proposal

April 11, 2022

1 Domain Background

Investment firms, hedge funds and even individuals have been using financial models to better understand market behavior and make profitable investments and trades. A wealth of information is available in the form of historical stock prices and company performance data, suitable for machine learning algorithms to process. Accurate prediction of stock market returns is a very challenging task due to volatile and non-linear nature of the financial stock markets. Since a lot of investors use AI models to work with stocks it's become very important to predict price for making decisison about future investment or porfolio rebalancing. For those who use Reinforcement Learning it's also quite important since agent should know when to hold or sell/buy new assets to expect better reword for the action in future.

Academic research prove that stock closing price can be predicted with ML algoritm. According to this research prediction is quite accurate for some stocks. Please check links of this research below. Even though that alogorithms poove to be working we still should keep in mind, that they are undable to predict future and sometime unexpected events might happend. For example natural disasters, issuew in supply chain, etc. can lead to stock price go down unexpectedly. Hence it's good to use different sources to take a good decision, because ML algorithm in most cases can find patterns in stock behavior rather than predict real stock price.

Common algorithms used to predict stock:

- RNN
- LSTM
- Random Forest
- Transformers

1.0.1 Reserches

- Stock Closing Price Prediction using Machine Learning Techniques
- A Data Organization Method for LSTM and Transformer When Predicting Chinese Banking Stock Prices

2 Problem Statement

Build stock price predictor which will use historical data to predict stock price for given date. It's quite clear for me that it's impossible to predict future sock movement, becouse it depends on a lot of different factors. For example:

- Social activity

- Company performance
- Natural desisters
- Etc.

Some of this factors can be predicted from experience but some of them still hard to predict, like natural desister. Hence the main goal of this project try to build predictor which try to find patterns in the past stock movement and help with decision making process together with some other information.

Ideally there should be a lot of such type of predictors whic use different types of informations, for example market news or satellite images, etc. Which combined in ensamble model for better prediction, but this require a lot of time and efforts to be developed. Hence in this project we try to reduce scope and build small part of bigger system and did small research if it worth to make this type of predictions at all.

3 Solution Statement

Here we have regression problem since we try to predict stock price. According to different researches such ML architectures as RNN and Transformers perform better than for example Random Forest (XGBoost). It's looks logical since stock price is time series and RNN and Transformers works good with this type of data. For this type of porject Transformers will be used since RNN networks has some following issues comparing to Transformers architectures.

RNN arcitecture issues: - parallel computing is difficult to implement; - for long sequences of data there is loss of information; - problem with vaninshing gradient.

To test the idea 2 random stocks from well known companies (for example: Apple, Google, Facebook etc.) which it part of S&P 500 index will be selected.

To solve this problem different techical inicators will be used.

Technical indicators:

- Simple moving average (SMA)
- Exponential moving average (EMA)
- Moving Average Convergence Divergence (MACD)
- Bollinger band
- Momentum
- Trading volume

3.0.1 Technical inidcator calculation

EMA and MA will be calculated for 3, 8 and 21 days

Simple moving average

$$SMA = \frac{A_1 + A_2 + ... + A_n}{n} \tag{1}$$

where: (2)

A = Average in period n (3)

n = Number of time periods (4)

(5)

Exponential moving average

$$EMA_t = \left[V_t \times \left(\frac{s}{1+d}\right)\right] + EMA_y \times \left[1 - \left(\frac{s}{1+d}\right)\right] \tag{6}$$

where: (7)

EMA_t = EMA today (8)

V_t = Value today (9)

EMA_y = EMA yesterday (10)

s = Smoothing (11)

d = Number of days (12)

(13)

Moving Average Convergence Divergence

$$\text{BOLU} = \text{MA}(\text{TP}, n) + m * \sigma[\text{TP}, n] \tag{14}$$

$$\text{BOLD} = \text{MA}(\text{TP}, n) - m * \sigma[\text{TP}, n] \tag{15}$$

where: (16)

BOLU = Upper Bollinger Band (17)

BOLD = Lower Bollinger Band (18)

MA = Moving average (19)

TP (typical price) = (High + Low + Close) $\div$ 3 (20)

n = Number of days in smoothing period (21)

m = Number of standard deviations (22)

$\sigma[\text{TP}, n]$ = Standard Deviation over last n periods of TP (23)

(24)

Momentum

$$\text{Momentum} = V - Vx \tag{25}$$

where: (26)

V = Latest price (27)

Vx = Closing price (28)

x = Number of days ago (29)

4 Datasets and Inputs

To train model historical data for last 5 years will be used if it's available.

There are several open sources for historical stock price data which you are free to use:

- Yahoo Finance API Specification
- Polygon Financial Market Data Platform
- TradingView
- Quandl.

Actual data will be loaded from Yahoo Finance. And has following structure:

- Open
- High
- Low
- Close
- Adj Close
- Volume
- company_name
- Date

For feature engineering following data will be used: - Adjusted Close - Simple moving average (SMA) - Exponential moving average (EMA) - Moving Average Convergence Divergence (MACD) - Bollinger band - Momentum - Trading volume - Price date

Price date will be splitted further to month, year, day of month, day of year.

Data will be separated to 3 sets: - train - validation - test

Model will be deployed as SageMaker endpoint.

5 Benchmark Model

To measure model performance mean squared error (MSE) will be used, better model should have lower error. When best model selected it will be tested against real stock data with measuring error. To check how well it perform we have to check result against similar models, for that reason similar researches was found. Please check links below.

According to read researches it's possible to acheive quite good accuracy on some markets. For this project try to acheived MSE <= 0.002 for markets with low volatility since markets with high volatility more risky and difficult to predic.

5.0.1 Research articles

- Stock Price Prediction Mobile Application
- Stock Market Analysis + Prediction using LSTM
- Stock predictions with state-of-the-art Transformer and Time Embeddings

6 Evaluation Metrics

As evaluation metric Mean Squared Error will be used.

7 Project Design

SageMaker will be used this project. For model development on of the SageMaker buil-in algorithms will be used.

Project stages:

- Feature engenering
- Training model
- Deploy endpoint