Stock Price Prediction Using CNN

Fateme Rahimi
Department of Mathematics
Shahid Beheshti University
fateme.rahimi1@sbu.ac.ir

Abstract

Predicting how the stock market will perform is one of the is a very hot topic in our life. in recent years more and more people are devoted to the study of the prediction and it becomes easier and easier for us to make stock prediction by using different ways now, including machine learning, deep learning and so on. here, we examined various machine learning methods for predicting stock prices, and finally proposed a deep learning method based on the Convolutional neural network and LSTM, which predicts stock closing price in the Tehran stock market. We set the closing price and date for some years as input and predict closing price in future, then run and test the program.

1 Introduction

There are many machine learning methods that have been successfully applied in various interesting financial problems, such as trend approximation, bankruptcy prediction, investment targets, credit evaluation, and portfolio selection. here, we will use a deep learning framework to analyze the trading time-series data. stock trading analysis is divided into two parts – Fundamental Analysis and Technical Analysis.

- Fundamental Analysis involves analyzing the company's future profitability on the basis of its current business environment and financial performance.
- Technical Analysis, on the other hand, includes reading the charts and using statistical figures to identify the trends in the stock market.

Our focus will be on the technical analysis part.the profit or loss calculation is usually determined by the closing price of a stock for the day, hence we will consider the closing price as the target variable.in this project, first we use machine learning methods to predict stock closing prices and review the results, then we use deep learning methods to make this prediction. this study has been done on data of the metal ores symbol in the Tehran stock market and it has been implemented with Python ,also deep learning models with Pytorch, you can find project code in the github repository[1].

2 Dataset

We'll be using a dataset from Tehran stocks[2]. there are multiple variables in the dataset:'date' 'open', 'high', 'low', 'adjClose', 'value', 'volume', 'count', 'close'.in the next table you can see some records of this database: we will work with below columns:

- close price: The close represent the final price at which the stock is traded on a particular day.

[1] https://github.com/ftmhrahimi/Deep$_L earning_F 20_P roject$

[2] https://pypi.org/project/pytse-client/

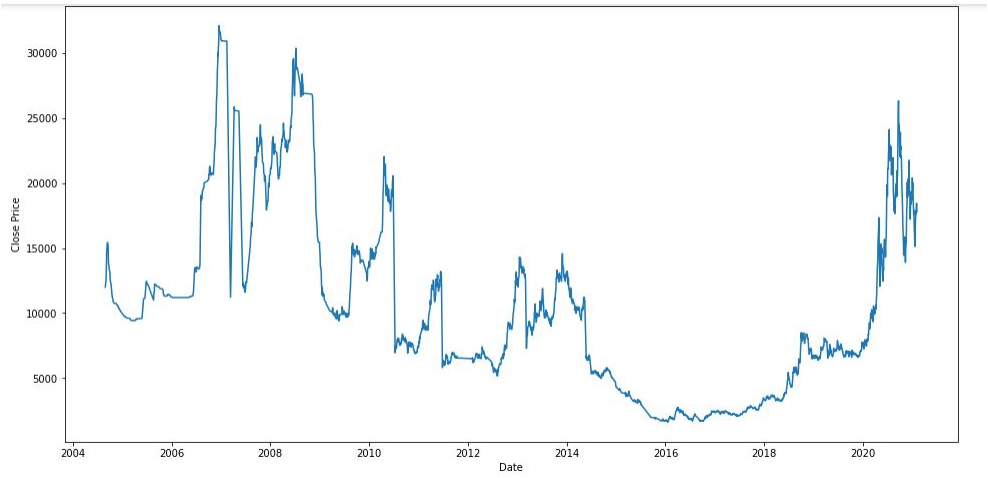
34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.

date	open	high	low	adjClose	value	volume	count	close
2004-08-29	12000.0	12021.0	12000.0	12000.0	18841605000	1570000	2708	12000.0
2004-09-04	12600.0	12600.0	12600.0	12600.0	12600000000	1000000	849	12600.0
2004-09-05	13230.0	13230.0	7115.0	13230.0	34449329770	2708823	3887	13230.0
2004-09-07	13891.0	13891.0	13891.0	13891.0	41395180000	2980000	996	13891.0
2004-09-08	14585.0	14585.0	14585.0	14585.0	20305659965	1392229	409	14585.0
...	...	...	...	...	...	...	...	...
2021-01-27	17500.0	17940.0	17010.0	17580.0	132698927140	7549349	1498	17890.0
2021-01-30	17800.0	18450.0	17600.0	18250.0	152027033200	8328164	2054	18450.0
2021-01-31	18630.0	18630.0	17650.0	18070.0	178086920360	9853128	2060	18440.0
2021-02-01	18240.0	18380.0	17540.0	17910.0	222684584670	12435398	1483	18100.0
2021-02-02	18160.0	18160.0	17110.0	17610.0	133482246650	7581492	1446	17790.0

- date: this data set contains information from 2004/08/29 to 2021/02/02. note that the market is closed on weekends and public holidays.

