Prediction Of Crude Oil Price Using Long-short tem Memory Network Model

I used PSO's (Pakistan State Oil) historical data which consisted the data of around past 20 years of fuel prices i.e. from February 2022 to January 2002. The dataset consisted of 7 columns for Date, Price, Opening price, Closing Price, High, %change, and total volume. The observations of the prices Ire of a single day and a total of 4921 observations.

```
In [4]: df1 = pd.read_csv('PSO Historical Data (2).csv')
        df1
```

Out[4]:

	Date	Price	Open	High	Low	Vol.	Change %
0	Feb 25, 2022	176.50	176.00	177.20	173.72	558.04K	0.85%
1	Feb 24, 2022	175.01	179.00	179.00	172.00	902.62K	-2.56%
2	Feb 23, 2022	179.60	182.45	182.45	179.20	566.24K	-0.76%
3	Feb 22, 2022	180.98	182.32	182.37	179.50	678.19K	-1.24%
4	Feb 21, 2022	183.25	183.00	185.19	182.93	819.53K	-0.01%
...	...	...	...	...	...	...	...
4916	Feb 01, 2002	35.67	33.44	35.67	33.26	42.88M	7.47%
4917	Jan 31, 2002	33.19	33.26	33.34	32.39	24.95M	0.24%
4918	Jan 30, 2002	33.11	32.73	33.49	32.55	35.28M	3.34%
4919	Jan 29, 2002	32.04	31.02	32.35	30.67	49.73M	3.99%
4920	Jan 28, 2002	30.81	29.70	30.81	29.53	44.32M	7.50%

4921 rows × 7 columns

As I wanted to predict the price only, I only used the price column (uni variate). I had to create different columns to generate more meaningful information for my model. I used the shift and Rolling mean average concepts in order to create 5 additional columns.

```
stamp = pd.DataFrame(df.Price.values)
df2 = pd.concat([stamp.shift(4),stamp.shift(3),stamp.shift(2),stamp.shift(1),stamp.shift(1).rolling(4).mean(),stamp],axis=1)
df2.columns = ['t-3','t-2','t-1','t','mean','t+1']
```

```
df3 = df2[4:].reset_index(drop=True)
```

```
df3
```

	t-3	t-2	t-1	t	mean	t+1
0	30.81	32.04	33.11	33.19	32.2875	35.67
1	32.04	33.11	33.19	35.67	33.5025	37.3
2	33.11	33.19	35.67	37.3	34.8175	36.52
3	33.19	35.67	37.3	36.52	35.6700	39.24
4	35.67	37.3	36.52	39.24	37.1825	42.18
...	...	...	...	...	...	...
4912	183.7	182.19	180.5	183.26	182.4125	183.25
4913	182.19	180.5	183.26	183.25	182.3000	180.98
4914	180.5	183.26	183.25	180.98	181.9975	179.6
4915	183.26	183.25	180.98	179.6	181.7725	175.01
4916	183.25	180.98	179.6	175.01	179.7100	176.5

I used 4500 entries for training and 400 entries for testing for my LSTM model.

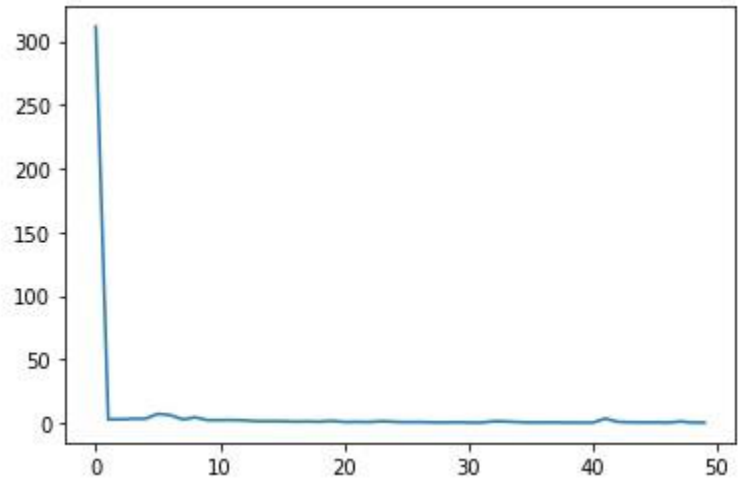
LSTM Network:

Long-short term memory networks is improved version of Recurrent Neural Network. LSTM networks have feedback connection which makes them different. This enables them to entire data sequence (such as time series) without processing points individually. It helps them to retain useful information and learn from the sequence.

The output of LSTM networks depends upon three things:

1. The current long-term memory of the network — the *cell state.*
2. The output at the previous point in time — the previous *hidden state.*
3. The input data at the current time step.

LSTM networks used series of gates in which information comes, is stored and then leaves the network.I build a sequential model using LSTM layers. I used **Adam Algorithm** as an optimizer and I evaluated my model performance or loss was determined by **Mean Squared Error (MSE).** I trained my model on 50 epochs with batch size = 5. The loss graph of model :




```
Epoch 50/50
900/900 [==============================] - 2s 2ms/step - loss: 0.1744
```

```
In [683]: model.evaluate(x_test,y_test,batch_size=5)

          80/80 [==============================] - 0s 1ms/step - loss: 0.0818
```

```
In [6]: #Enter the previous time stamp values. D(t) represents the current day

        a = float(input('Enter Day(t) price:'))
        b = float(input('Enter Day(t-1) price:'))
        c = float(input('Enter Day(t-2) price:'))
        d = float(input('Enter Day(t-3) price: '))
        e = (a+b+c+d)/4

        next_pred(a,b,c,d,e)

        Enter Day(t) price:183.25
        Enter Day(t-1) price:180.98
        Enter Day(t-2) price:179.6
        Enter Day(t-3) price: 175.01
        The prediction for tomorrow's price :  178.7664
```

I created final pipeline where the user would have to enter the prices of fuel of past 4 days and the model I built will predict the price for the next day as you can see above.