**CSCE 823 Advanced Topics in Statistical Machine Learning Project Proposal**

**Predicting Weather Conditions Utilizing Artificial Neural Networks for C-17 Mission Planning**

**1st Lt Garrett A Alarcon**

**Abstract**

Reducing cost within the military is always an issue facing senior leaders. A big one for the Air Force is reducing the cost from fuel consumption, especially among C-17s which are the workhorse for air mobility. This project looked at the first step in reducing fuel consumption which is accurate weather predictions. If temperatures and wind speeds are known ahead of time for the region of travel, mission planners can develop the most fuel-efficient route for operators to fly thus saving money in the process. To accomplish this, data from NOAA was used to build a neural network capable of predicting temperature and wind speeds 12 hours into the future. This model was trained on a specific latitude/longitude coordinate at a pressure layer of 2 kPa, which is roughly 26 km. After training and validating, the model was tested against a variety of coordinates among varying altitudes and lat/longs. The results were extremely impressive with the model showing an incredible ability to generalize and predict weather factors in the future.

**Introduction**

The Department of Defense (DoD) has an obligation to the American people to be stewards of their tax dollars in all defense related spending. As such, the DoD is always searching for ways to minimize spending while also increasing combat capabilities, military readiness, and operational effectiveness. Aircraft are an integral part of the United States Air Force (USAF) and by its very nature, fulfilling these goals incurs a substantial cost for procuring and consuming fuel. From the 2019 fiscal year budget for the DoD, $24 billion was requested for fuel consumption with $6.6 billion going to operations and $4.5 billion going to transportation [1]. The USAF consumes around half of this budget for aviation fuel with majority of it being used by Air Mobility Command (AMC), a major command (MAJCOM) within the USAF structure [2]. Of the many aircraft within the AMC inventory, the C-17 fleet represents the primary aircraft responsible for global transportation and cargo, and as such consumes the largest amount of the aviation fuel [3]. Increasing efficiency in fuel consumption within this fleet can have an immense impact on cost savings for the UASF, and DoD as a whole.

There are a multitude of ways to address increasing efficiency in fuel consumption among C-17s. This project focuses primarily on the first step in this problem, which is predicting temperature and wind speeds within the region of travel for C-17 operations. Having accurate predictions for the varying spatial regions of the atmosphere will enable mission planners to plot the most fuel-efficient route from origin to destination. Currently operations within AMC still rely on a deterministic weather forecast built from instrumentation readings and various weather models. These forecasts have been shown to be not the most accurate, so any improvement upon this will be substantial for mission planners.

The data for this project comes from the National Oceanic and Atmospheric Administration’s (NOAA) database for deterministic weather models. The data obtained includes observations for every latitude and longitude point at 31 different pressure layers of the atmosphere, which was taken every six hours. The data used runs from 20 July 2018 to 18 December 2018, containing 603 readings. It is important to note that the forecasts were not used, only the actual weather model results and instrumentation readings for each time interval.

As the nature of the data is time series, a long-short term memory (LSTM) recurrent neural network (RNN) was chosen to model future time steps of the weather. These networks have been shown to be incredibly effective at predicting values from a time series sequence, by “remembering” and storing different pieces and patterns within the sequence [4]. As a supervised learning task, this problem involves numeric inputs along with numeric outputs some time step in the future. Therefore, a regression approach was taken with the network since the outputs need to be real value numbers.

The results from this project will support the larger research being conducted on weather prediction. In particular this project will help determine the feasibility of building a prediction trained on one latitude/longitude coordinate at a specified pressure layer, and whether that can be generalized for all coordinates in that pressure layer across the planet, and in different pressure layers. This is the first step in building a larger, more robust model, and computationally faster model that mission planners can rely on.

The following sections will focus on the previous works done in this area, along with details of the methodology employed in this project. The results of this work will then be explored along with any concluding remarks.

**Related Work**

Many different works have explored how to accomplish weather prediction, and the different types of techniques and network architectures to employ. When doing regression of a sequence of data, the first thing to think of might be function approximation. This is a popular thought as this is what a neural network is hoping to accomplish. To that end the activation function of the network has been shown to have an important role. While not used in this project, there are many papers that detail the effectiveness of using radial basis function (RBF) networks. These are one-layer networks where the activation function is a gaussian distribution, and the nodes represent cluster centroids or observations within the dataset. These layered with ensemble techniques have shown to provide incredibly low RMSE scores along with fast training times. When compared against basic neural networks, they been shown to outperform them relatively easily [5]. This has also been shown to work for single location weather forecast for up to 24 hours out at a constant altitude. An RBF network was used for its speed of learning and ability to approximate functions to develop an accurate model [6].

Another method focused on building a multilayered network but using raw data that wasn’t normalized. This method focused on the applicability of multilayered deep networks that were not RNNs, and if raw data would be an issue. It was found that deeper networks work better for weather forecasting, as well as the tanh activation function [7].

The next step from all of this is to implement LSTM into the network and see how the model fares for predictive ability. One study used multilayered LSTMs to map future weather sequences for certain cities 24 and 72 hours out. It was trained on 15 years’ worth of data and showed excellent results compared with the leading statistical models [8]. This showcased the effectiveness of not only neural networks in the field of weather, but LSTMs for time series forecasting.

