Documentation

Project Description

In this project, gold closing price prediction was made. The dataset downloaded from kaggle from this link :
https://www.kaggle.com/datasets/faisaljanjua0555/daily-gold-price-historical-dataset/data
This project, which is a time series forecasting problem, uses time series data.
The dataset has 5703 data, features are : Date, Open, High, Low, Close, Volume, Currency.
Date ranges from 2000-01-04 to 2022-09-02, Open 257 to 2.08k, High 259 to 2.09k, Low 255 to 2.05k, Close 257 to 2.07k, Volume 0 to 817k and Currency is USD.

Data Preprocessing

- Checked whether there is a null value in the dataset.
- viewed number of unique value for each column
- Irrelevant or without variance feature deleted (currency)
- MinMax scaling used.

Data Visualization

- [Open, High, Low, Close, Volume] feature historical line charts are created.
- [15, 30, 60, 90, 180, 360] day list is used to plot mechanism of day line charts. 60 day is selected for feature extraction.
- Daily Return is calculated and plotted, histogram plot shows that distribution is gaussian.

Feature Engineering

- In this section several features are created, they are RSI, MACD, MACD_signal, MACD_hist, BB_upper, BB_middle and BB_lower.
- For creation parameters are folowing : RSI: [period = 14], MACD: [fastperiod=12, slowperiod=26, signalperiod=9], BBANDS: [timeperiod=20, nbdevup=2, nbdevdn=2, matype=0].
- After the creation process, these newly created features have nan values. I have imputed these values by predicting them based on other data. LSTM is used in the prediction process. Since they are also time series data as we see their line chart, this nan value prediction approach to impute them and using the LSTM model in this process is very suitable.

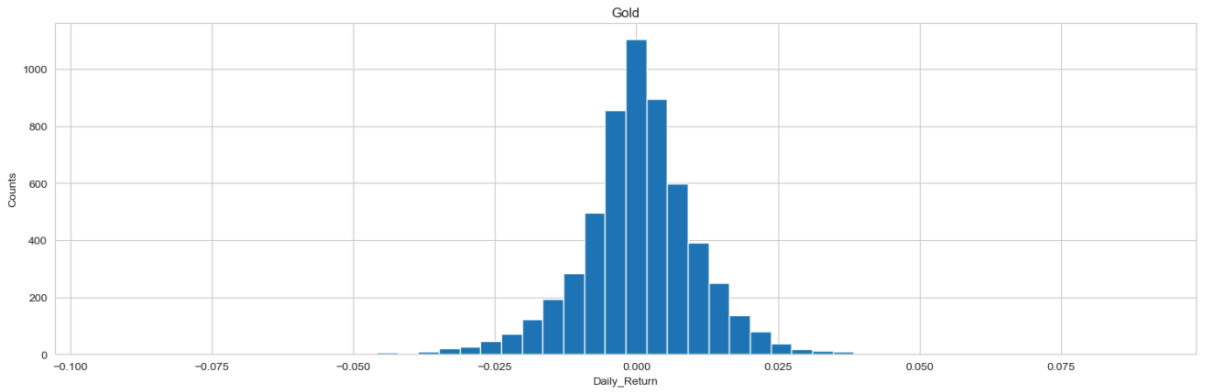
Prediction Phase

- 3 models and different techniques are used. Close feature selected as target feature.
- First is the LSTM model that uses the above created features and mechanism of day 60 features.
- Second is using only the “Close” feature and sliding window approach used.
- Third, LightGBM is used to see what will be the result.
- Models are not overfitted, both train and test accuracies are compared and always train is higher and both results are consistent, plotted loss curves also proves that. Early stopping is used to prevent overfitting and shorten the training time of the models. Batch size is 64, for optimizer adam optimizer and 3 hidden layers is used, for loss calculation mean squared error is used. Hidden layer size and neuron sizes are selected by trying several training processes. Since this project deals with time series data, not classification, mean squared error calculation is used to calculate accuracies and the other methods are also tried but not bring good results.

