**Weather Forecasting Using Machine Learning Models and Multi-variate Decagon Framework Prediction Proposal**

**Levan Sulimanov, Aishwarya Kale, Sarang R. Patil, Spandana Chennamaneni,**

**Suvidha Vaddula, Sai Karteek. Edumudi**

**University of Maryland, Baltimore County (UMBC)**

**May 2021**

**ABSTRACT:**

Weather forecasting has long been a scientific challenge with significant social and economic implications due to the chaotic nature of the atmosphere, the massive volume of data and the complexities of the underlying physics. Traditionally, Weather forecasting predictions are executed with the help of massive & complex physics models that consider a multitude of atmospheric conditions over a long period of time. These models generally run-on hundreds of nodes in a large-scale High-Performance Computing (HPC) environment, which consumes a lot of energy and is a computationally expensive task. In recent years, machine learning and deep learning methods have shown significant advances in many research areas that deal with large datasets. In this work, we tackle a particular problem of weather forecasting that uses NOAA historical data to train simple machine learning and deep learning models which can provide reliable forecasts for specific weather conditions in the near future in a short amount of time. The models can be run in environments that are much less resource intensive. According to the WMO, the traditional weather forecasting requires gathering of weather data from 15 satellites, 100 stationary buoys, 600 drifting buoys, 3,000 aircraft, 7,300 ships, and some 10,000 land-based stations. Whereas the introduced model can forecast with the data only from weather stations. This reduces the data collection cost along with support & maintenance. We introduce RFR; Random forest regression and RNN, a neural network that forecasts temperature and extends our scope to Decagon Models Framework which can predict for all features like temperature, humidity etc.The models has capability to predict up to 24 hours into the future taking into account past 7 days atmospheric conditions. The results of this study demonstrate that the models' accuracy is satisfactory to be used alongside existing state-of-the-art techniques.

Keywords – Weather Forecasting, Data preprocessing, Time series, Machine Learning, Deep Learning , LSTM.

**INTRODUCTION:**

Weather forecasts have been evolving since the late 1800s, when multiple weather stations were sharing observations to develop weather prediction systems. In the 20th century, weather forecasting evolved to a whole new level with the help of satellites which helped to monitor the Earth system in an enormous amount of detail. These days, weather forecasts are so prevalent that we take notice of them only when the prediction is wrong. The good news is, weather forecasts can always keep getting better and better as more efficient algorithms of AI are implemented.

By using NOA’s enhanced collection of weather dataset, this project extrapolates various statistical data and analysis that shows patterns among parameters. In this work, we present a case study on predicting the weather conditions and show that our machine learning and deep learning models of Random Forest Regression (RFR) and Random Neural Network (RNN) can provide reliable weather forecasts when compared with NASA Power Weather predictive models.

The major contribution of this paper includes:

1. Analysis of NOAA weather data and examination of common observed hypothesis of weather conditions.
2. Use of several machine learning + Deep learning models in weather forecast like RFR and LSTM
3. Thorough evaluation of the proposed technique and comparison of used machine learning models with NASA Power Model in the prediction of future weather conditions.
4. Extending the scope and variance of predicting weather conditions by introducing Decagon model to not only predict common measures like temperature but ten other conditions such as humidity, precipitation etc.
5. The utilization of machine learning in prediction of weather conditions which can run on less resource-intensive machines.

**RELATED WORK:**

The use of machine learning in weather forecasting is a relatively new development in research. This subject is covered in several works.

Holmstrom et al. suggested a technique to forecast the maximum and the minimum temperature of the next seven days, provided the data of the past two days [1]. They used both a linear regression model and a functional linear regression model variance. They found that competent weather forecasting providers outperformed all models for up to seven-day forecasting. They showed that both the models were outperformed by professional weather forecasting services for the prediction of up to seven days. In the study done by A H M Jakerria , Md Musharraf Hossain ,Mohammad Ashiqur Rahman , regression techniques were used for weather forecasting as expected outcomes are to be constant numeric values, in this case of temperature. It was discovered that Random Forest Regression (RFR) is the superior regressor as its ensembles multiple decision trees while making decisions. The trained model predicts next day’s hourly temperature while inputting previous days atmospheric information as test data. n. In addition, comparison of several other state-of-the-art ML techniques with the RFR technique were demonstrated. The incorporated regression techniques are Ridge Regression (Ridge), Support Vector (SVR), Multi-layer Perceptron (MLPR), and Extra-Tree Regression (ETR). [2] In this research work, a proposed model for a weather forecasting system is implemented using recurrent neural networks with LSTM technique. In this model the data is trained using the LSTM algorithm. From experimental result, it is observed that Long-Short Term Memory neural network gives substantial results with high accuracy among the other weather forecasting techniques.[3] The work by Mohamed Akram Zaytar , Chaker El Amrani presented a deep neural network architecture for weather prediction. The study uses multi stacked LSTMs to forecast 24- and 72-hours’ worth of weather data (for Temperature, Humidity and Wind Speed). Approximately 15 years (2000-2015) of hourly meteorological data was used to train the model. The results showed that LSTM based neural networks are competitive with the traditional methods and can be considered a better alternative to forecast general weather conditions.[4]

**DATASET**:

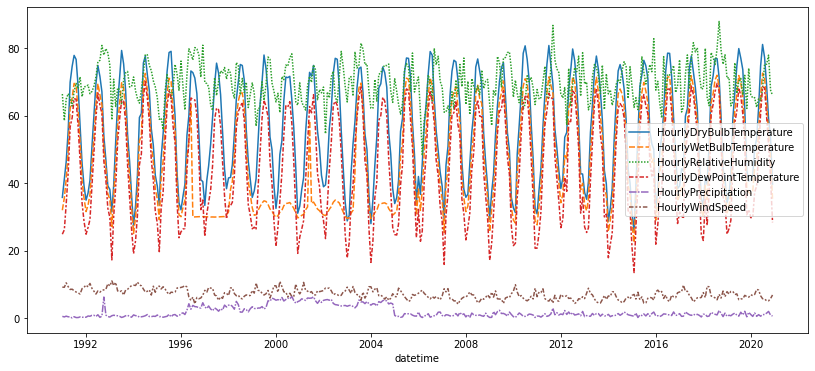
Dataset from Weather Station: The real weather data was collected for the Baltimore/Washington International Thurgood Marshall Airport (KBWI) location, MD from NOAA Data Tools for Local Climatological Data (LCD) at<https://www.ncdc.noaa.gov/cdo-web/datatools/lcd>. Type of weather data records provided was Surface Aviation Weather Observations (SAO). SAO are a compilation of elements of the current weather at individual ground stations across the United States. It was very important to review the type of weather data collection, since some of the resources available provided records from model predictions, rather than actual recordings from physical devices. For this, NOAA tools provided great flexibility with date choices together with available set of features.

SAO definition:<https://www.faa.gov/regulations_policies/handbooks_manuals/aviation/phak/media/15_phak_ch13.pdf>

**DATA SUMMARY & VISAULIZATION:**

Features considered: Altimeter Setting, Dew Point Temperature, Dry Bulb Temperature, Precipitation, Relative Humidity, Sea Level Pressure, Station Pressure, Visibility, Wet Bulb Temperature, Wind Direction, Wind Speed

1. **Distribution of Major Features Across the Years**



The above graph shows how few features maintained some sort of congruency throughout the years. Dry Bulb Temperature, Wet Bulb Temperature, Dew Point Temperature and Relative Humidity follow a fluctuating pattern while Precipitation and Windspeed shows a consistent pattern throughout the years.