**Methodology**

**Data**

The data was obtained from NOAA’s numeric weather prediction models. These models start with initial states of weather at the given coordinates, then build forecasts from those readings and model outputs. The data set used for this project only examined the initial weather state readings and did not utilize any forecasted data. The data required lots of preprocessing before being usable, as majority of weather data is saved in file formats more compatible with LINUX machines. The file type used here was Gridded Binary (GRIB2), which is a commonly used data format in meteorology for storing historical and forecast weather data. Preprocessing and restructuring the data consumed majority of the time in this project and was easily the most difficult part. The GRIB2 files cannot be read easily on a Windows machine without applications to convert into a NetCDF first. Luckily MATLAB has a lesser known tool to do this. Once converted these files contained 117 variables, each varying from 1-D to 5-D arrays of data. The variables are interest were extracted from this data object each as a 4-D array and reprocessed into a single 2-D array containing the three features of interest shown in Table 1. This was repeated for each file, or time step, until a full data frame was constructed. This data frame contained the three features for every time interval at one latitude/longitude coordinate, and at one pressure layer. Table 2 shows an example of the data frame structure.

|  |  |  |
| --- | --- | --- |
| Feature | Units | Description |
| Temperature | Kelvin (K) | Temperature at measured coordinates |
| Wind U-Component | Meters per Second (m/s) | Zonal velocity, the component of wind towards the east |
| Wind V-Component | m/s | Meridional velocity, the component of wind towards the north |

Table 1: Table of features used in model

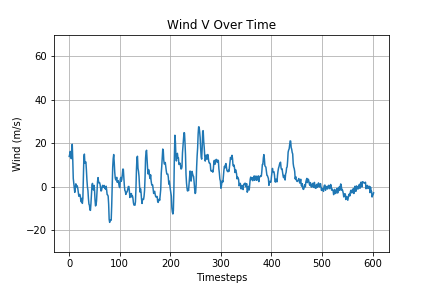
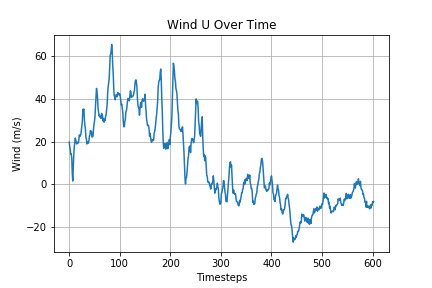
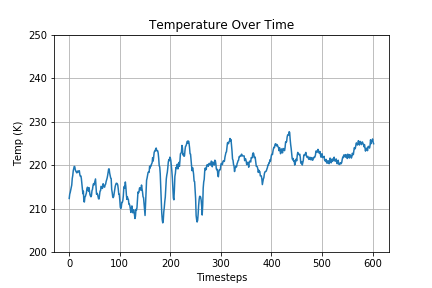
|  |
| --- |
| ***Latitude: 40***  ***Longitude: 75***  ***Pressure: 2 kPa***  ***Average Altitude: 26.254 km*** |

|  |  |  |  |
| --- | --- | --- | --- |
| Time Step | Temperature | Wind U-Component | Wind V-Component |
| 0 | 212.317 | 19.762 | 13.938 |
| 1 | 213.300 | 17.760 | 14.977 |
| 2 | 213.700 | 16.464 | 16.253 |
| 3 | 214.293 | 14.064 | 15.334 |

Table 2: Data frame example for given coordinates showing first four entries

Each coordinate had 603 observations, corresponding with the 603 timesteps within the time range of the data. Since this is a regression problem and the project aimed to predict two timesteps into the future, the truth label for the observations was derived from each feature’s value two timesteps in the future. This is constructed later when building the training and test sets for the model.

Data was explored for a multitude of coordinates, but the one chosen for the training set is the one shown in Table 2. This is because weather in the upper atmosphere tends to relatively calmer, but experiences many different patterns over time, which can be beneficial in training a model to generalize for other coordinates. Figure 1 displays charts of the three features over time to showcase the property discussed above.



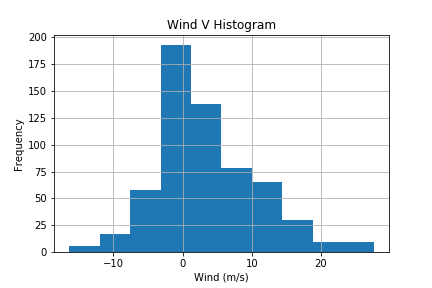
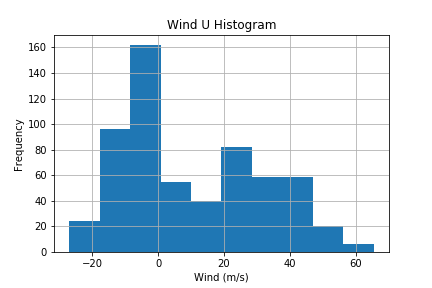
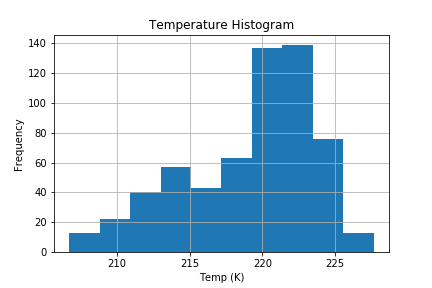
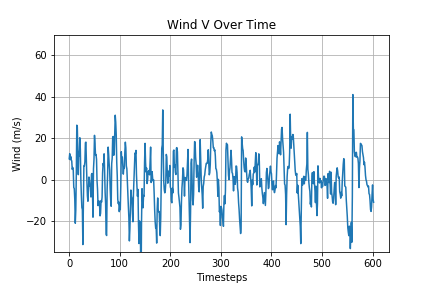
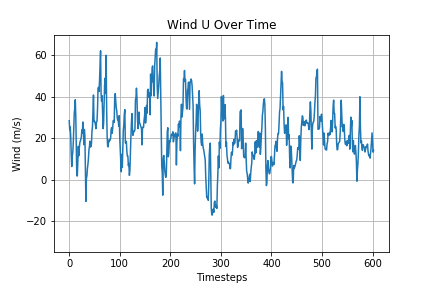
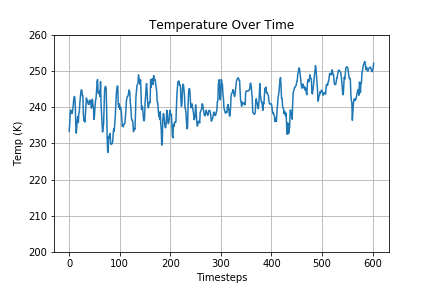


Figure 1: Charts showcasing time series plots and histograms for the features at coordinate 40 N, 75 E, at an average altitude of 26.254 km

The histograms do not display an overly normal distribution for this coordinate. The wind components are the closest but display a level of skewness. While the temperature almost appears to be bimodal. The time series charts show something more interesting though. Looking at the temperature and the U wind component, there are a variety of patterns present. Chaotic up and downs, along with smooth up and down movements appear in all three features, giving credence to the variety of patterns that could be learned by the model. Compare these with the charts shown in Figure 2. These charts show the same thing for the same coordinate except at a lower altitude. While the second coordinate may show a more normal pattern in its range of values, observing the time series charts shows a bit more chaotic nature to the pattern. This can indicate a coordinate that if learned, might not be able to generalize as well to other coordinates. It also highlights the nature of weather in the different levels of the atmosphere.