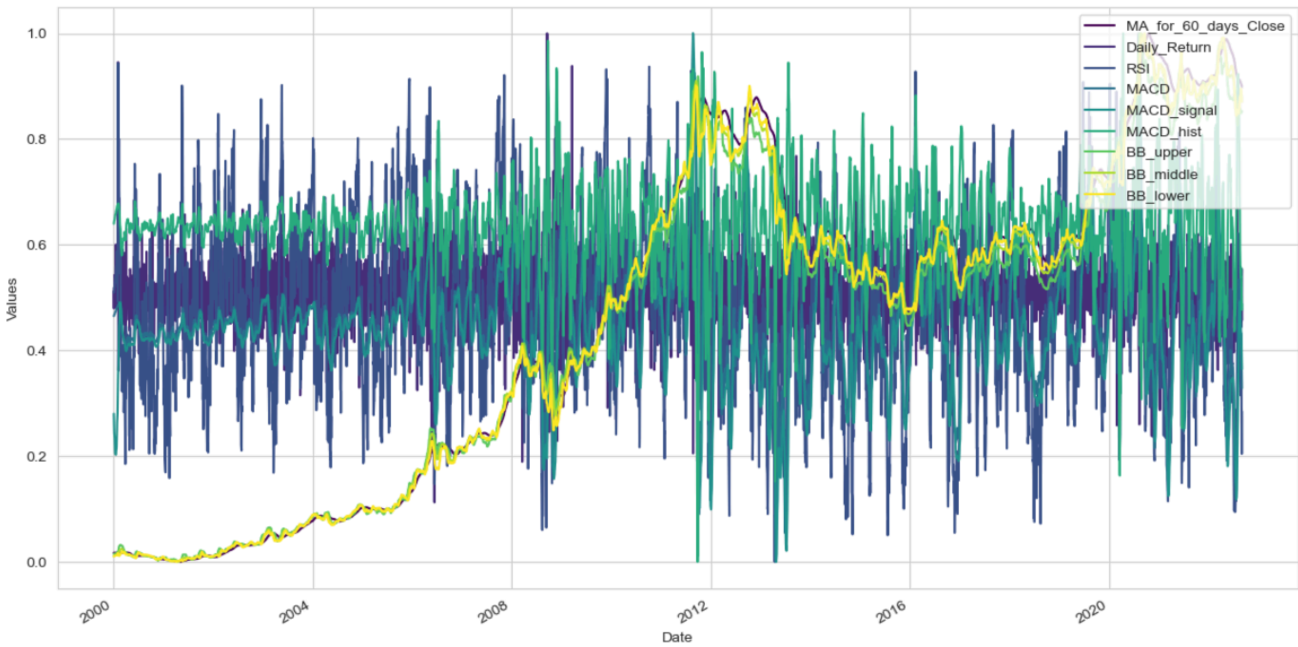
Some Visualizations

Mechanism of day:

Daily Return feature histogram:

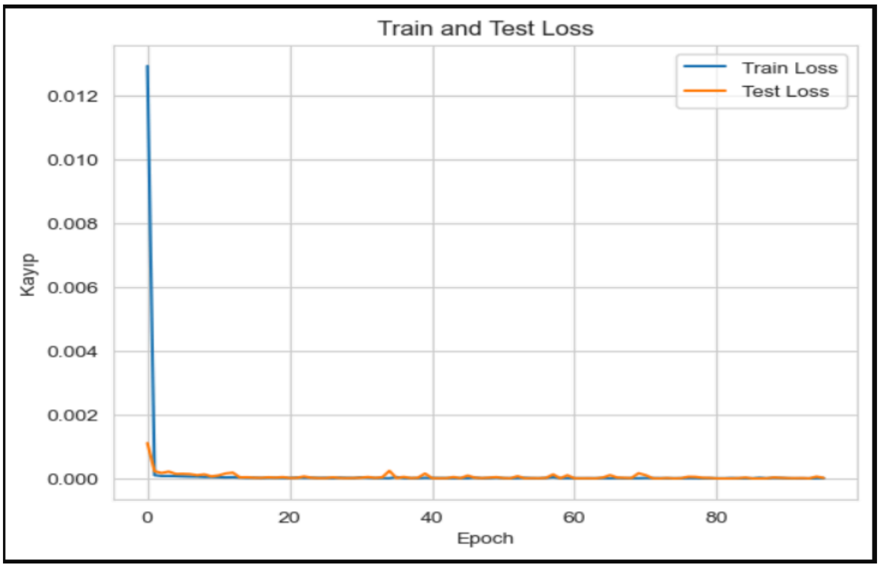


After imputing nan values, features graphs:



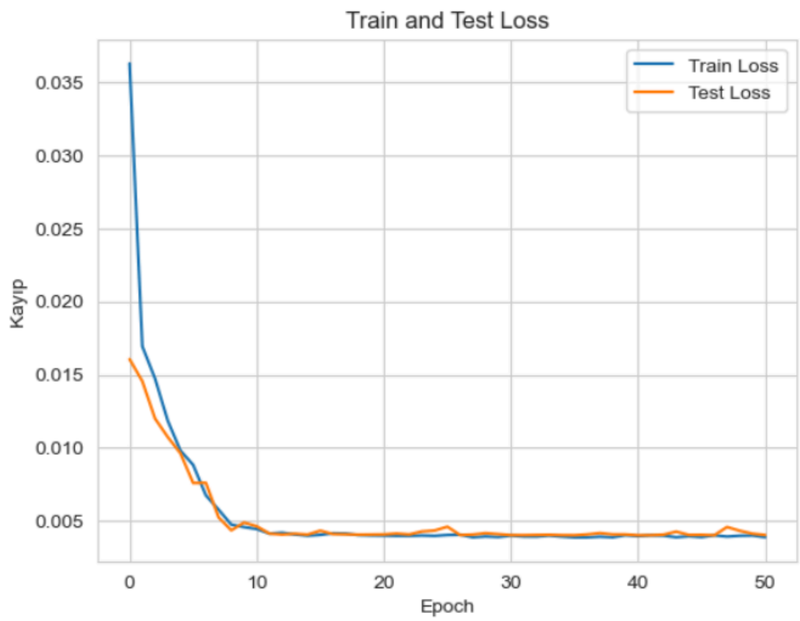
Imputation results:

