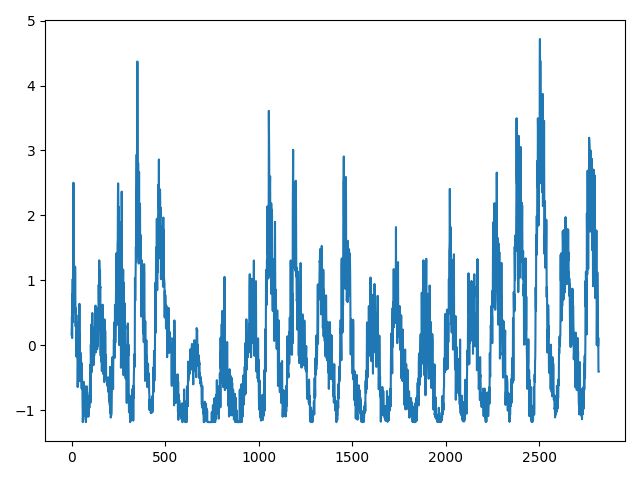
**CMP5133-Artificial Neural Networks**

**Homework 2 - Time-Delay Neural Networks vs Recurrent Neural Networks**

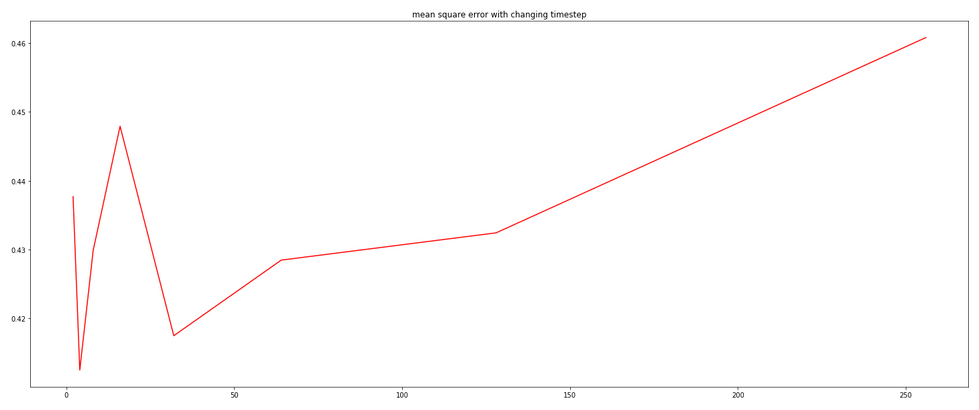
**Habip Hakan Isler**

**1906437**

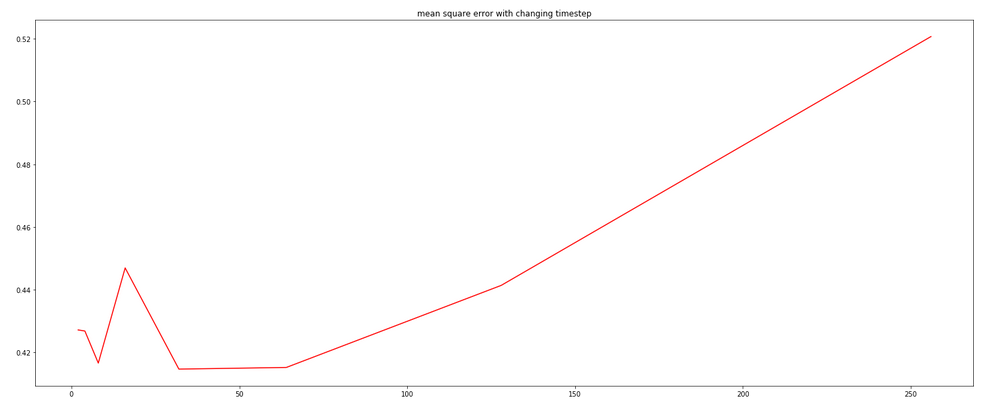
1. **Plot the original data set**

****

1. **Plot of mean squared error of LSTM-RNN with changing timestep on test set**