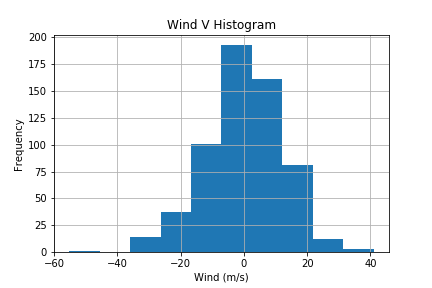
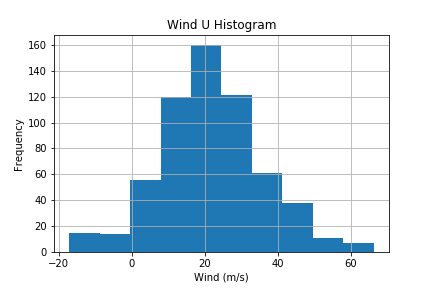
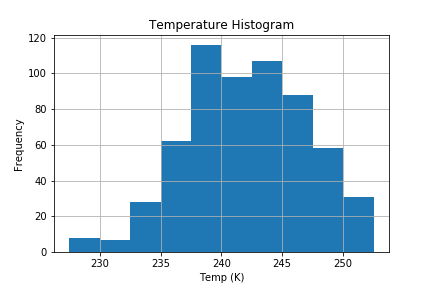


Figure 2: Charts showcasing time series plots and histograms for the features at coordinate 40 N, 75 E, at an average altitude of 7.136 km

The final thing done to the data was to normalize it as neural networks generally perform better for regression with normalized and scaled data.

**Model Architecture**

After preprocessing and examining the data, the next step is to build the architecture for the RNN. To perform the machine learning task of supervised regression, an LSTM architecture was chosen. This is because of the time series nature of the data, and to support the goal of predicting future values. Two models were examined for comparison in deciding which to use for final testing purposes. The first being a small model, and the second being a larger model. The goal here was to see if a small model could accomplish the task, and if the large one was better, would it be significantly better?

The small model is a one-layer LSTM network without any sort of regularization features. It has a small layer width of only 8 nodes, with a hyperbolic tangent (tanh) activation function. With an input shape of (2,3), that gives 411 trainable parameters for the model. The input shape’s dimensions represent the number of time steps, and the number of features respectively. The output shape then is only (3) since we are predicting the future values of the three features. This represents a many to many style input/output relationship. Also, a linear activation function is used in the output in order to provide continuous values for regression. Figure 3 shows the diagram of the small’s structure.

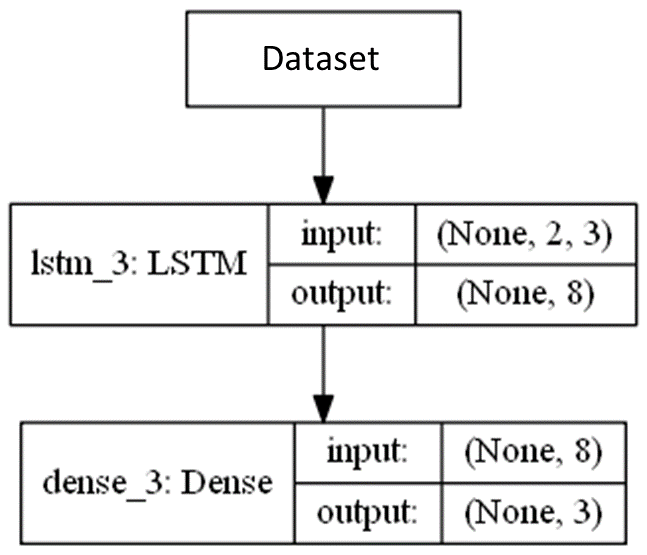


Figure 3: Architecture for small LSTM network

The large model still kept one LSTM layer but increased the layer width to 64 nodes. Regularization elements were considered to be added, but testing different architectures showed that batch normalization and dropout made the performance worse. This new network had 17.603 trainable parameters, a significant amount more than the small network. The input and output shapes remained the same, as did the activation functions. Figure 4 shows the architecture for this network.

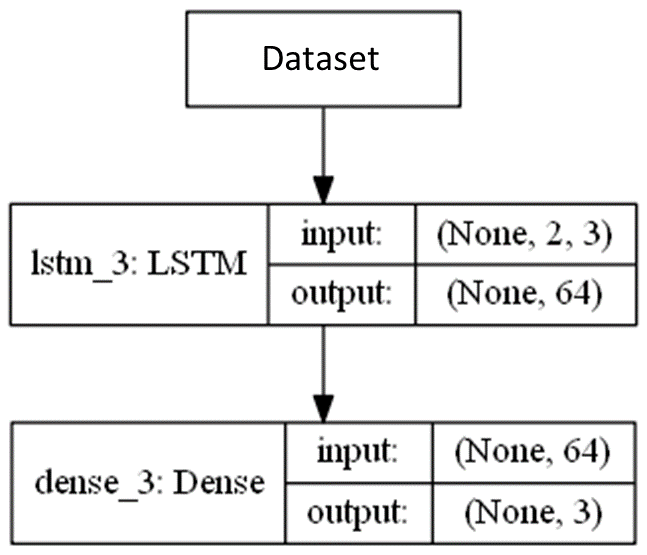


Figure 4: Architecture for large LSTM network

**Model Fitting**

After completing the architectures, the data needed to be partitioned into training, validation, and test sets. Since the data is time series, order needs to be preserved among the observations. Also, since one coordinate only has 603 data points, this makes it a very small sample to split into three sets and still get results that can be generalized. A way around this is to use the full set of observations for one coordinate to train the model. This way it can learn all the patterns represented within the entire series. Since validation is supposed to ensure overfitting is not happening, it’s the same as asking if the model can generalize. If the model has overfit, then generalization to other points will become extremely difficult. Therefore, the entire series for a different coordinate can be used for the validation set. Thus, leaving the test set to be another entire sequence from a different coordinate. Since there are sets for every latitude/longitude, and 31 different pressure layers, this yields 2,008,800 different coordinates to choose and train/test from. This also ensures no overlap exist with the training, validation, and test sets, and truly test and validates the training model’s ability to generalize and not overfit nor underfit. Figure 5 shows the breakdown of the three sets along with which coordinates were used in the process.