'MA_for_60_days_Close':



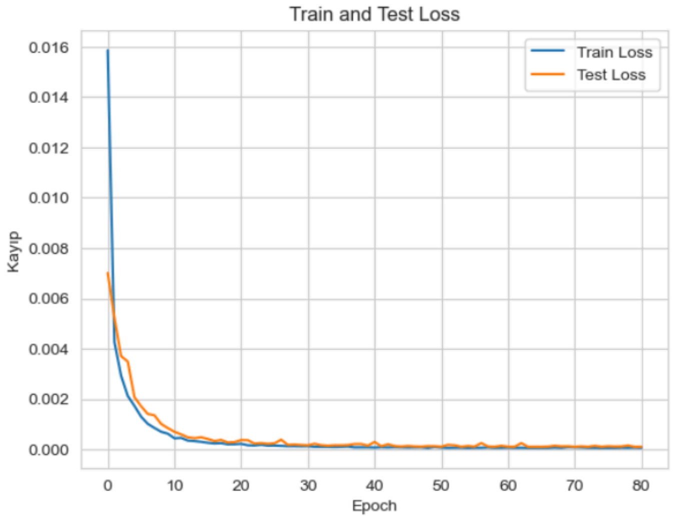

```
Train Result
138/138 [==============================] - 2s 12ms/step - loss: 3.4768e-06
138/138 [==============================] - 2s 12ms/step
Loss: 3.4768e-06
MAPE: 3.4768e-06
Accuracy: 0.9999965232122141
RMSE: 0.0018646146481091001
=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*
Test Result
37/37 [==============================] - 1s 13ms/step - loss: 5.0977e-06
37/37 [==============================] - 0s 12ms/step
Loss: 5.0977e-06
MAPE: 5.0977e-06
Accuracy: 0.9999949022959885
RMSE: 0.0022578095604992056
```

'RSI':



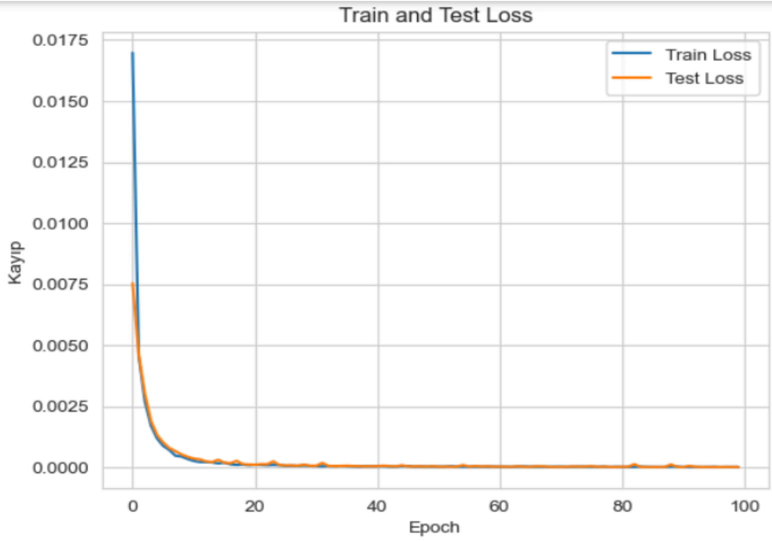

```
Train Result
141/141 [==============================] - 1s 7ms/step - loss: 0.0038
141/141 [==============================] - 1s 6ms/step
Loss: 0.0038032883
MAPE: 0.0038032873
Accuracy: 0.9961967126770845
RMSE: 0.06167079797534196
=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*
Test Result
37/37 [==============================] - 0s 7ms/step - loss: 0.0040
37/37 [==============================] - 0s 6ms/step
Loss: 0.0040270695
MAPE: 0.0040270697
Accuracy: 0.995972930277652
RMSE: 0.0634591973030549
```

'MACD':