2.1 Close price

As we said, the calculation of profit or loss is usually determined by the closing price of the stock for the day, so we considered it as a target variable. First, we plot this target variable to understand how it is formed in our data:



As can be seen from the plot, the closing price of these stocks fluctuated slightly between 2014 and 2016, but in other years, a lot of fluctuations can be seen in it.

3 Machine Learning Methods

In this section, we will look at different machine learning techniques for predicting daily stock closing prices.

3.1 Moving Average

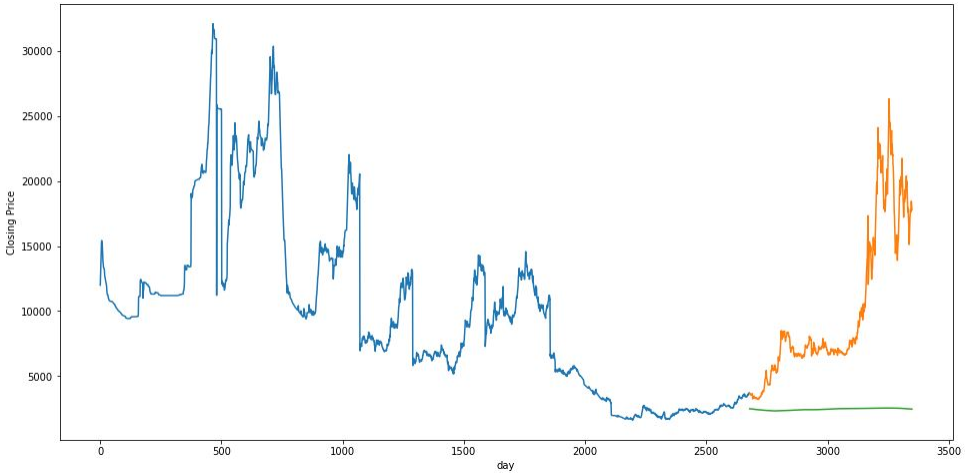
In the moving average, the predicted closing price for each day will be the average of a set of previously observed values. Instead of using the simple average, we will be using the moving average technique which uses the latest set of values for each prediction. In other words, for each subsequent

step, the predicted values are taken into consideration while removing the oldest observed value from the set.

First we split the data into validation and training sets, we put 80 percentage of the data in the training set and the rest in the validation set, and with moving average predict closing price in validation set. To check the accuracy of the model, we obtain the RMSE value:

RMSE value on validation set:
9421.57997563058

Review alone does not help us understand model performance, so we visualize this to get a more intuitive understanding. here is a plot of the predicted values along with the actual values:



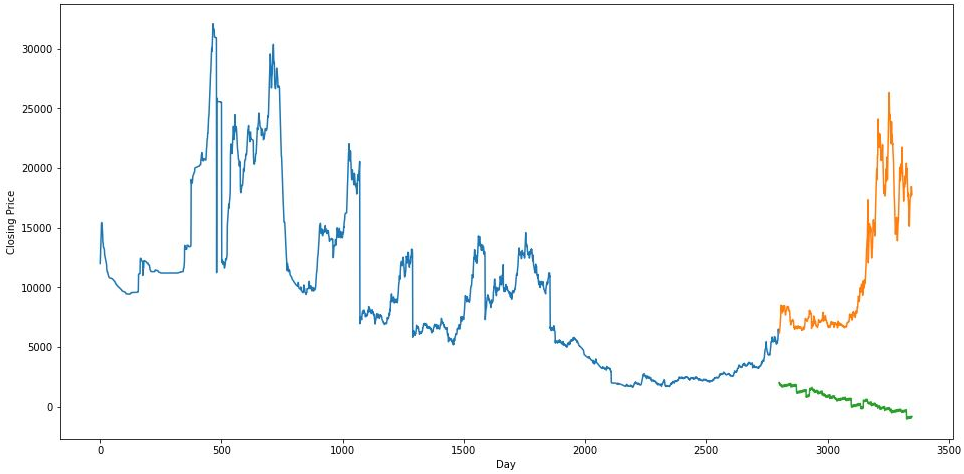
As shown in the figure using this method, the predicted closing price is very different from the actual values.