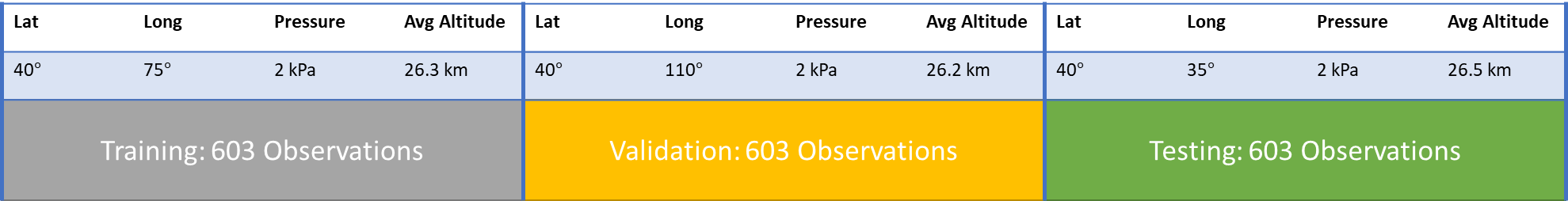


Figure 5: Breakdown of training, validation, and testing sets

It was mentioned earlier that the labels still needed to be added to the data. Since two-time steps is being chosen, this is as simple as making a copy of the training set, removing the first two entries, and shifting everything back by two. This is now the y training set to complement the x training set. Table 3 shows an example of these together to offer evidence that the x observations are being trained to the proper labels two-time steps in the future.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Time Step | Temperature | Wind U | Wind V | Temp T+2 | Wind U T+2 | Wind V T+2 |
| 0 | 212.317 | 19.762 | 13.938 | 213.700 | 16.464 | 16.253 |
| 1 | 213.300 | 17.760 | 14.977 | 214.293 | 14.064 | 15.334 |
| 2 | 213.700 | 16.464 | 16.253 | 214.897 | 14.190 | 12.910 |
| 3 | 214.293 | 14.064 | 15.334 | 215.396 | 12.266 | 16.791 |

Table 3: example training set showing the future values for the features two-time steps forward

The validation set will be used to ensure the model doesn’t overfit with the given parameters. Given the limitation of time to accomplish this project, hyperparameter tuning wasn’t explored in this iteration. That is something that will be looked at in continuing future work.

**Model Evaluation**

To evaluate performance between the two models, and eventually performance on the test set, mean squared error (MSE) will be used. This will also be used as the loss function on the model. With this as the loss function, then model checkpointing can be implemented allowing the program to save the best model with the lowest MSE on the validation set. This is to ensure we do not take the final model at the end of the epochs which has the potential to be overfitted. The accuracy of the prediction will also be shown graphically with an overlay of the predicted series over the original series.

Residual analysis will also be looked at but not necessarily reported. The data is known to be nonlinear, therefore searching the residuals for a uniform distributed appearance would be in error since it should be expected to not be. The MSE holds more value for the error than the residuals for a nonlinear function.

**Results**

This section will detail the results of the project. It will investigate and analyze the performance metrics for the two models. MSE scores, and training/validation curves will be discussed below in more detail

**Training/Validation**

Both models were trained with 120 epochs and a patience of 25 epochs. The patience refers to how many epochs the function will wait before terminating the fitting process. Criteria for termination is if the validation MSE has not decreased within the patience limit. With model checkpoint being used, we can be assured the best model will be saved if/when the fitting process terminates. Figure 6 shows the training curves for the two models.

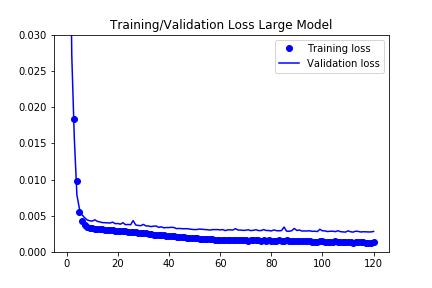
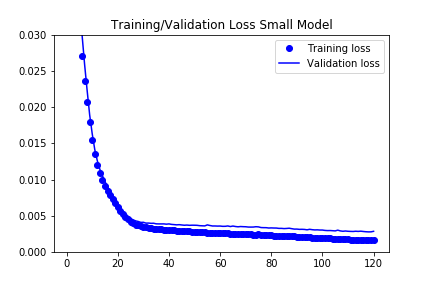


Figure 6: Loss curves for both models being tested

From examining the curves, the larger model reached a lower MSE faster. This was expected though given the increased size of the LSTM layer. Both also had good separation on the validation curve from the training curve displaying that overfitting is not occurring yet. The plateauing of the curves hints that either more capacity could be added to the network, or both have reached an excellent point of performance. Given how low the MSE is on a normalized set (so an MSE between 0-1), the latter is more likely the answer. Increasing the capacity would seem to lead to diminishing returns as is evident from the two models already shown. Table 4 gives the MSE scores for the two models on the validation set.

|  |  |
| --- | --- |
| Model | MSE |
| Small | 0.00277 |
| Large | 0.00271 |

Table 4: MSE scores for the two models

The increased network size only saw marginal improvements in reducing the MSE, thus validating that a simple network would be just as effective in this case.

Moving forward to testing, the large model was kept just due to the better performance, even if it was marginal.