1. **Sea Level Pressure vs Station Pressure**

|  |  |
| --- | --- |
|  | Above Graph shows how Sea Level Pressure and Station Level Pressure are linear and increase throughout the year in a similar pattern. Station pressure is the pressure measured at a specific altitude, whereas Sea level pressure is the pressure measured at mean sea level. The diagonal running at an approximate 45-degree angle across Sea Level Pressure and Station Pressures suggests that Atmospheric Pressure at a certain altitude as well as at mean sea level did not show much variation in case of KBWI airport. |

**Hourly | Monthly | Yearly Mean Temperature.**

|  |  |  |
| --- | --- | --- |
| Figure 1 |  |  |
| Hourly Mean Value | Monthly Mean Value | Yearly mean value |

The above graph shows the mean value of temperature with respect to hour, month and year.  
It is observed that the median temperature is the highest around noon and the lowest around midnight. The cyclical pattern is even more evident in the temperatures grouped by month — the hottest months are June up to August and the coldest ones are December to February. The temperature seems to be increasing as we progress into further years.

1. **Distribution of Temperature**

|  |  |
| --- | --- |
| Figure 1  Represents the variation of dry bulb temperature month wise throughout the years | Figure 2  Represents a heatmap, to verify the fluid pattern of weather across our dataset and to check any odd or anomalies. |
|  |  |
| Figure 3  Distribution of different temperature types | Figure 4  Distribution of default temperature across month |
|  |  |

1. **Heatmap for all features**

|  |  |
| --- | --- |
|  | The above heatmap helps in estimating the correlation between the features. The features that are closer to value ‘1’ implies that the correlation is strong, whereas the farther the value goes from ‘1’, the lesser the correlation between the features. |

**DATA ANALYSIS:**

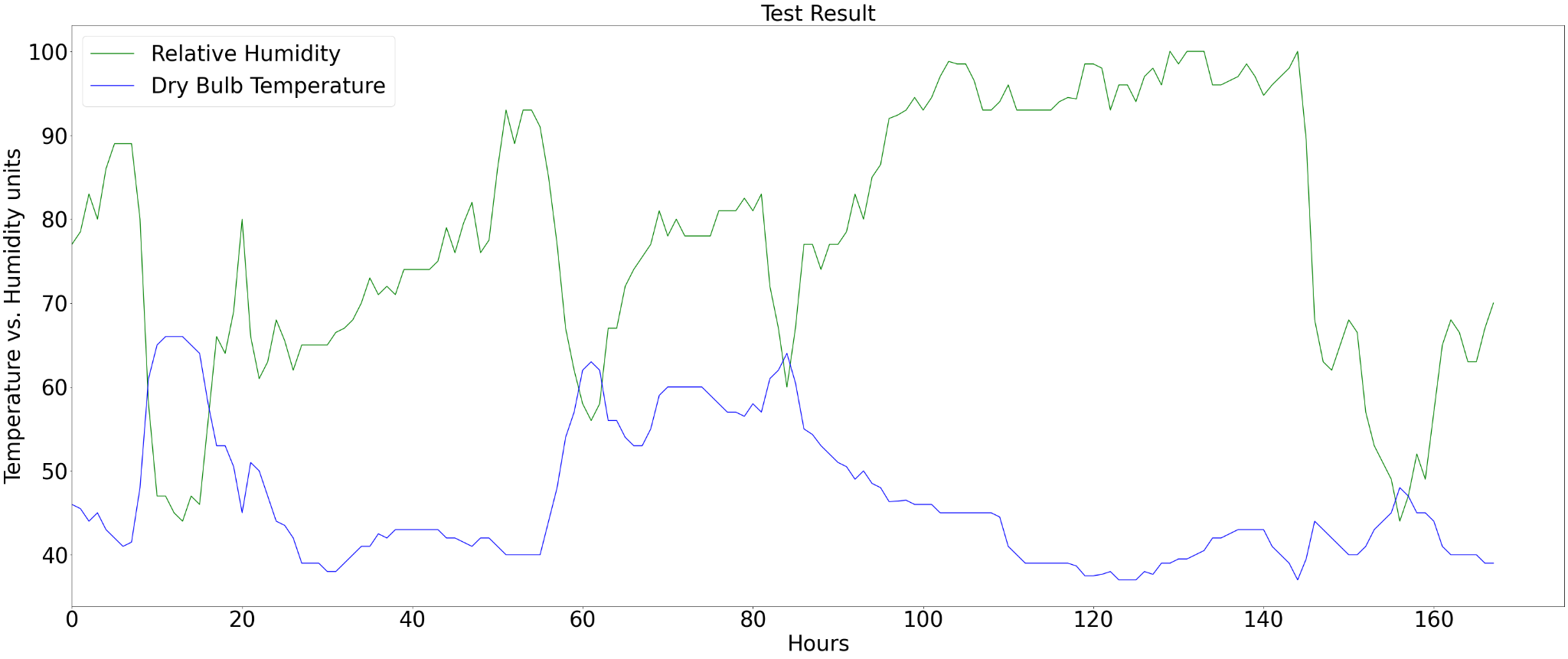
**Hypothesis:**

**1.** **Humidity versus Dry bulb temperature:**

*Hypothesis Statement:* As air temperature increases, air can hold more water molecules, and its relative humidity decreases. When the temperature drops, relative humidity increases. Therefore, the temperature is directly related to the amount of moisture the atmosphere can hold.

*Findings & Insights:* The above hypothesis holds for our dataset, proving that humidity and temperature are inversely correlated.

Fig 1: Graph of Hourly Relative Humidity and Dry Bulb Temperature

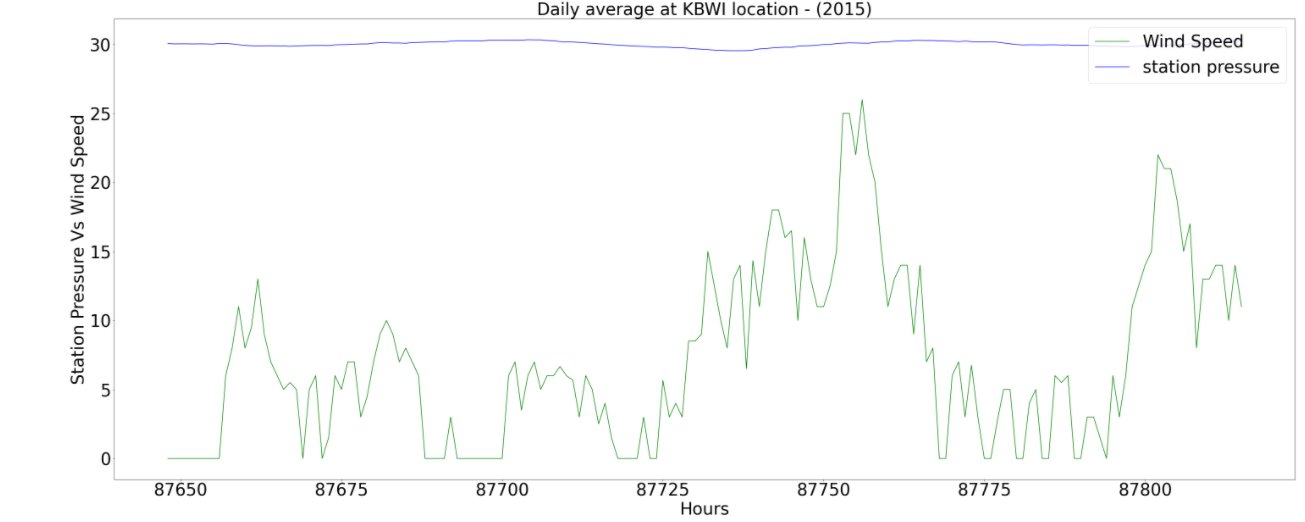


**2. Station Pressure and Wind speed:**

*Hypothesis Statement:* Station pressure is measured at a station without any adjustment. The wind is air pressure converted into movement of air. When air slows down, its pressure increases; higher wind speeds will show lower air pressure readings. Station pressure can impact the wind speed.

