STOCK PRICE TREND PREDICTION WITH LSTM

Project By : Prachi Narayan Padyal

Introduction:

In the dynamic world of financial markets, predicting stock price trends has always been a topic of significant interest for investors, analysts and researchers. Predicting stock trends is a complex task due to volatile and non-linear nature of financial markets. Traditional models often fail to capture such patterns but deep learning techniques like Long Short- Term Memory(LSTM) network have been shown strong performance in time series forecasting. This project uses an LSTM based model to predict stock price trends using historical data including features like closing prices, opening prices, volume.

Abstract:

Stock price prediction is a challenging yet crucial task in the financial domain due to inherent volatility and complexity of market behavior. This project explores the application of LSTM network for forecasting stock price trends using historical time series data. The model is trained on features such as closing prices, opening prices, high, low, volume. The approach involves data preprocessing, model development, evaluation and visualization of predicted trends. The results demonstrate that LSTM models can effectively learn from past stock data to anticipate future trends offering valuable insights for traders and investors.

Tools like Python (Jupyter Notebook), Machine Learning, Deep Learning(LSTM) are used.

Data Analysis:

- Import necessary libraries and load dataset.
- In data preprocessing, drop null values and by using feature variables closing prices, opening prices, volume, adj close find MA20, MA50 and RSI values.

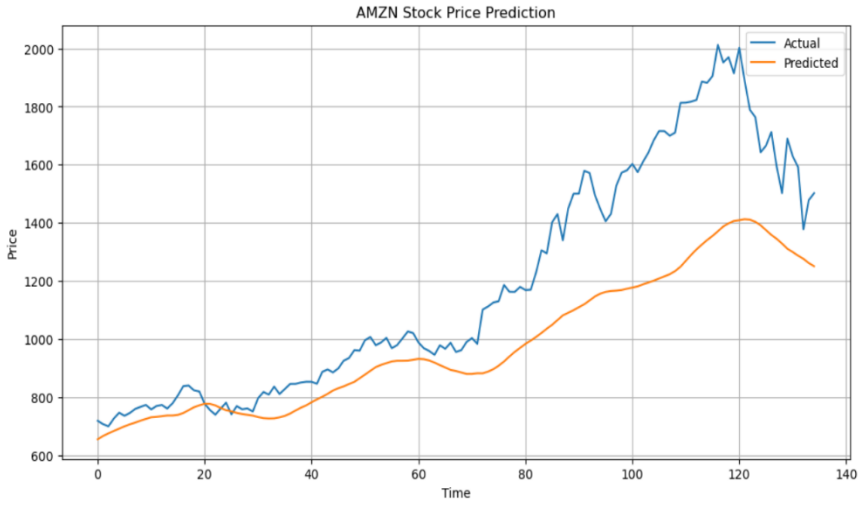
[6]:
```
# Moving averages
data['MA20'] = data['Close'].rolling(window=20).mean()
data['MA50'] = data['Close'].rolling(window=50).mean()

# RSI calculation
delta = data['Close'].diff()
gain = np.where(delta > 0, delta, 0)
loss = np.where(delta < 0, -delta, 0)
avg_gain = pd.Series(gain).rolling(window=14).mean()
avg_loss = pd.Series(loss).rolling(window=14).mean()
rs = avg_gain / avg_loss
data['RSI'] = 100 - (100 / (1 + rs))

data.dropna(inplace=True)
data.tail()
```

[6]:

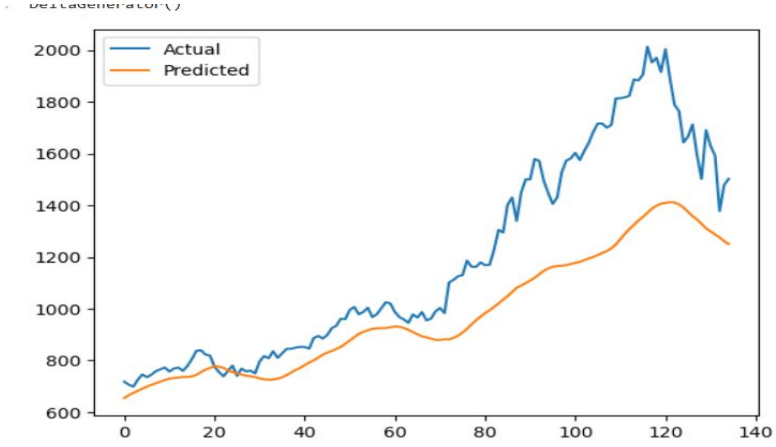
	Date	Open	High	Low	Close	Adj Close	Volume	MA20	MA50	RSI
779	2018-12-03	1769.459961	1778.339966	1609.849976	1629.130005	1629.130005	31922300	1802.264014	1637.130405	32.741213
780	2018-12-10	1623.839966	1704.989990	1585.000000	1591.910034	1591.910034	31976000	1790.996014	1645.579207	33.446097
781	2018-12-17	1566.000000	1584.530029	1363.959961	1377.449951	1377.449951	47777100	1768.704010	1648.545405	26.921459
782	2018-12-24	1346.000000	1513.469971	1307.000000	1478.020020	1478.020020	36182800	1748.290009	1652.001807	33.566493
783	2018-12-31	1510.800049	1520.760010	1487.000000	1501.969971	1501.969971	6954500	1729.277509	1656.149607	30.204736

- Variables like Close, MA20, MA50 and RSI are scaled by using MinMax scalar. Then dataset is split into training and testing data.
- By using tensorflow library, Train and validate LSTM model. Then actual and predicted price graph is plotted.

➢ Long Short Term Memory (LSTM) is recurrent neural network. LSTMs remember past values to make better future predictions.



- Plot LSTM plot of actual and predicted price



Conclusion:

Actual stock prices are higher than predicted stock prices by LSTM model. Actual stock price is at high price i.e. 2000 for time period 120 whereas predicted stock price by LSTM model is 1400. i.e. prices predicted by LSTM model is lesser than actual prices. This suggests that model may be overly smooth or not responsive enough to sudden market shifts.