**Testing**

Weather and climate zones change particularly as latitude changes. A relationship exists between latitude and temperature which is evident as one moves from the equator to the north pole. Therefore, the first round of testing looked at different points along the same latitude and pressure level, but varied the longitude. Table 5 shows the results of these tests along with location of the coordinate tested.

|  |  |  |  |
| --- | --- | --- | --- |
| Pressure | Lat | Long | MSE |
| 2000 | 40 | -150 | 0.00254 |
| 2000 | 40 | -110 | 0.00208 |
| 2000 | 40 | -75 | 0.00199 |
| 2000 | 40 | -35 | 0.00191 |
| 2000 | 40 | 0 | 0.00200 |
| 2000 | 40 | 35 | 0.00104 |
| 2000 | 40 | 150 | 0.00177 |
| 2000 | 40 | 180 | 0.00201 |

Table 5: MSE scores for model tested against different longitudes

The results from the testing show that the model accurately predicts the sequences to a high degree. Figure 7 shows one example of the predicted overlaid with the original data to show the fit of accuracy.

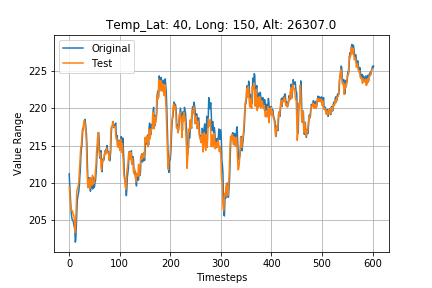
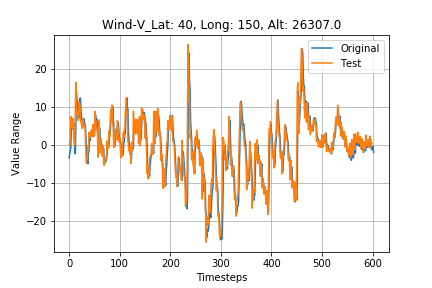
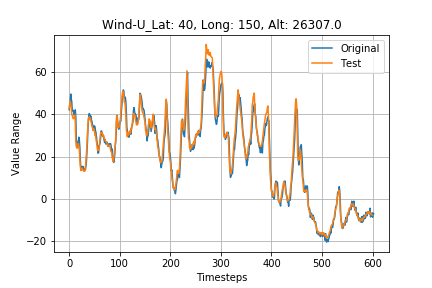
 

Figure 7: Charts of predicted values over original values for the labeled coordinate

The results from this first round of testing were extremely promising, so the next step was to change the latitude to the opposite hemisphere where the weather would be quite different and see how it would fare. These results are shown in Table 6 along with a chart of one of the coordinates in Figure 8.

|  |  |  |  |
| --- | --- | --- | --- |
| Pressure | Lat | Long | MSE |
| 2000 | -40 | -150 | 0.00415 |
| 2000 | -40 | -110 | 0.00249 |
| 2000 | -40 | -75 | 0.00500 |
| 2000 | -40 | -35 | 0.00563 |
| 2000 | -40 | 0 | 0.00521 |
| 2000 | -40 | 35 | 0.00425 |
| 2000 | -40 | 75 | 0.00369 |
| 2000 | -40 | 110 | 0.00296 |
| 2000 | -40 | 150 | 0.00539 |
| 2000 | -40 | 180 | 0.00515 |

Table 6: MSE scores for model tested against opposite latitude and varying longitudes

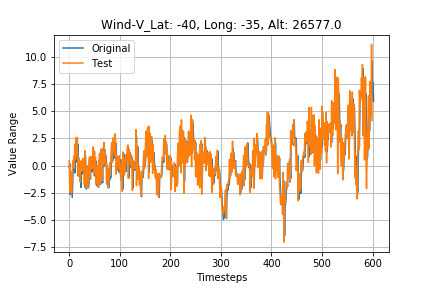
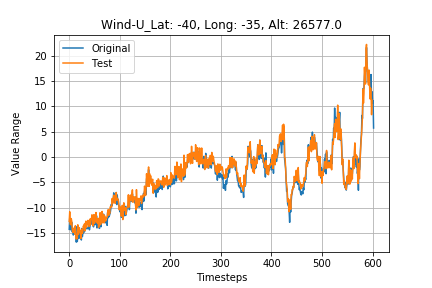
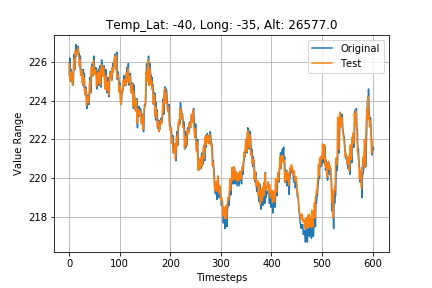


Figure 8: Charts of predicted values over original values for the labeled coordinate

While the results where not as good as the previous set, they were by no means terrible. They were still exceptionally good which is even more promising considering the weather differences on the opposite hemisphere. The next check of model generalizability was to look at changing the pressure layer, or more figuratively decreasing the altitude. Again, this is an area that also sees drastic weather pattern changes and would be an excellent test for the model. Table 7 shows MSE results for these coordinates, and Figure 9 gives a display of the results for one coordinate.

|  |  |  |  |
| --- | --- | --- | --- |
| Pressure | Lat | Long | MSE |
| 40000 | 40 | -150 | 0.00810 |
| 40000 | 40 | -110 | 0.00682 |
| 40000 | 40 | -75 | 0.00736 |
| 40000 | 40 | -35 | 0.00676 |
| 40000 | 40 | 0 | 0.00542 |
| 40000 | 40 | 35 | 0.00687 |
| 40000 | 40 | 75 | 0.00584 |
| 40000 | 40 | 110 | 0.00719 |
| 40000 | 40 | 150 | 0.00675 |
| 40000 | 40 | 180 | 0.00616 |

Table 7: MSE scores for model tested against varying longitudes and lower altitude