*Findings & Insights:* However, from the below graph, we can say that there is no correlation between wind speed and station pressure as wind speed lies in different frequencies and station pressure is uniform throughout, thus not following the ideal relationship as described.

Fig 2: Graph of wind speed and Station pressure

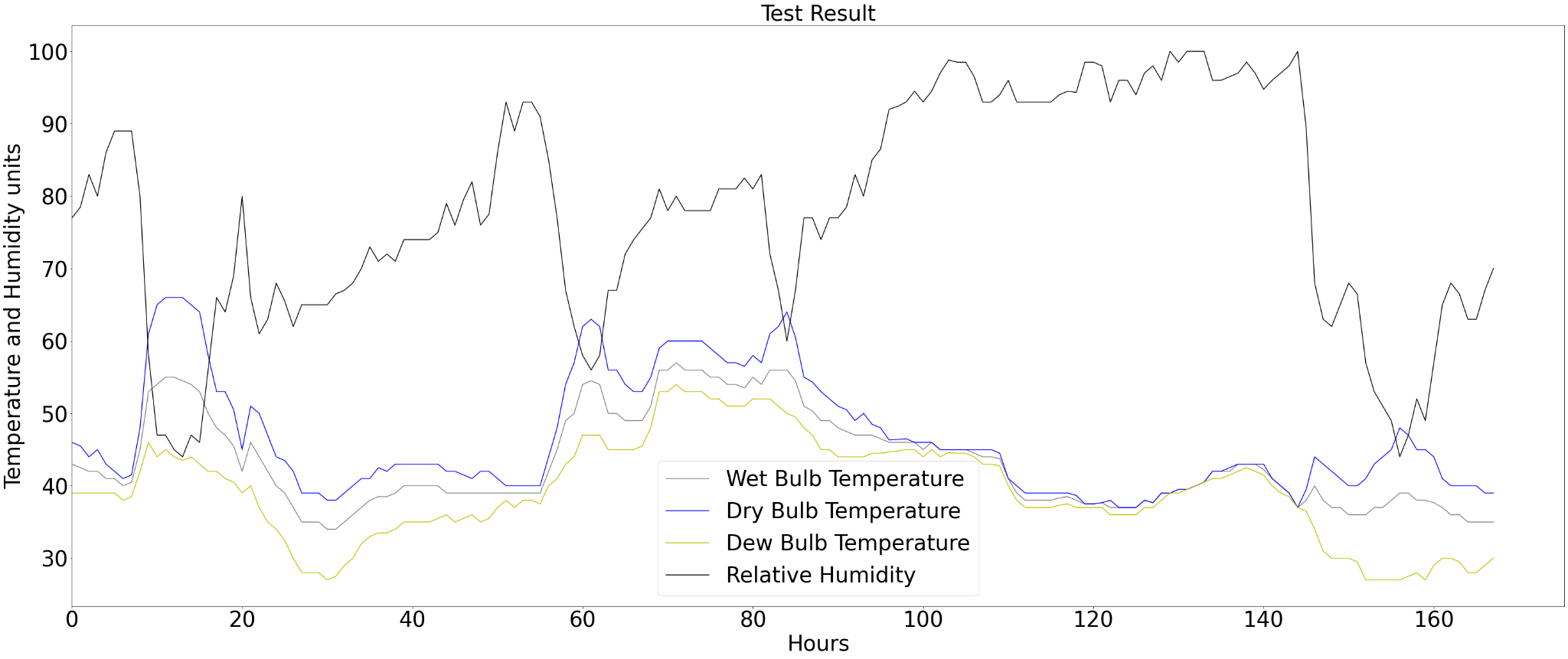


**3. Dry Bulb, Wet Bulb, and Dew point temperature:**

*Hypothesis Statement:* Temperatures are essential to determine the state of humid air. The wet-bulb temperature is always lower than the dry-bulb temperature but will be negatively correlated with relative humidity, so as the Dew|Dry|Wet temperatures.

*Findings & Insights:* Combining the dry bulb and wet bulb temperature gives the humid air. Looking at the below graph, we can say that both the dry bulb and the wet bulb are always inversely correlated to wet and dry bulb temperature and the wet bulb temp is always lower than the dry bulb.

Fig 3: Graph showing Humidity, Dry Bulb and Wet Bulb Temperatures.

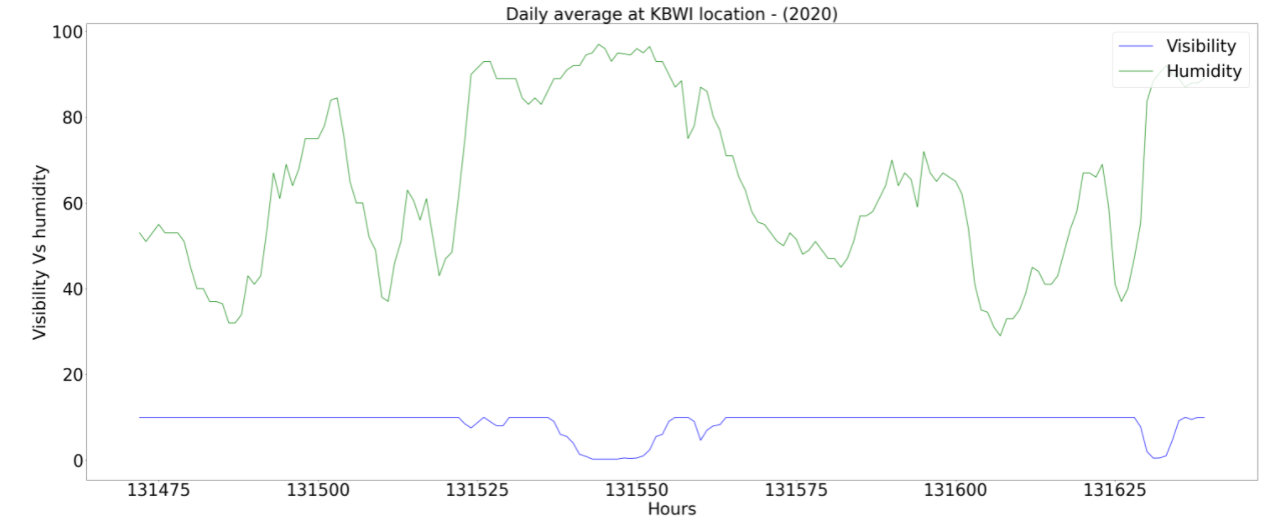


**4**. **Relative humidity and Visibility:**

*Hypothesis Statement:* Humidity decreases visibility. There is a variation in these two as humidity increases, visibility goes down, and humidity decreases visibility goes high.

*Findings & Insights:* From the below graph of sample data, it's evident that although at some instances the visibility goes down when humidity rises, it does not hold throughout the data. So, we can reject this hypothesis.

Fig 5: Graph of humidity and visibility



**5. Dew point temperature and humidity:**

*Hypothesis Statement:* If the dew-point temperature is close to the air temperature, the relative humidity is high, and if the dew point is well below the air temperature, the relative humidity is low. The higher the dew point rises, the greater the amount of moisture in the air. Higher the dew points, the humidity is increasing.

*Findings & Insights:* We can consider this hypothesis acceptable as the below data graph follows the pattern of humidity going high if the dew point and dry bulb are close to each other.

Fig 4: Graph for Relative Humidity and Dew point



**6. Average Wet Bulb and Dry Bulb Temperature Across each Months**

*Hypothesis Statement:* Evaporation being an endothermic process, the wet wick absorbs some of the heat, which is why the wet-bulb temperature is usually lower than the dry-bulb temperature.