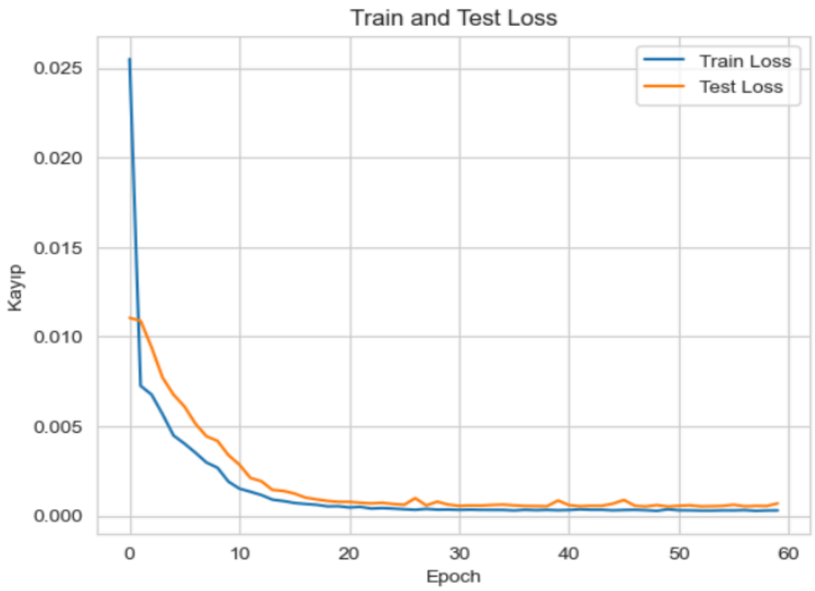

```
Train Result
140/140 [==============================] - 1s 10ms/step - loss: 5.7321e-05
140/140 [==============================] - 2s 9ms/step
Loss: 5.73211e-05
MAPE: 5.73211e-05
Accuracy: 0.9999426788767488
RMSE: 0.0075710714731260275
=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*
Test Result
37/37 [==============================] - 0s 11ms/step - loss: 1.1206e-04
37/37 [==============================] - 0s 10ms/step
Loss: 0.0001120573
MAPE: 0.0001120572
Accuracy: 0.9998879427509044
RMSE: 0.010585709664240646
```

'MACD_signal':



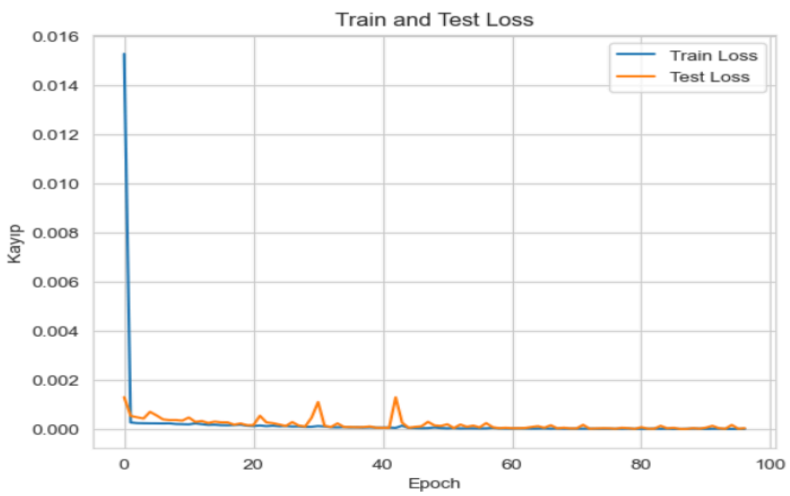

```
Train Result
140/140 [==============================] - 1s 10ms/step - loss: 8.4874e-06
140/140 [==============================] - 2s 9ms/step
Loss: 8.4874e-06
MAPE: 8.4874e-06
Accuracy: 0.99999151258831
RMSE: 0.002913316270168876
=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*
Test Result
37/37 [==============================] - 0s 10ms/step - loss: 1.3862e-05
37/37 [==============================] - 0s 10ms/step
Loss: 1.38621e-05
MAPE: 1.38621e-05
Accuracy: 0.9999861378766053
RMSE: 0.003723187262905818
```

'MACD_hist'