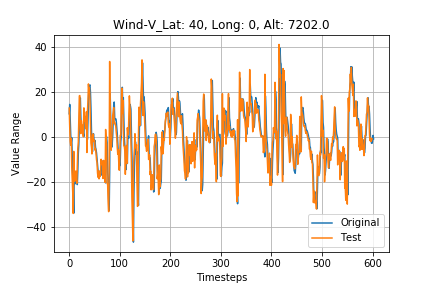
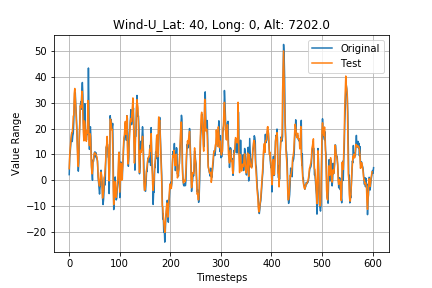
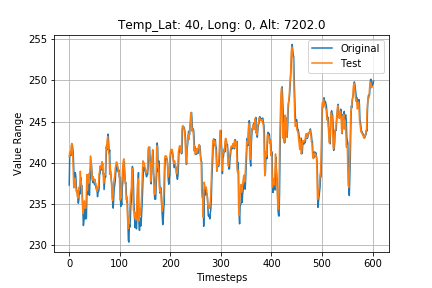


Figure 9: Charts of predicted values over original values for the labeled coordinate

The results were slightly worse which is expected when you move lower in altitude, but they are still very good. They showcase just how good the model can generalize across latitudes and through altitudes. Even examining these charts, you can see how the range in the wind speeds and temperature has drastically increased, but the model handled it very well for prediction accuracy. The last thing looked at for testing was to randomly pick 50 coordinates at three different pressure levels and test the model. Altogether that is 150 different sequences the model is being tested against. The full results can be found in Appendix A, since displaying the full table would take a couple pages. Figure 10 shows a sample of one of the coordinates

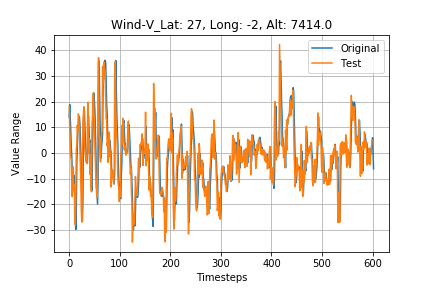
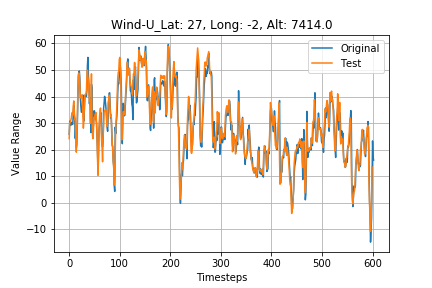
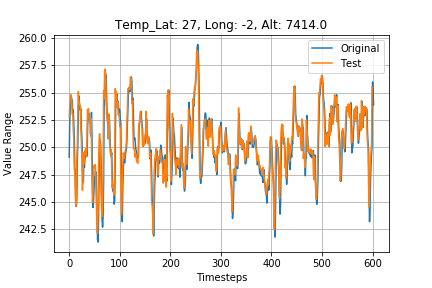


Figure 10: Charts of predicted values over original values for the labeled coordinate

After reviewing the results from testing phase the model performed admirably on all coordinates. The places if it issues involved coordinates in the southern hemisphere that were at lower altitudes. This is expected considering the model was trained on a coordinate in the northern hemisphere. The fact though that the model was still able to generalize incredibly well across all the points adds validity to model and its ability to predict. The next step would be to increase the time steps and forecast further out.

**Conclusion**

In reviewing the results of the project, we can see that the architecture for the LSTM network worked incredibly well for predicting temperature and wind speeds at varying locations across the planet. The weaknesses in the model only seemed to involve predictions in the lower atmosphere, within the southern hemisphere. That indicates that future work that can be completed when more time is available.

One piece of future work involves testing the limits of the model to much lower altitude levels. The lowest tested here was at the 15th of 31 pressure levels which means there are 14 more lower levels to test model performance. The prediction is that it will get worse, but that may not be the case.

Another piece of future work will involve training the model to more timesteps using the same architecture, and see if the performance still holds. The goal would be to reach out to 2 days in the future which would be eight timesteps. This would help with C-17 operations as a two-day lead on weather would be more helpful for route planning than 12 hours.

The last piece of future work would involve building an architecture that accepts a cube of data, as in the entire planet, and predicts some timestep in the future the next iteration of weather across the planet. This is a far-reaching goal but would have major implications for weather modeling and the aviation business.

The results of this project will help feed the continued research of weather prediction. More importantly it gets closer to accomplishing the first task in optimizing route planning for C-17 missions, and increasing efficiency for fuel usage. Thus, saving the government and tax payers lots of money.

**References**

|  |  |
| --- | --- |
| 1. | U.S. Department of Defense, *Fiscal Year 2019 Operational Energy Budget Certification Report*, 2018 |
| 2. | U.S. Department of Defense, *United States Department of Defense Fiscal Year 202 Budget Request,* 2019 |
| 3. | C. Poland, “How the Air Force Got Smarter About its Aviation Fuel Use in 2018," 2018, accessed June 5, 2019.  <https://www.af.mil/News/Article-Display/Article/1711969/how-the-air-force-got-smarter-about-its-aviation-fuel-use-in> |
| 4. | I. Goodfellow, Y. Bengio, A. Courville, (2016) *Deep Learning*, MIT Press |
| 5. | Maqsood, I., Khan, M. R., & Abraham, A. (2004). An ensemble of neural networks for weather forecasting. *Neural Computing & Applications*, *13*(2), 112-122. |
| 6. | Culclasure, A. (2013). Using neural networks to provide local weather forecasts. |
| 7. | Abhishek, K., Singh, M. P., Ghosh, S., & Anand, A. (2012). Weather forecasting model using artificial neural network. *Procedia Technology*, *4*, 311-318. |
|  |  |
|  |  |
|  |  |