*Findings & Insights:* We can see from the above graph how wet bulb temperature and dry bulb temperature increase and decrease congruent to each other; the wet-bulb temperature always remains consistently less to the dry-bulb temperature as the wick of wet bulb temperature is wet.

*Hypothesis Statement:* The drier the air is, the lesser is the temperature picked up by wet bulb temperature, and the greater is the difference between dry-bulb temperatures.

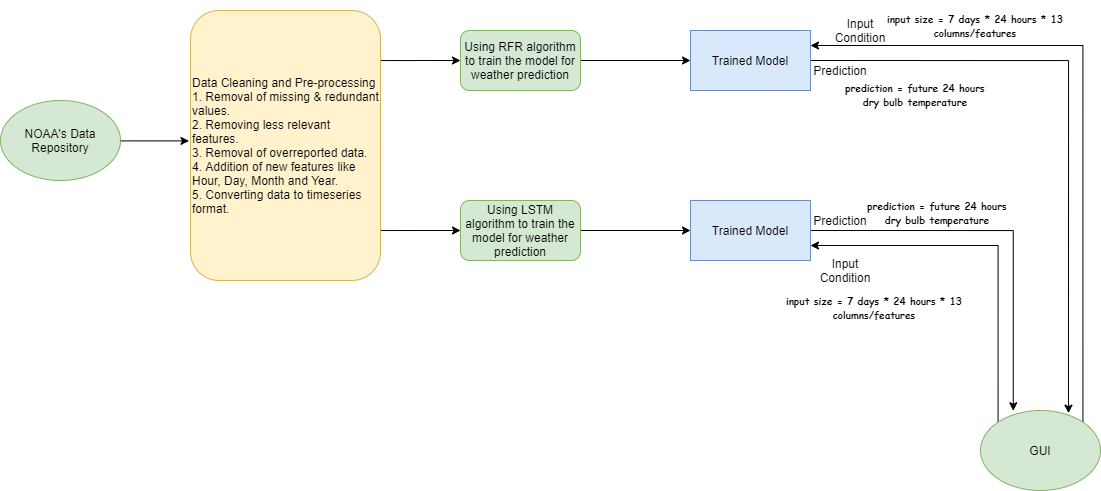
Findings & Insights: We can see the evidence of the following in the graph. During the winter, rainy months, the difference between dry-bulb temperature and wet bulb temperature is pretty low. In the summer months, the temperature difference between dry bulb temperature and wet bulb temperature is slightly higher. This is due to the air being drier, causing a more significant distinction of recording between dry bulb temperature and wet bulb temperature.

****

**MAIN HYPOTHESIS:** Machine learning model trained on common atmospheric conditions is going to perform at the level of numerical weather prediction models for predicting daily temperature, by taking into account the input of 7 days of historical data and providing a 24-hour temperature forecast.

*Commonly used atmospheric conditions and features:* Altimeter Setting, (Dew | Dry | Wet) Bulb Temperatures, Precipitation, Relative Humidity, Sea Level Pressure, Station Pressure, Visibility, Wet Bulb Temperature, Wind Direction, wind speed.

**SYSTEM ARCHITECTURE:**



The above figure illustrates the system architecture of the project. Initially, the dataset was obtained from NOAA’s data repository Various cleaning and preprocessing was carried on the dataset as it had a lot of inconsistencies and missing data. Later the grouping of data is performed on hour and date to get rid of overreported data caused by multiple readings for single hour/instance of day .The data was trained using two different models: Random Forest Regression(RFR) and Long Term Short Memory(LSTM). Once the model is trained, we can provide input like 7 days of hourly features for which we would in turn get a prediction for the next 24 hours of hourly prediction.

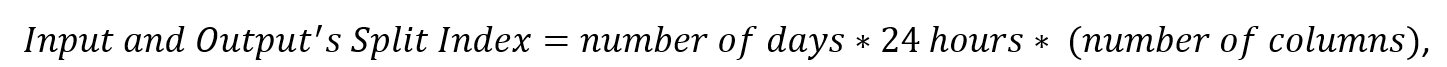
**METHODOLOGY:**

**Data Cleaning:** For a given place and predefined date range, using LCD, we received an hourly distribution of weather features on the daily basis. Initial data range spanned from 1991 till the end of 2020. Weather features include hourly altimeter setting, dew | dry | wet bulb point temperatures, precipitation, relative humidity, sea level pressure, station pressure, visibility, and wind speed. Original data also included Daily, Monthly averages, and short duration for mentioned columns above together with several backup columns, but they were discarded right away, since no evident use of them was seen. After discarding unnecessary columns, the dataset was cleaned from characters and special symbols that were met in most of the remaining columns that had to have integer or float values only. Appropriate data types were assigned to columns.

**Data Preprocessing:** The Distribution of data does not change dramatically across the years, although the years 1996-2004 had massive amounts of data which were missing hence was removed from the dataset, also to retain the continuity of the data, years before them were also not considered. As a result ,16 years of data from 2004-2020 is used for training weather forecasting models. It was also observed that for many instances of day i.e., hour, the data was overreported by the weather station. To remove the redundancy caused due to overreporting of data, the dataset was grouped and averaged by the hour & date. Few features like Wind Direction variable did not have enough data to consider hence it was eliminated. Unimportant and highly correlated features like HourlyPresentWeatherType, HourlyPressureChange, HourlyPressureTendency, HourlyWindGustSpeed were eliminated from the dataset too. Additional features like Day, Month and Hour were introduced in feature engineering phase which later proved to be important features while performing analysis and modeling.

|  |  |
| --- | --- |
|  |  |
| Nan Distribution before grouping | Nan distribution after grouping |
| **Nan Distribution before and after grouping**  Above graphs shows the Nan Distribution after and before grouping the data by Hours, Date and Year to remove the redundancy caused by overreported data. | |

**Data Preparation for Modelling:**

In order to prepare data for training, we convert it into the time series format. The input structure of this set consists of 7 days of historical data for all selected features, including Dry Bulb Temperature. The output contains Dry Bulb Temperature values for the next 24 hours. As a result, single training sample contain 8 days of weather data, where input and output is being separated by column index calculated by following formula:   
 where in our scenario the number of days equals 7 and the number of columns is 13.

**Model Descriptions:** To proceed with weather forecasting, we have found that Random Forest Regressor and LSTM can provide weather predictions and perform relatively close to the numerical weather model. To prepare models for predictions, we have kept our approach to the following format: supervised learning to predict multivariate time series data.