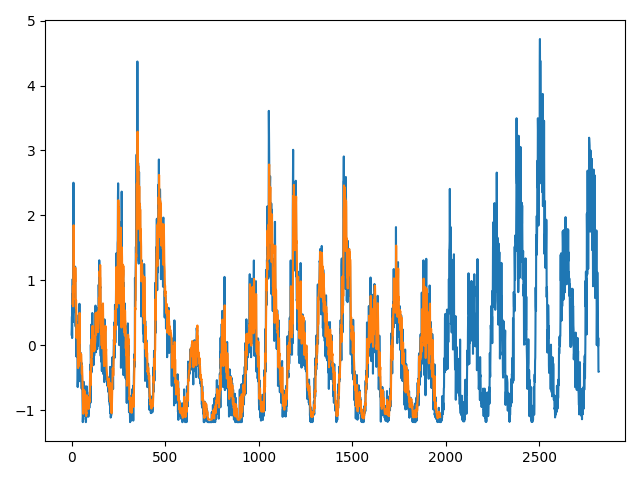


1. **Plot of mean squared error of MLP with changing timestep on test set**

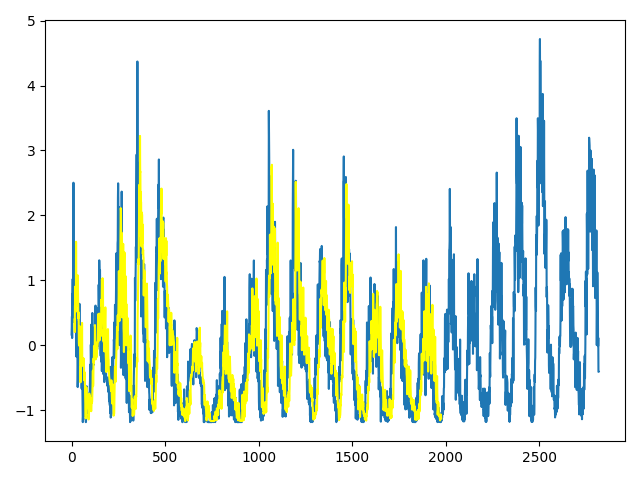


1. **Plot of the predictions of best MLP and LSTM-RNNs models along with the actual values of training set**

*4.1 Plot of the predictions of LSTM-RNNs model along with the actual values of training set* ***(****Orange- Training Set Blue – Actual Values****)***