**Appendix A**

Table of full results for testing model against multiple random coordinates.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Pressure | Lat | Long | Altitude | MSE |
| 40000 | -85 | -3 | 6827.291 | 0.003603 |
| 40000 | -85 | 155 | 6814.038 | 0.002976 |
| 40000 | -78 | 8 | 6907.644 | 0.004824 |
| 40000 | -73 | 115 | 6803.441 | 0.004684 |
| 40000 | -69 | -73 | 6923.474 | 0.004515 |
| 40000 | -67 | -117 | 7009.971 | 0.008233 |
| 40000 | -64 | 58 | 7046.549 | 0.008455 |
| 40000 | -60 | 173 | 7078.432 | 0.006582 |
| 40000 | -55 | -169 | 7262.761 | 0.006545 |
| 40000 | -49 | -3 | 7150.393 | 0.00591 |
| 40000 | -47 | 140 | 7300.287 | 0.007974 |
| 40000 | -43 | -59 | 7253.667 | 0.005828 |
| 40000 | -43 | 109 | 7346.193 | 0.005519 |
| 40000 | -33 | -27 | 7537.934 | 0.003569 |
| 40000 | -30 | 72 | 7547.009 | 0.004605 |
| 40000 | -20 | 120 | 7595.251 | 0.00596 |
| 40000 | -19 | -47 | 7594.704 | 0.003924 |
| 40000 | -19 | 10 | 7588.877 | 0.005975 |
| 40000 | -15 | 57 | 7588.782 | 0.003989 |
| 40000 | -13 | -174 | 7595.724 | 0.006403 |
| 40000 | -9 | -24 | 7592.528 | 0.00965 |
| 40000 | -8 | 167 | 7590.014 | 0.008593 |
| 40000 | -6 | -158 | 7589.487 | 0.011511 |
| 40000 | -4 | 146 | 7588.316 | 0.008962 |
| 40000 | 1 | -149 | 7587.389 | 0.011594 |
| 40000 | 3 | 171 | 7588.178 | 0.00527 |
| 40000 | 8 | 8 | 7592.495 | 0.007156 |
| 40000 | 13 | -114 | 7591.357 | 0.003904 |
| 40000 | 14 | -94 | 7591.317 | 0.00338 |
| 40000 | 19 | 43 | 7559.495 | 0.003447 |
| 40000 | 22 | 65 | 7551.463 | 0.002953 |
| 40000 | 23 | -74 | 7522.062 | 0.003732 |
| 40000 | 26 | 109 | 7509.631 | 0.003767 |
| 40000 | 27 | -2 | 7413.677 | 0.006164 |
| 40000 | 32 | -164 | 7419.415 | 0.005677 |
| 40000 | 32 | -130 | 7442.54 | 0.00377 |
| 40000 | 38 | 1 | 7224.425 | 0.005945 |
| 40000 | 39 | -141 | 7292.485 | 0.006109 |
| 40000 | 41 | -4 | 7190.504 | 0.006864 |
| 40000 | 43 | 1 | 7173.298 | 0.006378 |
| 40000 | 50 | 78 | 6968.053 | 0.00733 |
| 40000 | 53 | -144 | 6804.617 | 0.013115 |
| 40000 | 54 | -158 | 6767.416 | 0.011543 |
| 40000 | 57 | 81 | 6753.406 | 0.010352 |
| 40000 | 58 | -149 | 6619.082 | 0.011347 |
| 40000 | 64 | 110 | 6610.067 | 0.010495 |
| 40000 | 69 | 149 | 6528.073 | 0.008035 |
| 40000 | 72 | 141 | 6506.996 | 0.006493 |
| 40000 | 87 | -166 | 6465.346 | 0.00398 |
| 40000 | 87 | -119 | 6470.111 | 0.003187 |
| 15000 | -88 | -164 | 13195.22 | 0.001362 |
| 15000 | -82 | 145 | 13210.8 | 0.001262 |
| 15000 | -80 | -95 | 13222.18 | 0.002013 |
| 15000 | -77 | -93 | 13238.55 | 0.001838 |
| 15000 | -76 | 155 | 13271.43 | 0.001799 |
| 15000 | -70 | -87 | 13291.12 | 0.001829 |
| 15000 | -70 | 101 | 13204.6 | 0.001643 |
| 15000 | -68 | -113 | 13371.91 | 0.002639 |
| 15000 | -67 | 160 | 13375.88 | 0.001641 |
| 15000 | -66 | 133 | 13296.89 | 0.002017 |
| 15000 | -64 | 128 | 13308.08 | 0.002208 |
| 15000 | -60 | -98 | 13430.32 | 0.001963 |
| 15000 | -60 | 54 | 13546.74 | 0.003275 |
| 15000 | -57 | -47 | 13418.02 | 0.001997 |
| 15000 | -57 | 146 | 13495.14 | 0.00292 |
| 15000 | -56 | 68 | 13567.2 | 0.003805 |
| 15000 | -55 | -98 | 13509.17 | 0.002707 |
| 15000 | -49 | 124 | 13691.56 | 0.004677 |
| 15000 | -47 | -29 | 13674.5 | 0.003649 |
| 15000 | -43 | 38 | 13879.17 | 0.003281 |
| 15000 | -40 | -150 | 13895.89 | 0.003322 |
| 15000 | -40 | -110 | 13910.25 | 0.003411 |
| 15000 | -40 | -75 | 13890.99 | 0.003511 |
| 15000 | -40 | -35 | 13921.1 | 0.003406 |
| 15000 | -40 | 0 | 13926.14 | 0.00375 |
| 15000 | -40 | 35 | 13925.53 | 0.003098 |
| 15000 | -40 | 75 | 13933.54 | 0.005194 |
| 15000 | -40 | 110 | 13956.67 | 0.003986 |
| 15000 | -40 | 150 | 13961.61 | 0.004851 |
| 15000 | -40 | 180 | 13912.95 | 0.004306 |
| 15000 | -37 | 89 | 14000.31 | 0.005793 |
| 15000 | -36 | -10 | 14070.84 | 0.003889 |