**Model RFR**

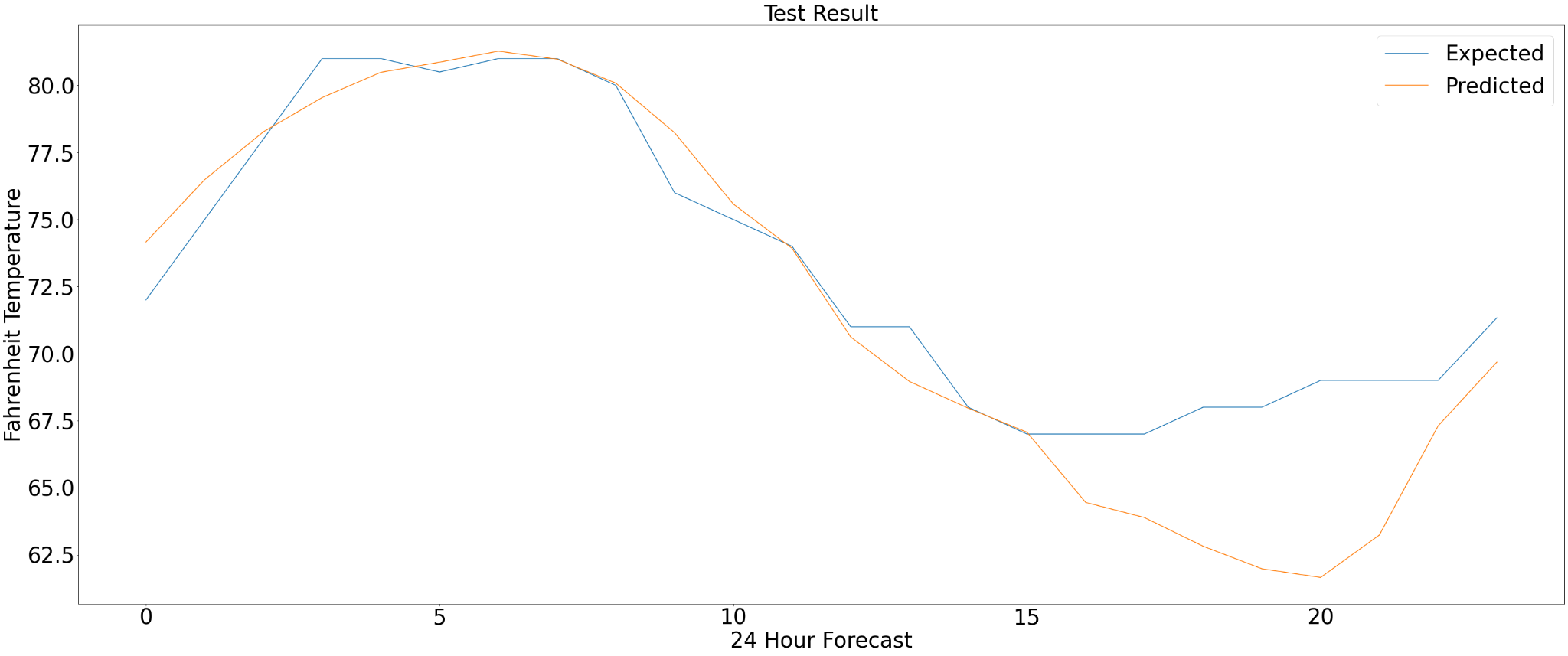
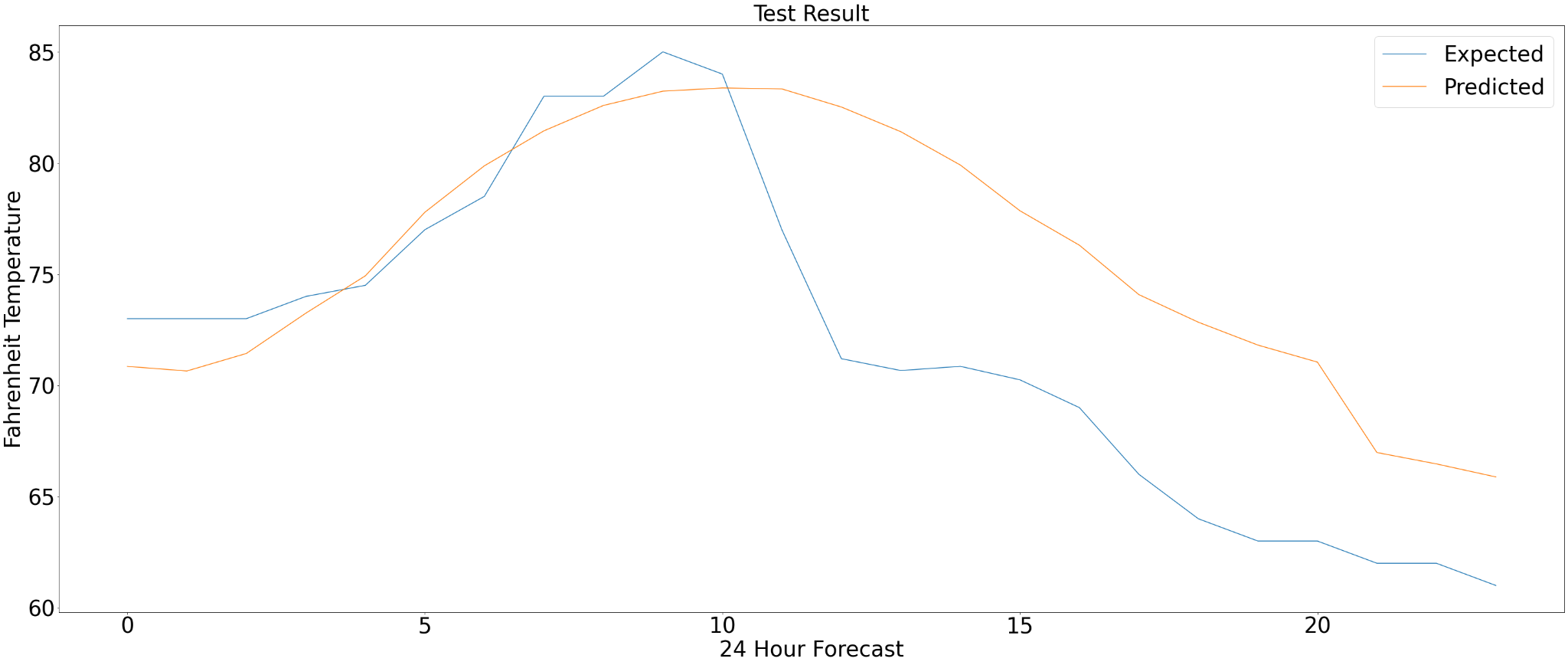
Random Forest model was initially set up with a large number of estimators and decreased down, as we saw that model with lower number of estimators still showed good performance on the RMSE and MAE metrics. The lowest number of estimators we have tried was 750 estimators. Our target variable for the initial model was Dry Bulb Temperature. In addition to selected features, we have also added hour and month features, in order to see how it would affect training performance.

**RFR RESULTS:**

|  |  |
| --- | --- |
| **RMSE** | 3.729 F |
| **MAE** | 5.04 F |

Using the past 168 hours of data and forecasting 24 hours ahead, the mean absolute error was 3.7 degrees of Fahrenheit and RMSE is 5.04.

Sample Results from Random Forest Model Prediction:

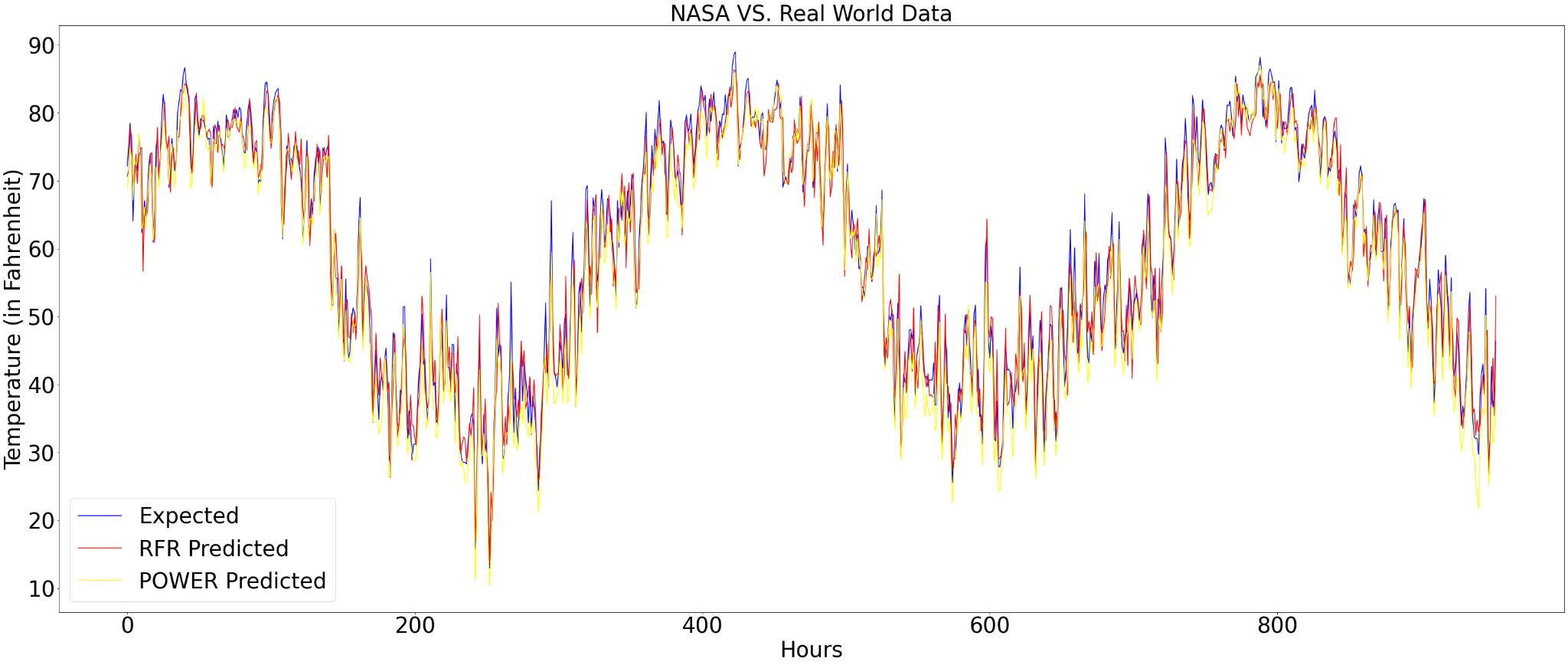
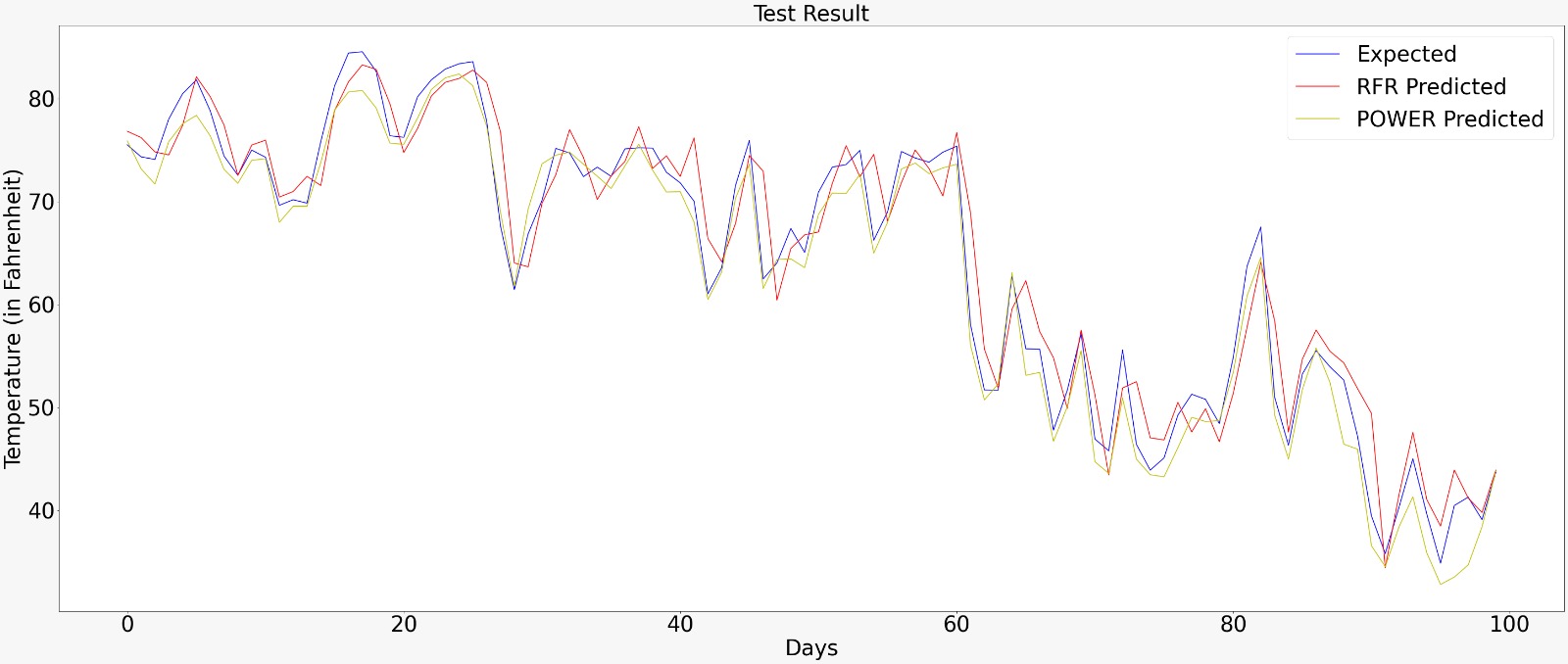


Chart, line chart

Description automatically generated

**RESULT OBTAINED BY COMPARISON OF RFR with POWER PREDICTION MODEL :**

The 24-hour forecast from the model showed accurate predictions that lined up closely with the expected temperature. To evaluate our results with the numerical model from the POWER dataset, we have averaged our daily predictions to keep it in the same format as NASA’s model. Performance of our model showed similar performance to numerical weather model and to verify that we also used RMSE and MAE, where our model showed slight underperformance of around 0.72 and 0.409 difference respectively to NASA’s numerical weather model.

**Model LSTM**

LSTM: Long Short-Term Memory (LSTM) is a type of recurrent neural network that can learn the order dependence between items in a sequence.

The aim is to forecast the future using past data. The data is in the form of a time series or a sequence. We have used a TensorFlow implementation of a recurrent neural network with an LSTM layer for sequence modeling.

The input to an LSTM network is a 3D array: [ samples, timesteps, features ]

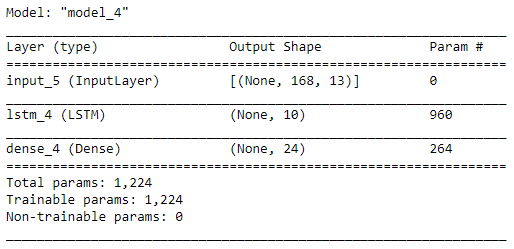
samples — total number of sequences constructed for training.

timesteps — the length of the samples.

features — number of features used.

