An attempt to predict the next closing price per share of a listed company

Given the bombastic title, from the onset, I must state that reliably predicting the next price per share of a listed company is not possible because the participants in the market have the ability to change it.

So, in simpler words, let's say I find the magic model that predicts the market. Well, by simply employing it, I will change the market sufficiently to render the model obsolete.

I do this as an exercise to improve my skills, and to have an idea where the share price of a company is heading.

In this project I import and preprocess the data, feed it into a model and try to find the best parameters for the said model.

Importing the data
The share price is being imported using the yfinance python package. After importing the data, only the close prices are kept.

Preprocessing the data
In preprocessing the data there are two stages. Extracting the changes and preparing the arrays for the model.

Given the big differences in the price, a helpful variable is what the change from the previous day is. In order to obtain this change, a percentage difference is being calculated. This difference is afterwards converted on a log scale.

The closing prices are being fit using a MinMax Transformer, meaning that if the prices of a share are between 20 and 90, a price of 20 will receive a value of 0 and a price of 90 a value of 1, everything in between will have a value between 0 and 1.

Preparing the array for the model means building two arrays, one in which an item is given by the n entries which represent the closing and the log change for the previous n days, and the other which represents the target price for the n+1th day. This way the arrays fed into the model have a lookout of n back days attempting to predict the nth + 1 day.

The model

A Sequential LTSM model is used. With 1 layer, a batch size of 16, the optimizer being “ADAM”. The number of epochs is a parameter that can be refined.

Finding best parameters

A simple loop in a loop system is being used to find the best parameters for the model. The parameters involved are the look back period, the lookback days and the number of epochs in the model.

The variable that needs to be minimised is the mean squared difference between the predicted and actual values divided by the last actual price times 100 (TestScore). Using this indicator, it is possible to assess the accuracy of the model as the percentage of the last actual price.

Results

When testing, running the model on the MSFT the TestScore is 3.477, meaning that if implemented in real time scenarios the predicted price will differ, theoretically, on average by 3.477 percent.

The sole testing company throughout the project was Microsoft. For Microsoft the results are satisfactory. For Apple or Pfizer for instance, the results are not satisfactory. The list of parameters used to find the lowest TestScore needs to be extended. This list currently employs periods of 1, 2 and 5 years, lookback days of 3, 13, and 30, and the number of epochs of 50 and 100. In this exercise the list was not extended as it requires great computational power.

Further Development

As stated above the list of the parameters needs to be extended. Also, when the computational power is available, the parameters can be extended to include for instance 150 epochs, or 150 lockback days. This however, necessitates time and computational power.

Do not attempt this at home

In this article I described a way to predict the next closing price. Please heed the warning at the beginning. You might lose a lot of money attempting to implement this model.

Also, please bear in mind this is not investment advice.