| 15000 | -35 | 4 | 14036.7 | 0.003947 |
| 15000 | -27 | 116 | 14148.58 | 0.003745 |
| 15000 | -22 | 113 | 14178.08 | 0.00377 |
| 15000 | -20 | -92 | 14274.31 | 0.004073 |
| 15000 | -18 | -145 | 14221.08 | 0.004502 |
| 15000 | -17 | 35 | 14211.04 | 0.005252 |
| 15000 | -12 | -68 | 14256.78 | 0.008393 |
| 15000 | -12 | -19 | 14250.16 | 0.005998 |
| 15000 | -9 | 17 | 14238.45 | 0.008294 |
| 15000 | -2 | -58 | 14228.81 | 0.007857 |
| 15000 | 1 | 115 | 14219.56 | 0.009531 |
| 15000 | 2 | -35 | 14234.16 | 0.008919 |
| 15000 | 5 | -53 | 14231.24 | 0.006121 |
| 15000 | 9 | 45 | 14212.55 | 0.005236 |
| 15000 | 14 | 18 | 14220.34 | 0.005799 |
| 15000 | 15 | -105 | 14217.69 | 0.004909 |
| 15000 | 21 | -53 | 14193.74 | 0.003314 |
| 15000 | 22 | 101 | 14099.52 | 0.003657 |
| 15000 | 38 | -100 | 13754.37 | 0.003434 |
| 15000 | 40 | -150 | 13688.49 | 0.005319 |
| 15000 | 40 | -110 | 13703.92 | 0.003709 |
| 15000 | 40 | -75 | 13628.64 | 0.004069 |
| 15000 | 40 | -35 | 13662.12 | 0.003545 |
| 15000 | 40 | 0 | 13658.52 | 0.00296 |
| 15000 | 40 | 35 | 13641.52 | 0.00289 |
| 15000 | 40 | 75 | 13636.73 | 0.003529 |
| 15000 | 40 | 110 | 13653.11 | 0.005579 |
| 15000 | 40 | 150 | 13648.87 | 0.004217 |
| 15000 | 40 | 180 | 13633.82 | 0.004151 |
| 15000 | 41 | 54 | 13611.01 | 0.00295 |
| 15000 | 49 | -4 | 13487.17 | 0.003097 |
| 15000 | 53 | 52 | 13241.01 | 0.002551 |
| 15000 | 72 | -47 | 12486.52 | 0.003439 |
| 15000 | 76 | -25 | 12432.33 | 0.003037 |
| 15000 | 79 | 136 | 12439.76 | 0.003079 |
| 15000 | 80 | -3 | 12391.22 | 0.002947 |
| 15000 | 82 | 28 | 12384.01 | 0.002792 |
| 15000 | 86 | -22 | 12378.62 | 0.002182 |
| 2000 | -84 | -22 | 26044.18 | 0.001298 |
| 2000 | -81 | 12 | 26142.27 | 0.00124 |
| 2000 | -75 | -97 | 26000.18 | 0.001732 |
| 2000 | -67 | 25 | 26478.4 | 0.001327 |
| 2000 | -66 | -126 | 26158.03 | 0.001334 |
| 2000 | -61 | -137 | 26301.44 | 0.001596 |
| 2000 | -50 | 91 | 26461.18 | 0.001674 |
| 2000 | -47 | 155 | 26527.09 | 0.001165 |
| 2000 | -45 | 3 | 26618.27 | 0.006404 |
| 2000 | -45 | 42 | 26594.61 | 0.003335 |
| 2000 | -45 | 114 | 26493.78 | 0.001922 |
| 2000 | -44 | 123 | 26508.91 | 0.001689 |
| 2000 | -43 | -127 | 26486.34 | 0.001922 |
| 2000 | -39 | 93 | 26518.85 | 0.006149 |
| 2000 | -33 | 163 | 26517.61 | 0.005341 |
| 2000 | -30 | -121 | 26501.58 | 0.009746 |
| 2000 | -29 | 31 | 26503.16 | 0.012239 |
| 2000 | -28 | -109 | 26495.15 | 0.011491 |
| 2000 | -22 | 96 | 26461.26 | 0.022704 |
| 2000 | -19 | -86 | 26442.23 | 0.021855 |
| 2000 | -11 | -141 | 26408.18 | 0.018223 |
| 2000 | -6 | -175 | 26405.51 | 0.018469 |
| 2000 | -5 | -53 | 26412.96 | 0.016463 |
| 2000 | -2 | -19 | 26416.46 | 0.013584 |
| 2000 | -1 | 166 | 26412.46 | 0.014834 |
| 2000 | 5 | -120 | 26411.48 | 0.010361 |
| 2000 | 8 | 151 | 26408.64 | 0.017792 |
| 2000 | 12 | -152 | 26411.72 | 0.01758 |
| 2000 | 18 | -85 | 26431.43 | 0.02017 |
| 2000 | 26 | 76 | 26438.82 | 0.010136 |
| 2000 | 27 | 69 | 26441.36 | 0.007415 |
| 2000 | 32 | 56 | 26461.01 | 0.003648 |
| 2000 | 35 | 141 | 26390.37 | 0.002822 |
| 2000 | 44 | -17 | 26583.14 | 0.001742 |
| 2000 | 45 | -80 | 26409.31 | 0.001394 |
| 2000 | 45 | -43 | 26529.21 | 0.001552 |
| 2000 | 49 | 115 | 25994.01 | 0.001572 |
| 2000 | 50 | 157 | 25862.55 | 0.001338 |
| 2000 | 52 | -67 | 26186.49 | 0.001439 |
| 2000 | 53 | -112 | 25904.53 | 0.001614 |
| 2000 | 54 | -24 | 26316.56 | 0.001533 |
| 2000 | 58 | 14 | 26072.31 | 0.001286 |
| 2000 | 58 | 71 | 25592.71 | 0.001934 |
| 2000 | 59 | 31 | 25889.5 | 0.001236 |
| 2000 | 66 | -149 | 24699.88 | 0.001472 |
| 2000 | 73 | -114 | 24425.12 | 0.001475 |
| 2000 | 74 | -125 | 24357.57 | 0.001316 |
| 2000 | 82 | 17 | 24359.65 | 0.002031 |
| 2000 | 83 | -178 | 24152.74 | 0.001035 |
| 2000 | 89 | -151 | 24163.32 | 0.001206 |