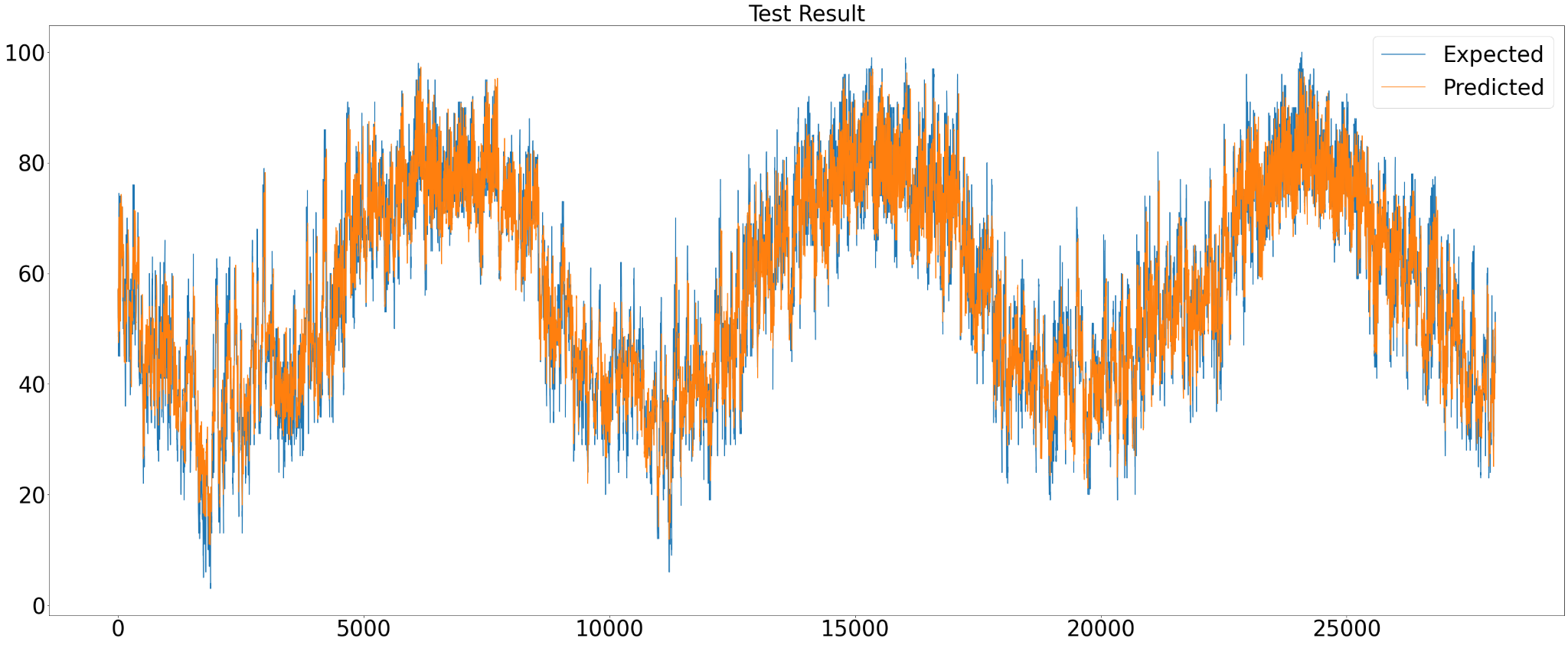
Scaling the data is the final step before generating all of the matrices for the modeling.

In other words, we perform mean scaling x ← (x−µ)/ σ to all the continuous variables so that the variables possess approximately zero mean, which in practice, reduces computational cost while training the models.



With the trained model we can forecast and compare the values with the original ones.

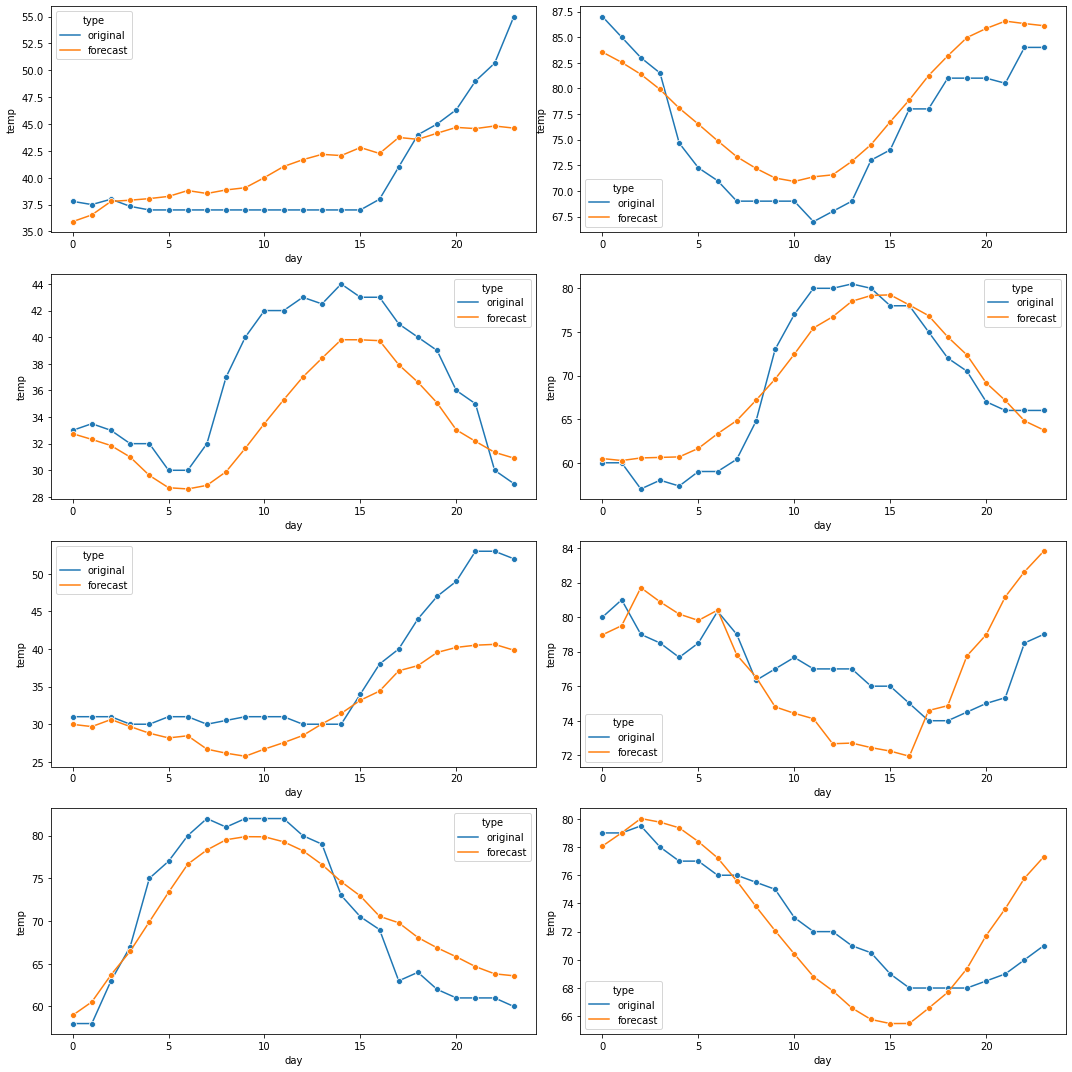
**RESULTS:**



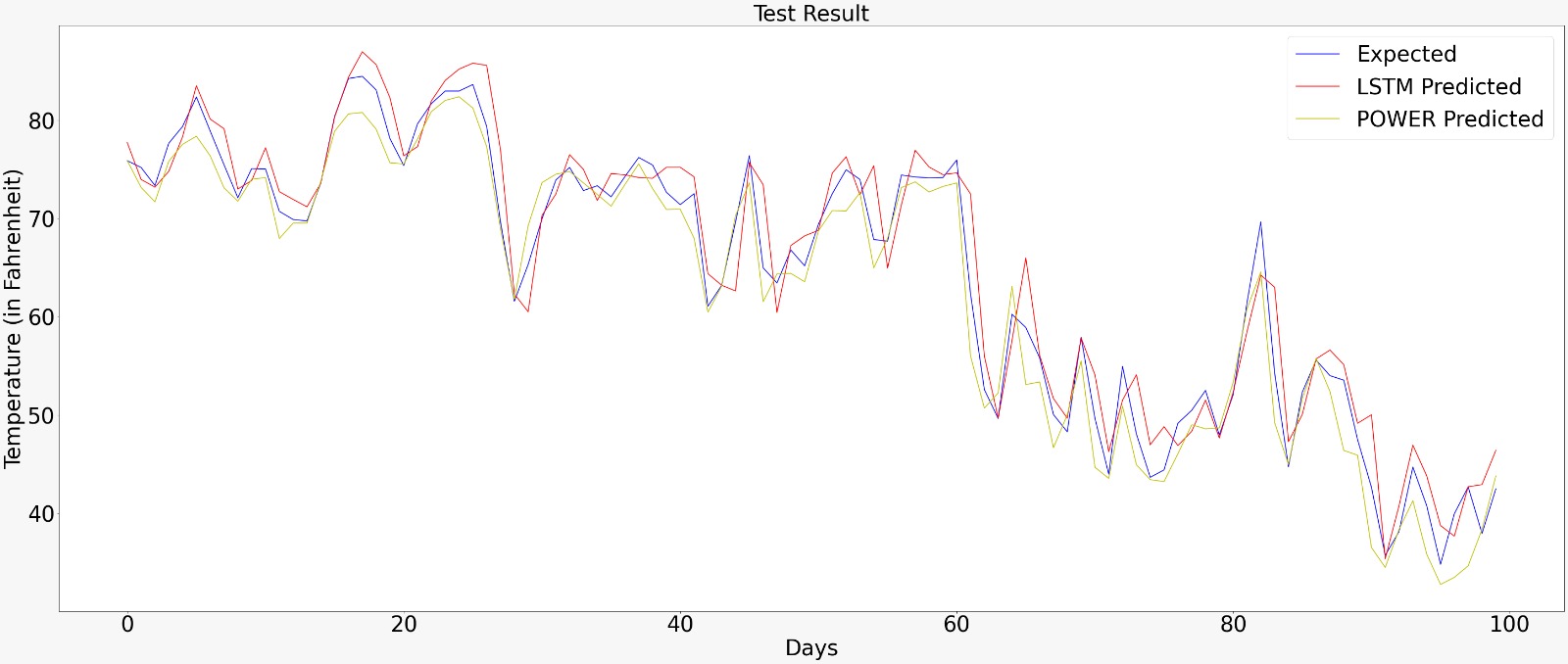
The orange line which demonstrates the predicted values while the blue ones which are barely visible are of expected ones. These two lines are very close to each other, they basically overlap.