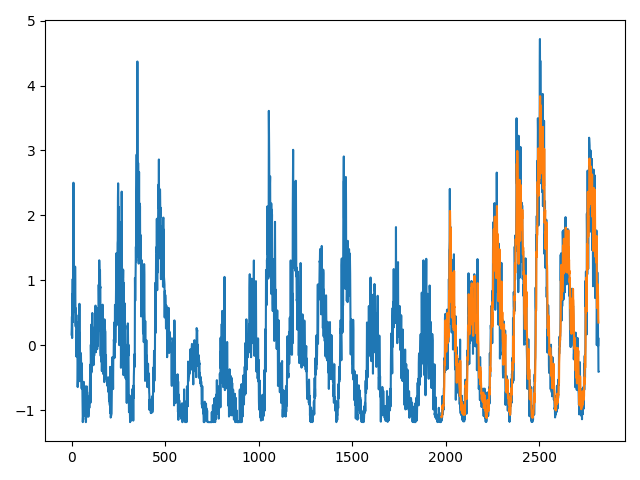
****

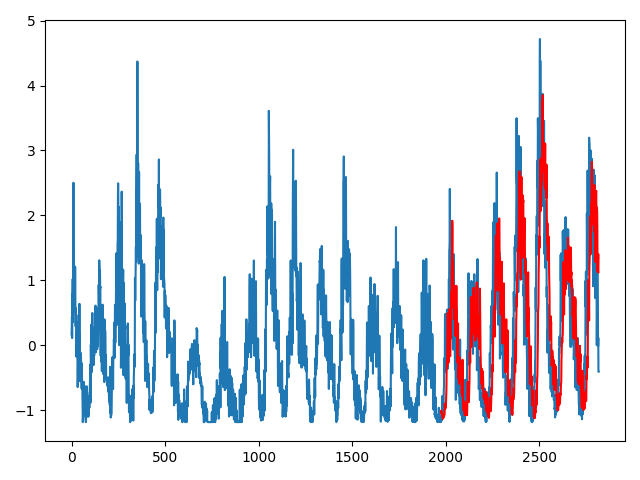
*4.2 Plot of the predictions of MLP model along with the actual values of training set* ***(****Yellow- Training Set , Blue – Actual Values****)***

****

1. **Plot of the predictions of best MLP and LSTM-RNNs models along with the actual values of test set**

*5.1 Plot of the predictions of LSTM-RNNs model along with the actual values of test set* ***(****Orange- Test Set , Blue – Actual Values****)***

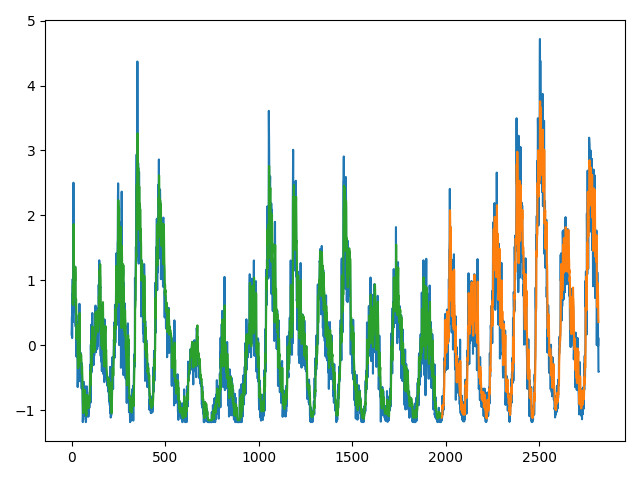
****

 *5.2 Plot of the predictions of MLP model along with the actual values of test set*