3.2 Linear Regression

The linear regression model returns an equation that determines the relationship between the independent variables, in our problem we do not have a set of independent variables. We have only the dates instead. we use the date column to extract features like – day, month, year etc. and then fit a linear regression model.
As before, for validation, we split data into validation and training set, and calculate RMSE:

RMSE value on validation set:
12311.415187756978

The RMSE value is higher than the previous technique, which clearly shows that linear regression has performed poorly. Let's look at the plot and understand why linear regression has not done well:

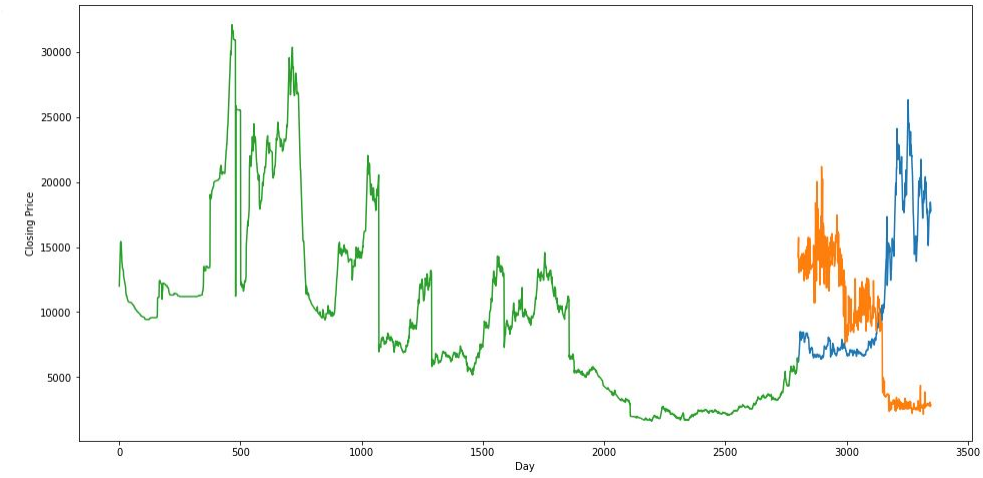


3.3 kNN

Another ML algorithm we use here is kNN (k nearest neighbours), that based on the independent variables. kNN finds the similarity between new data points and old data points.
We use the same training and validation set from the last section, and because of this method find the similarity of points based on distance, we first scale data and set them in 0 , 1 interval.
As before, for validation, we split data into validation and training set, and calculate RMSE:

RMSE value on validation set:
10449.254642820486

There is not a huge difference in the RMSE value, but a plot for the predicted and actual values should provide a more clear understanding:



4 Deep Learning Methods

In this section we try another advanced techniques in the deep learning.

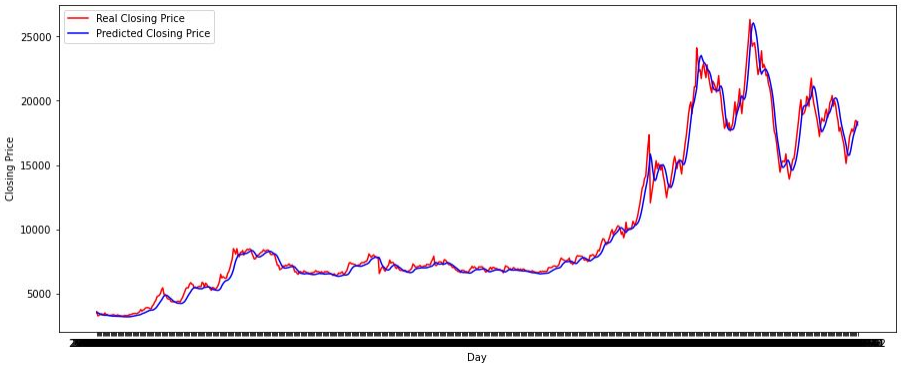
4.1 Long Short Term Memory (LSTM)

Here we use LSTM, that is widely used for sequence prediction problems and have proven to be extremely effective. The reason they work so well is because LSTM is able to store past information that is important, and forget the information that is not.

