEEE 19th International Conference on Communication Technology (ICCT), Xi'an, China, 2019, pp. 1544-1549, doi: 10.1109/ICCT46805.2019.8947142.
[9] Kumar Prakhar, Sountharrajan S, Suganya E, Karthiga M Effective Stock Price Prediction using Time Series Forecasting IEEE 2022 6th International Conference on Trends in Electronics and Informatics (ICOEI)

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)