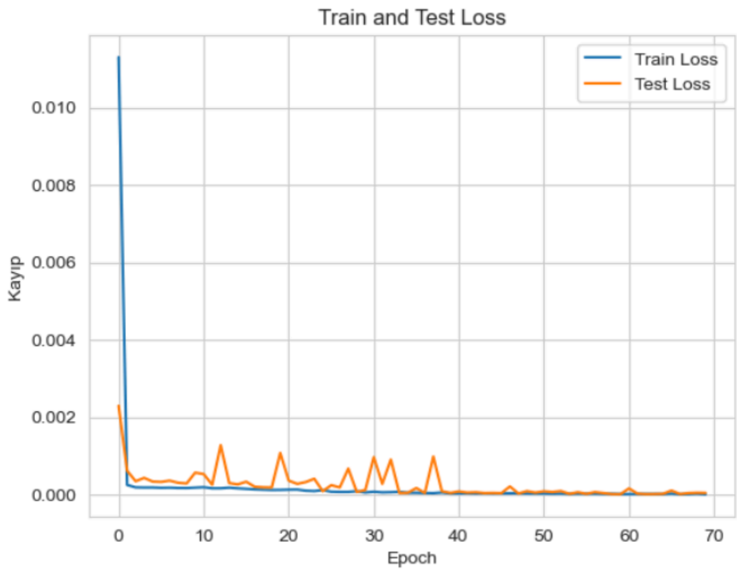

```
Train Result
140/140 [==============================] - 1s 10ms/step - loss: 2.6785e-04
140/140 [==============================] - 2s 9ms/step
Loss: 0.0002678458
MAPE: 0.0002678457
Accuracy: 0.9997321543350205
RMSE: 0.016365991108988304
=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*
Test Result
37/37 [==============================] - 0s 9ms/step - loss: 5.1785e-04
37/37 [==============================] - 0s 8ms/step
Loss: 0.0005178502
MAPE: 0.0005178502
Accuracy: 0.999482149791996
RMSE: 0.022756322374321136
```

'BB_upper':



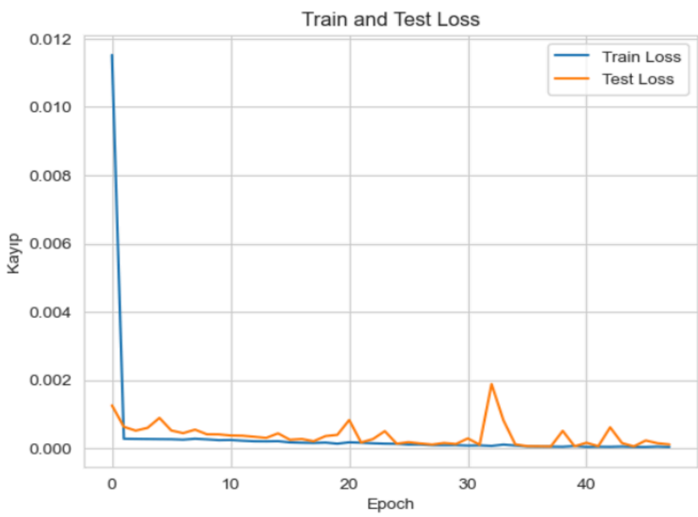

```
Train Result
141/141 [==============================] - 1s 8ms/step - loss: 1.1053e-05
141/141 [==============================] - 2s 7ms/step
Loss: 1.10531e-05
MAPE: 1.10531e-05
Accuracy: 0.999988946862688
RMSE: 0.003324625890533163
=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*
Test Result
37/37 [==============================] - 0s 7ms/step - loss: 1.5087e-05
37/37 [==============================] - 0s 6ms/step
Loss: 1.50871e-05
MAPE: 1.50871e-05
Accuracy: 0.999984912948122
RMSE: 0.003884205437156792
```

'BB_middle':




```
Train Result
141/141 [==============================] - 1s 8ms/step - loss: 1.4955e-05
141/141 [==============================] - 2s 7ms/step
Loss: 1.4955e-05
MAPE: 1.4955e-05
Accuracy: 0.9999850449829268
RMSE: 0.0038671717149774947
=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*
Test Result
37/37 [==============================] - 0s 8ms/step - loss: 1.6102e-05
37/37 [==============================] - 0s 7ms/step
Loss: 1.61017e-05
MAPE: 1.61017e-05
Accuracy: 0.9999838982941895
RMSE: 0.0040126930869964815
```

