

MERLIN Documentation

Idea :-

It is a project which Visualizes and Predicts the corona cases around the world. It is self learning in nature.

Project:-

The project has seen over 200 iterations and was a deep dive into Deep Learning and Machine Learning. Right now the project has two versions CPU Model ,GPU Model Both have the same ideas but different architecture and right now the better in terms of Speed and Accuracy . The CPU Model was able to beat in terms of speed a Team of Researchers from University Of Reijka with almost the same accuracy. It only has a python Script and that is all that is required to run. Subhashree is trying to make a Front-End for the Project.

Project Working Technical:-

Both Projects Share almost the same Data EDA and Graphs.

CPU Model:-

The Model Uses MLP (Multi Layer Perceptron) Regressor from Scikit-Learn and uses a Custom made Training module which finds the best model with Number of Hidden layers , Activation Function ,Alpha Value as the iteration values for the model trainer.

The Models Excecution Time From Start to Finish is around 4 Min and I was able to achieve that using some Debugging and Finding The Correct Scaling (MinMax [0,1]).

GPU Model:-

The Model uses a Custom Made (Multi Layer Seequential Model) using Keras Libarary and uses a Custom Made Training Module Using RandomSearchCV, The Excecution Time is Around 6 Min there are some bugs in this model as the Accuracy is very low and High Training Time , It requires further Work.

Data Set:-

Both Use Data From SRK COVID-19 Dataset and use just 1 file from it namely (covid19Datset)

Data Set Link

--><https://www.kaggle.com/sudalairajkumar/novel-corona-virus-2019-data-set>

Key Points :-

CPU:-

Number of Versions - 100+

Execution Time - 180 Sec

Man Hours - 70-150 Hrs

Improvement Description :-

In this model I started with Linear Regression and then moved to Polynomial Regression but the problem was Low Accuracy , and Polynomial Regression was not good for extrapolation of Data due to Nature of Curve Forcing.

Then I moved onto MLP model , It was my first Deep Learning Project so the implementation could have been a bit better, But most of time was used to make the custom Training And Learning Feature , which uses a CSV to Track all the models.

Custom Learning was just automation of Train/Test Spilt Reduction.

It also detects a Pre-Trained model with high accuracy and implements that model while skipping the Training Feature otherwise trains the model