To use this method, we use only the closing price column and set the data of this column to the interval 1 and -1. LSTMs expect our data to be in a specific format, usually a 3D array. We start by creating data in 60 timesteps and converting it into an array using NumPy. we create an LSTM model with one LSTM layer of 32 neurons and one linear layer as readout layer. as before, for validation, we split data into validation and training set, and calculate RMSE:

RMSE value on validation set:
677.86

Comparing the value of RMSE with other models, it is clear that this method has worked better than others in predicting price. This is clear from the figure:

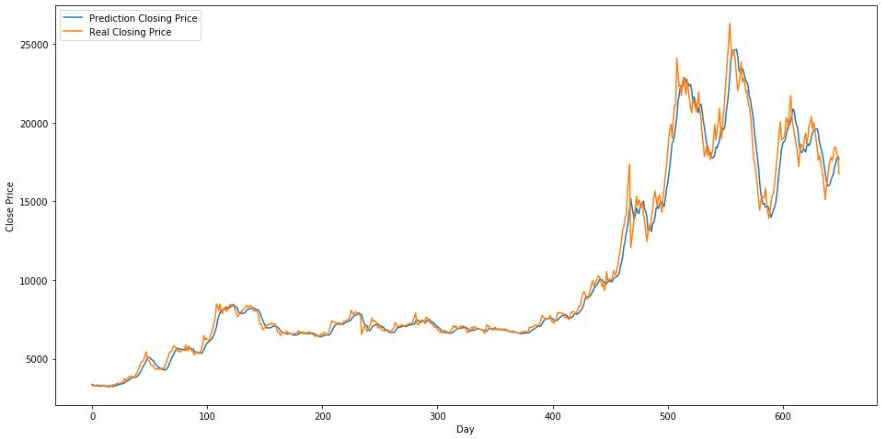


4.2 CNN

Convolutional Neural Network models, or CNNs for short, can be applied to time series forecasting.although traditionally developed for two-dimensional image data, CNNs can be used to model univariate time series forecasting problems.here we want to use Univariate CNN Models, Univariate time series are datasets comprised of a single series of observations with a temporal ordering and a model is required to learn from the series of past observations to predict the next value in the sequence.the CNN model will learn a function that maps a sequence of past observations as input to an output observation. As such, the sequence of observations must be transformed into multiple examples from which the model can learn.we can divide the sequence into multiple input/output patterns called samples, here 20 time steps are used as input and one time step is used as output for the one-step prediction that is being learned.
In our model we are using one 1D convolution layer with 64 filter maps and a kernel size of 2,ReLu acitivation function and linear layers.The model is fit using the efficient Adam version of stochastic gradient descent and optimized using the mean squared error, or ‘mse‘, loss function. finally, we calculated the RMSE on the test set:

RMSE value on validation set:
283.6042175292969

In the following plot, you can see the prediction of this model:

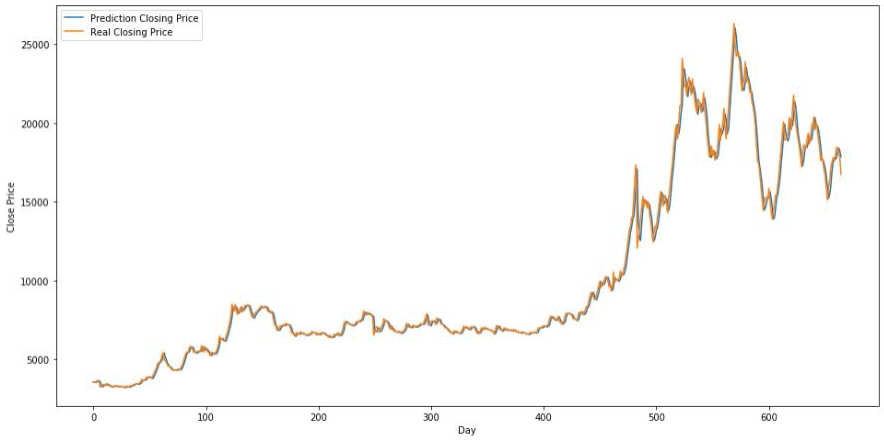


4.2.1 Tuning CNN parameters with Optuna

We set 5 as time step, and with Optuna[3] Tune hyperparameter like number of kernel layer and learning rate. finally we calculate RMSE on test set:

RMSE value on validation set:
182.49559020996094

As can be seen from the diagram, the performance of the model is improved by tuning the hyperparameters:



5 Conclusion

As we have seen, machine learning methods did not perform well for time series prediction, and deep learning methods performed better for time series prediction. Note that we have tune the parameters of the CNN Model here, and the LSTM can do better by setting the parameters. But in this study with CNN method with the introduced parameters has the lowest RMSE and the best performance.

References

[1] Jou-Fan Chen & Wei-Lun Chen & Chun-Ping Huang (2016)Financial Time-series Data Analysis using Deep Convolutional Neural Networks, *2016 7th International Conference on Cloud Computing and Big Data*
[2] https://machinelearningmastery.com/how-to-develop-convolutional-neural-network-models-for-time-series-forecasting/

[3]https://optuna.org/