|  |  |
| --- | --- |
|  | Using the past 168 hours of data and forecasting 24 hours ahead, the mean absolute error is 3.08 degrees of Fahrenheit and RMSE with 4.12 F. |

Inspecting some random chunks of 24 hours predicted data



**RESULT OBTAINED BY COMPARISON OF LSTM with POWER Prediction Model:**



**Decagon Models Framework**

After observing that some features such as Dry Bulb Temperature and Relative Humidity showed high correlation, we have decided to see if we could provide predictions for all available features. LSTM was used to proceed with such forecast, and the output was converted from single 24-hour output into 10 features, where we have kept only atmospheric conditions and excluded Month, Day, and Hour features from the output label. After several training phases, the model showed poor results in predicting 10 features.

Next approach that we came up with was Decagon Framework. This framework provides us with capability to approach each feature prediction individually and train each model with the focus on predicting a single feature. From current development, we were able to train models individually for each feature, and 4 out of 10 features showed good performance. Once the proper hyperparameter for LSTM model is found, these hyperparameters can be used in

Decagon Framework, which would automatically feed in pre-selected hyperparameters per each model and proceed with the training. At the end of training process, Decagon Framework is going to return dictionary where key is associated with feature and its value is the location of saved model. In addition to training framework, we also provide evaluation framework that provides us with functionality to input data once for desired forecast and simultaneously receive the predictions for all features, keeping in mind that appropriate hyperparameters were used for each model training in the Decagon Learning Framework.

Our approach for providing this framework allows everyone to come up with single feature prediction’s hyperparameters and share them with the current framework. As a result, when each of the features have their appropriate model hyperparameters, our decagon framework can provide users with 10 features predictions on the 24-hour range.

**FINAL COMPARISON RESULTS:**

**Comparison between predictions, observations and NASA’s Power data predictions**

****

The above graph illustrates the performance of our models compared to NASA’s Power dataset temperature predictions and expected temperature.

**Model Metrics:**

|  |  |  |
| --- | --- | --- |
| **Hourly Table** | RFR | LSTM |
| RMSE | 5.04 (F) | 4.12 (F) |
| MAE | 3.73 (F) | 3.08 (F) |

Following table shows the root mean square error and mean absolute error of RFR and LSTM when the predictions are evaluated hourly

|  |  |  |  |
| --- | --- | --- | --- |
| **Daily Average** | POWER | RFR | LSTM |
| RMSE | 3.02 (F) | 3.75 (F) | 3.68 (F) |
| MAE | 2.48 (F) | 2.89 (F) | 2.80 (F) |

Following table shows the root mean square error, mean absolute error of RFR, LSTM as well as NASA’s Power Dataset predictions when the predictions are averaged daily. In all cases, LSTM performs slightly better than RFR.

**CONCLUSION AND FUTURE WORK:**

In this paper, Machine Learning models were used to provide temperature forecasts, with the focus on 7 days to 1 day prediction. Using the dataset received from NOAA that consisted only of Land Based Weather Station, we have investigated on generating model to predict weather forecast at the level of numerical model that we have observed from POWER Dataset, developed by NASA. Our Random Forest and LSTM models were able to perform close to numerical weather. Compared to numerical weather model’s requirement for aircrafts and satellites to collect data for its input, our models used historical data used only from weather stations that are available across the globe. In addition to the simplified weather requirement, our models were not only able to perform closely to NASA’s numerical weather model on the daily temperature average forecast, but also predict expected temperature for each hour individually. Furthermore, we decided to experiment with generating predictions for all the features used in the dataset and provide multivariate prediction per feature. Framework flow resembles stacking ensemble, but instead of evaluations on single target, each model provides its own prediction for associated feature. Decagon Training Framework is used to train individual LSTM model for each feature and pass this into Decagon Forecasting Framework that would use trained models to provide predictions for each feature for the next hours. With further development of weather forecasts, Decagon Framework can be used as the ground base for multivariate predictions. With future work being done and models being computed with accurate performance, this framework could be used as an interpolation technique for weather datasets and providing multivariate weather forecasts.

**REFERENCES:**

1. Mark Holmstrom, Dylan Liu, and Christopher Vo. 2016. Machine Learning Applied to Weather Forecasting

2. A H M Jakaria , Md Mosharaf Hossain ,Mohammad Ashiqur Rahman

Smart Weather Forecasting Using Machine Learning: A Case Study in Tennessee

3. Dires Negash Fente; Dheeraj Kumar Singh   
Weather Forecasting Using Artificial Neural Network

4. Mohamed Akram Zaytar , Chaker El Amrani

Sequence to Sequence Weather Forecasting with Long Short-Term Memory Recurrent Neural Networks

5. Lochandaka Ranathunga ,Eranga De Saa

Comparison between ARIMA and Deep Learning Models for Temperature Forecasting

6. S. Hochreiter and J. Schmidhuber, "LONG SHORT-TERM MEMORY," in Neural Computation, 1997.

7. Krasnopolsky, Vladimir M., and Michael S. FoxRabinovitz.   
‘’Complex hybrid models combining deterministic and machine learning components for numerical climate modeling and weather prediction.”Neural Networks19.2 (2006)

8. Nebeker, F, "Calculating the weather: Meteorology in the 20th Century", Academic Press, New York, 1995.

9. L. LAI, A. BRAUN, Q. P. ZHANG, Q. WU, Y. N. MA, W. C. SUN and L. YANG, "INTELLIGENT WEATHER FORECAST," in Proceedings of the Third International Conference on Machine Learning and Cybernetics, Shanghai

10. National Weather Service <https://www.weather.gov/arx/why_dewpoint_vs_humidity#:~:text=The%20dew%20point%20is%20the,water%20in%20the%20gas%20form>

11. National Weather Service <https://www.weather.gov/source/zhu/ZHU_Training_Page/definitions/dry_wet_bulb_definition/dry_wet_bulb.html#:~:text=The%20rate%20of%20evaporation%20from,the%20humidity%20of%20the%20air.&text=The%20wet%20bulb%20temperature%20is%20always%20lower%20than%20the%20dry,is%20at%20the%20saturation%20line>