***(****Red- Test Set , Blue – Actual Values****)***

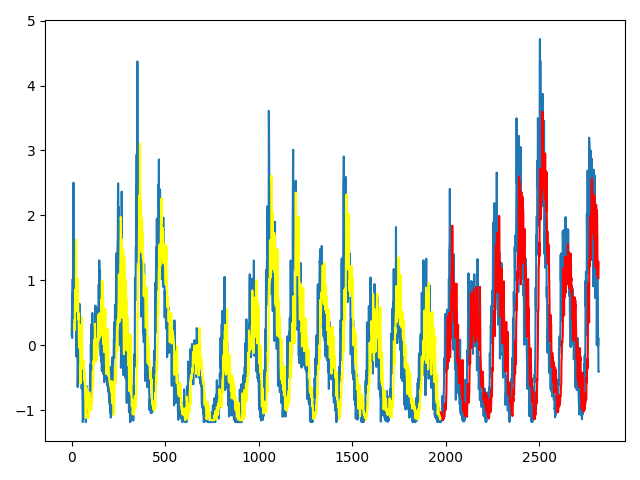
**Predictions of Best LSTM\_RNN Model**

*Green- Train Set , Orange-Test Set , Blue - Actual*

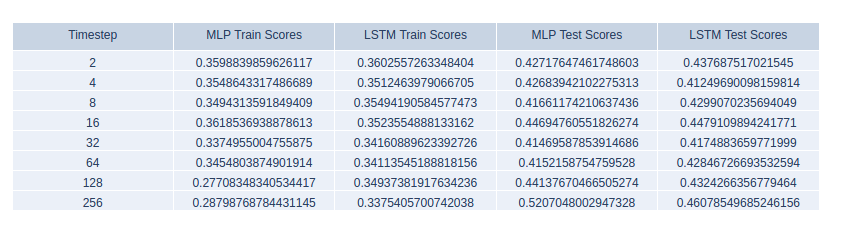
****

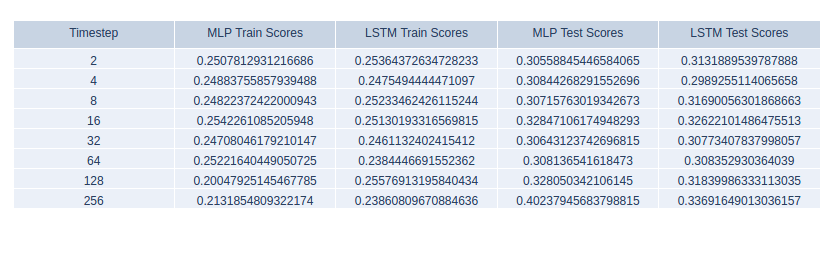
**Predictions of Best MLP Model**

Yellow- Train Set , Red -Test Set, Blue Actual



1. **Table including the Mean Squared Error and Mean Absolute error for both training and test sets**

**Root Mean Square Error**

**Mean Absolute Error**

**Discussions**

Hyperparameter optimization experiments show that for MLP and LSTM-RNN models, the minimum error can be achieved by only one hidden layer. In neural networks, more layers mean more complexity, so it can be inferred that a hidden layer is enough to understand the pattern in the data set we selected, and adding more layers will lead to overfitting, resulting in more High test error. The hidden unit number and dropout rate in each layer are also related to the selected data set, and it turns out that the two models have the same number. In MLP, the optimal batch size is larger than in LSTM-RNN.

Of course, through more experiments, more accurate values ​​can be obtained for the hyperparameters. Moreover, the training and testing time of LSTM-RNN is relatively high (I did not measure it, but it is worth noting) because its structure is more complex and understandable.

When we look at the mean square error of the time steps of the two proposed models, we can see that for time steps below 64, the errors are close to each other. For some values, LSTM-RNN is better, and for other values, MLP is better. However, as we continue to increase the time step size, LSTM-RNN achieves a lower mean square error on the same data set. It can be concluded that because LSTM can better learn long-term dependencies, MLP can only compete with LSTM for lower time step values.

Trying LSTM 1 step on MLP makes no sense, because when the data is processed only once, its behavior will be the same as MLP with the same network architecture. The idea of ​​LSTM is to store the final output in memory to use it as input for subsequent steps.

By using the best hyperparameters and best time step values ​​of the two models, the best model is determined as the model with the smallest mean square error on the test set. For MLP, the optimal time step value is 32, and for the LSTM-RNN model, the optimal time step value is 4, and these time step values ​​are used for prediction.The predictions of the training and test sets of the best MLP and LSTM-RNN models are very good. This can be inferred from the actual and predicted value charts, and the errors of the training and test sets can also be calculated. The best MLP error on the training set is a little higher than the LSTM, and the best LSTM obtained better results on the test set. Since we are concerned about generalization errors, we can conclude that LSTM is better on time series data sets.

In the last step, the model test and training prediction for each time step test are displayed on the table root mean square error (RMSE) and mean absolute error (MAE). MAE is just the average of absolute errors. In RMSE, we first square the error and then root. This means that when there is a large error in the prediction, we will get a larger RMSE than MAE; when all errors are equal, the RMSE will be equal to MAE. Therefore, when we need to punish large errors to a greater extent, we should pay attention to RMSE, or when we are concerned about average errors, MAE can be used.

For both architectures, the RMSE value is always higher than the MAE value, which can be explained by the fact that some predicted values ​​are very different from the actual values. For MLP, the minimum MAE is obtained in a time step shorter than the minimum RMSE. For LSTM, the minimum value of the two is the same. For both cases, the test error is higher than the training error, which is expected, and as we increase the time step, the difference will become larger and larger.