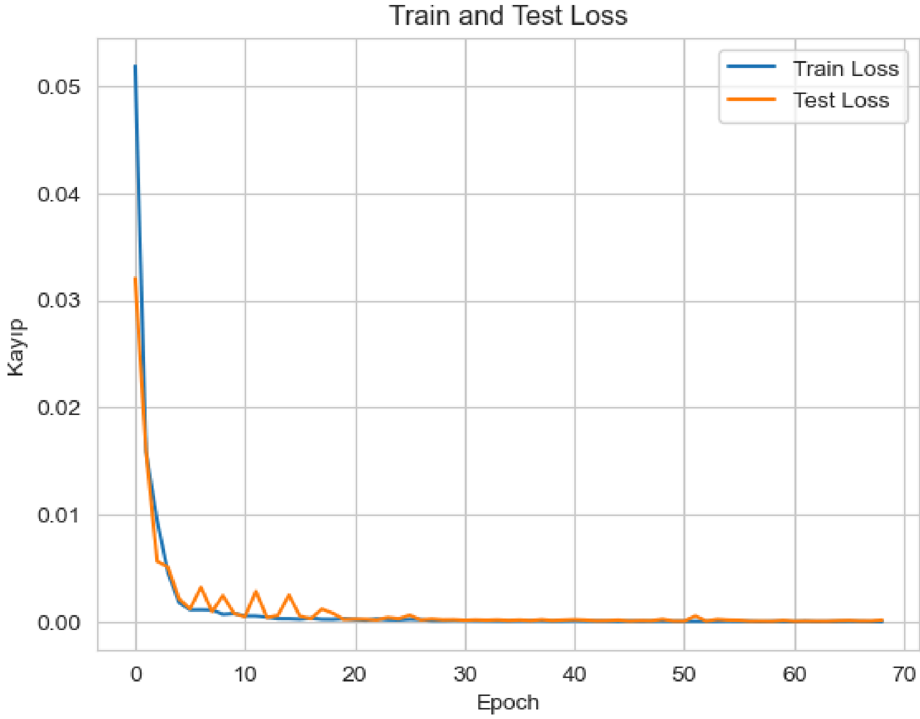
'BB_lower'




```
Train Result
141/141 [==============================] - 1s 8ms/step - loss: 4.5300e-05
141/141 [==============================] - 1s 6ms/step
Loss: 4.52996e-05
MAPE: 4.52996e-05
Accuracy: 0.9999547004161556
RMSE: 0.006730496552587309
=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*
Test Result
37/37 [==============================] - 0s 7ms/step - loss: 5.9874e-05
37/37 [==============================] - 0s 6ms/step
Loss: 5.98736e-05
MAPE: 5.98736e-05
Accuracy: 0.9999401263884475
RMSE: 0.007737804052344294
```

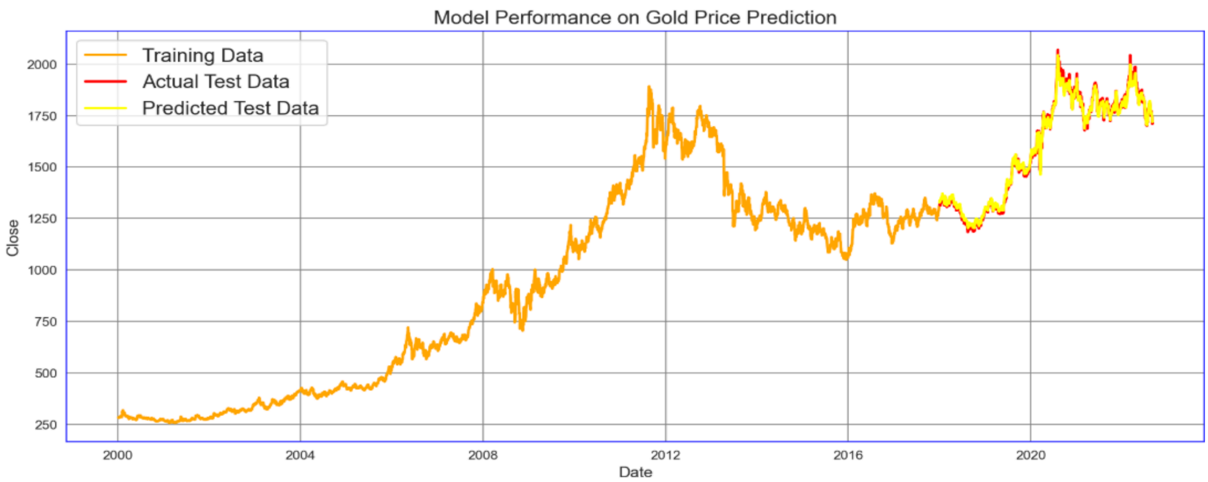
Prediction Results

First Model Result:



