Title : Shell.ai Hackathon 2021

Objective

Given a weather data containing different parameters of variables. The goal is to infer the future observation given the current observation. To be precise, we need to forecast four horizons, at 30 mins, 60 mins, 90 mins and 120 mins.

Note

I didn't use pandas or ipython notebook as I prefer to write my own function in native python. All necessary scripts are found in the directory.

List of methods tried

To start off, I went with simple machine learning algorithms such as SVM. But it did not produce the accuracy as expected. Next I have used xgboost as it was widely used for time-series. Here I got a validation accuracy of 85 %. Same goes with random forests. I have not incorporated any preprocessing up and till now.

Data Preprocessing

After analysing the training weather data, I found there are some cloud cover entries that have a huge negative value (-7999). Of course, deleting the rows will change the time order and might influence the accuracy of the model. So to avoid it, I took the positive value that is present nearest (simple recursive algorithm) to the current negative value. By default the entries that have cloud cover of -1 are removed. To detect outliers in the time series data, I used tsmoother python library also.

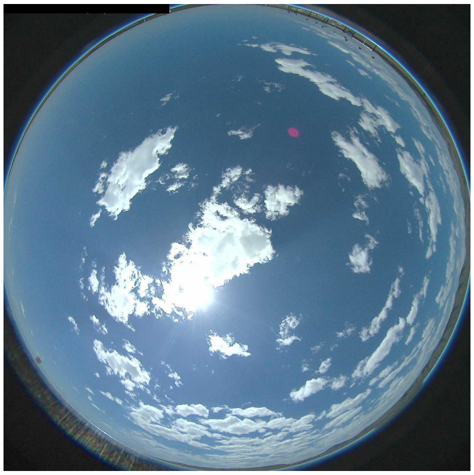
Model accuracy using only the weather data

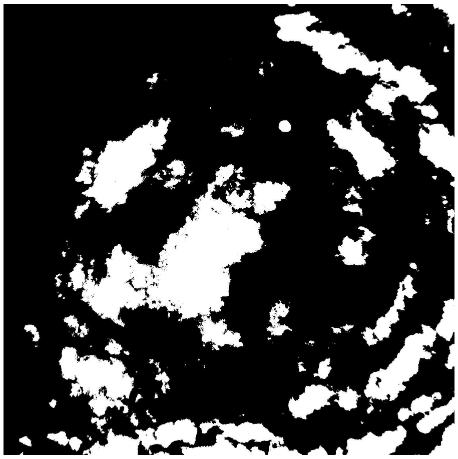
Here, I have used the LSTM deep learning method to predict the future value. The fixed window size of 100 is kept for all four horizons.

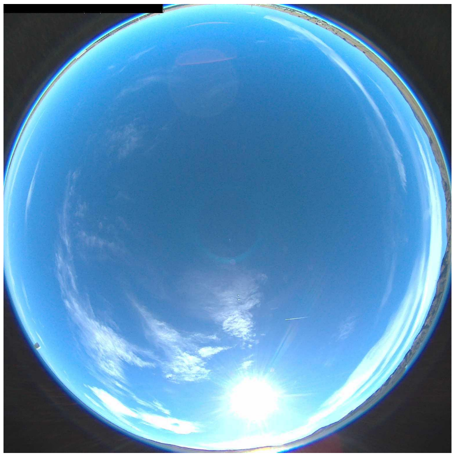
Use of sky camera image during training

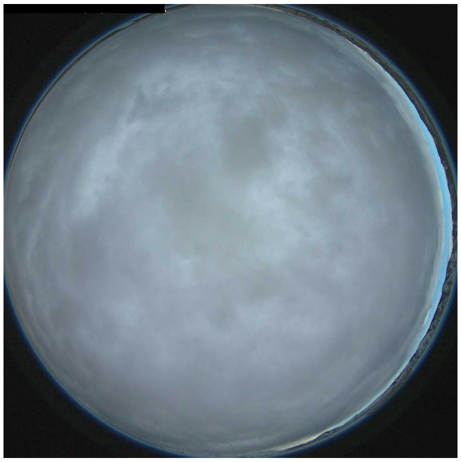
To efficiently use the sky camera image, I used the Defisheye library. Then I converted the color image into a binary value image. White color represents the clouds and black color represents sky.

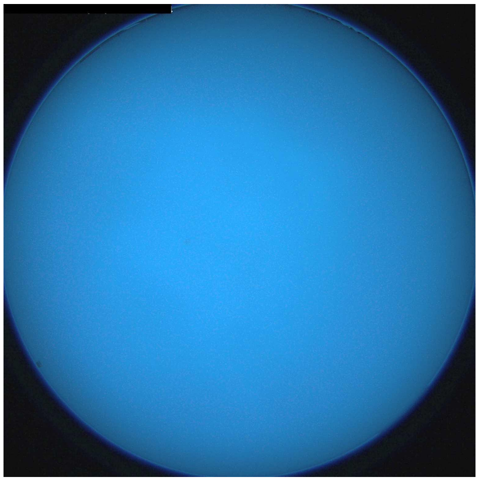
After that i used convolutional LSTM using a series of first 6 images to predict the value at the corresponding horizon.

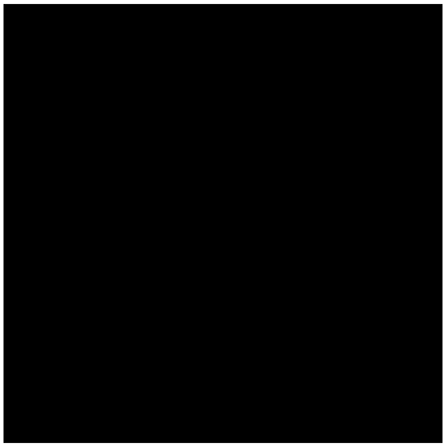












Finally, the weights of both weather trained model and conv-LSTM are concatenated to get the final output. Activation function of RelU, MAE loss is employed.