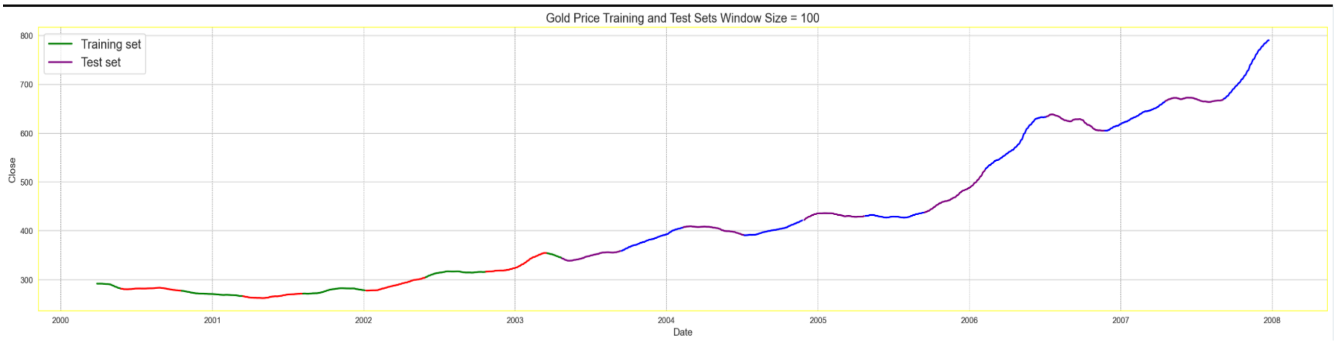

```
Train Result
142/142 [==============================] - 1s 6ms/step - loss: 4.2345e-05
142/142 [==============================] - 1s 6ms/step
Loss: 4.23455e-05
MAPE: 4.23455e-05
Accuracy: 0.9999576545262494
RMSE: 0.00650733999039276
=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*
Test Result
37/37 [==============================] - 0s 7ms/step - loss: 7.0927e-05
37/37 [==============================] - 0s 5ms/step
Loss: 7.09275e-05
MAPE: 7.09275e-05
Accuracy: 0.9999290725289296
RMSE: 0.008421844873325828
```

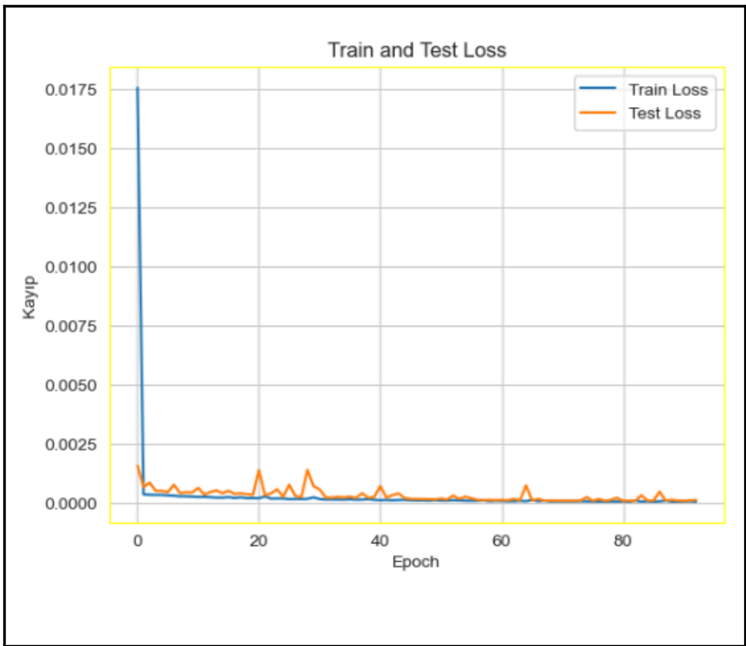
Plot Result:



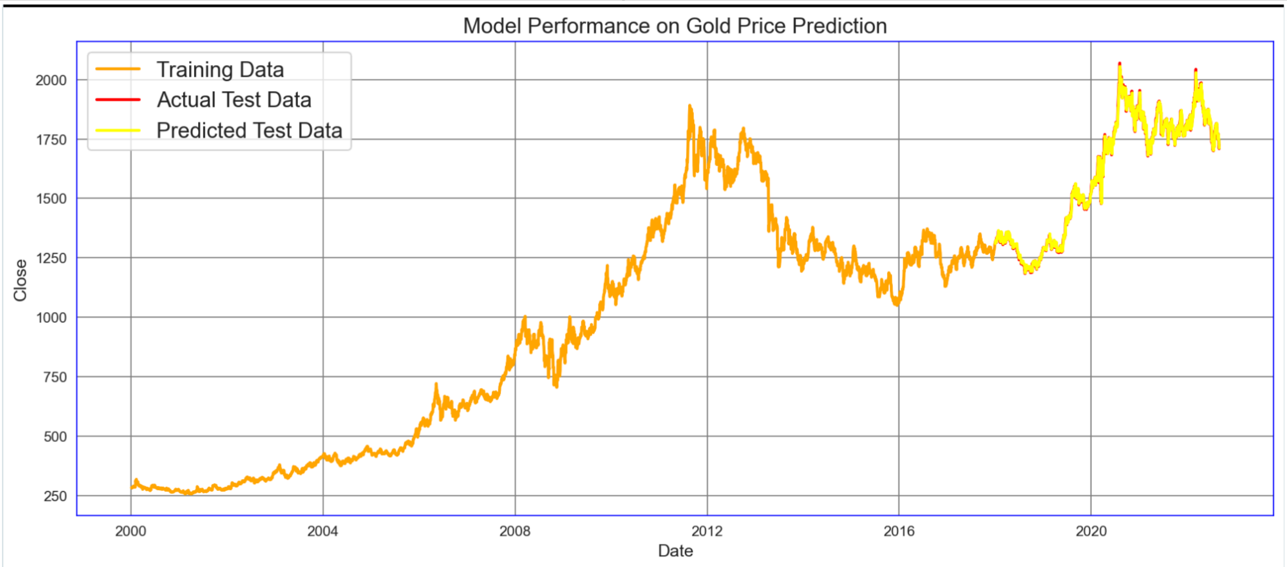
Plot shows that the result is good, predicted and test data are overlapping.

Second Model Result:
Selected sliding window size, show coloring on data between 0 to 2000 range:





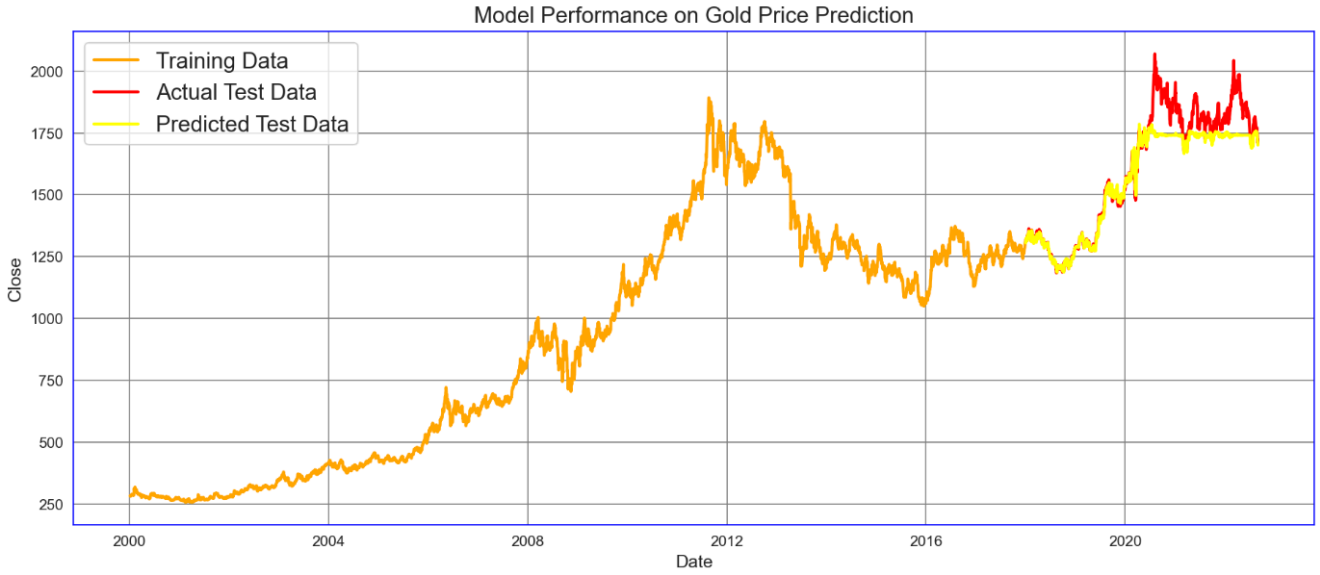

```
Train Result
139/139 [==============================] - 2s 14ms/step - loss: 4.7208e-0
139/139 [==============================] - 3s 17ms/step
Loss: 4.7208e-05
MAPE: 4.7208e-05
Accuracy: 0.9999527920152294
RMSE: 0.006870806704505645
=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=*=
Test Result
37/37 [==============================] - 1s 19ms/step - loss: 8.2586e-05
37/37 [==============================] - 1s 18ms/step
Loss: 8.25863e-05
MAPE: 8.25863e-05
Accuracy: 0.9999174137068665
RMSE: 0.009087700101427013
```



Plot shows that the result is good, predicted and test data are overlapping.

Third Model Result:

```
test size: 1177
[LightGBM] [Info] Auto-choosing col-wise multi-threading, the overhead of testing was 0.004042 seconds.
You can set `force_col_wise=true` to remove the overhead.
[LightGBM] [Info] Total Bins 25500
[LightGBM] [Info] Number of data points in the train set: 4426, number of used features: 100
[LightGBM] [Info] Start training from score 0.359197
Train RMSE: 0.003974934320488614
Test RMSE: 0.045220138602538026
Accuracy : 0.9979551390647673
```



Plot shows that the result of LightGBM is not as good as previous results, predicted and test data are not overlapping.

Conclusion

As we can see, the first model which is LSTM and uses extracted features is the most successful model among selected models.

Zeliha Erim
Data Scientist-Software Engineer