STOCK PRICE PREDICTION

Thanit Tativannarat, Phathit Sriburi, Nattanat Chatthee, Natthanon Tunyanun

Department of Computer Engineering
Chulalongkorn Univerisy
Bangkok, Thailand

ABSTRACT

Predicting a stock price is a very challenge task.It requires high accuracy go together with high flexibility.Nowadays, there are many people working about Algorithmic trading.May be this is the new trend or it can be a new job that appeared in the future.

1. INTRODUCTION

1.1. Motivation

Nowaday, we can see many of securities company provide a software that help customers to plan their investment strategy. the software is also known as EA (Expert Advisor) ,this brings our team curiosity.Does the EA really work? Our target is to analyze the models and to find out which model is the most practical in the stock market.

1.2. Previous Work

There are many different works about predicting stock price. Some of them use linear regression model,Long Short Term Memory (LSTM),Gated Recurrent Unit (GRU),Recurrent Neural Network (RNN) to deal with time series data.Some of them use Convolutional Neural Network (CNN) to find the chart pattern in the stock chart.

1.3. What We Are Going to Do

Before our experiments, we tried some basic Deep Neural Network that base on dataset without feature engineering and found that the test score is not good as we expected.After that, our team tried to add some useful features such as moving average to the dataset but, the test score of our new model stills not good as we expected.We found that we did not deal with noises in the dataset.The noise can be generated by trading psychology or news.These leads to focusing on the noises.After that,we create the ARIMA model that relate with time-series data.The result of the model still like the others.
We know that Newyork stock market (Dowjones) opened before Japan stock market (Nikkei) and these two stock markets are not overlapse each other.Our next milestone is using the close price of Dowjones stock market to predict the close price of Nikkei stock market.

1.4. Organization of the Paper

In Section Necessary Background we provide the background on the investment, basic time-series analysis, Machine Learning and Backtesting.

2. NECESSARY BACKGROUND

We divide topics in this part into 6 parts are as follows :
1. Knowledge about investment
2. What is time-series
3. ARIMA model
4. Feature engineering
5. Training a model
6. Backtesting

2.1. Investment

Everyday,there are many investors trade stocks with each other.The buyers have the money but, they would like to hold stocks instead of money , on the other hand , the sellers already have stocks but, they would like to sell their stocks and keep money instead.The market is the place that allow investors to trade with each other.Trading transaction will be happened when buyer and seller make a deal. Price mechanism is also created by this event too.

2.2. Basic time-series Analysis

- What is time-series?
  A time series is a sequence of data that ordered by time.Stock price at time N will effect to stock price at time N+1

- Stocks data is also a time series that has seasonality (each stock has opportunity day itself).

2.3. ARIMA Model

ARIMA, short for ”Auto Regressive Integrated Moving Average”, is one of the machine learning model for manipulate with time series data.The ARIMA model is divided into 3 terms :
p is the order of AR term
q is the order of MA term
d is the number of differencing required to make the time series stationary.
This model is very flexible and has strong underlying theory.Main concepts of this model are about order and differencing of the dataset.

2.4. Feature engineering

Feature engineering is the method that add some useful data to the dataset.This method is created for improving the model.

2.5. Model training

Model training is how to make a model learns from data that feeded into it.The model will improve itself by tuning weights and biases to minimize loss.

2.6. Backtesting

Backtesting is the process that use the model to predict the output then, display or find errors between the output and the test data.This process is for testing how much flexibility does the model have.

3. YOUR PROPOSED METHOD

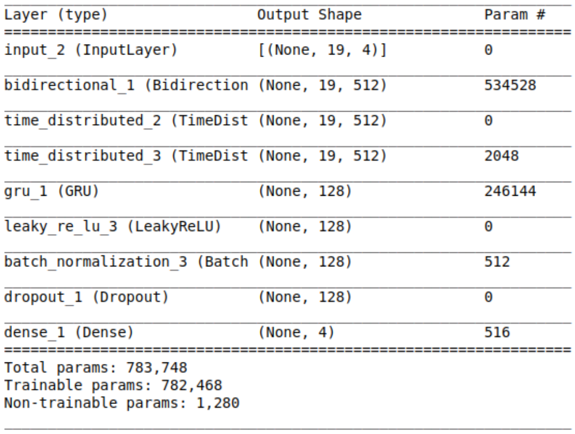
This section is about feature engineering and implementation’s details.

3.1. Dataset

First, download these two dataset from link below.All dataset which is used in our experiment are from Yahoo Finance.

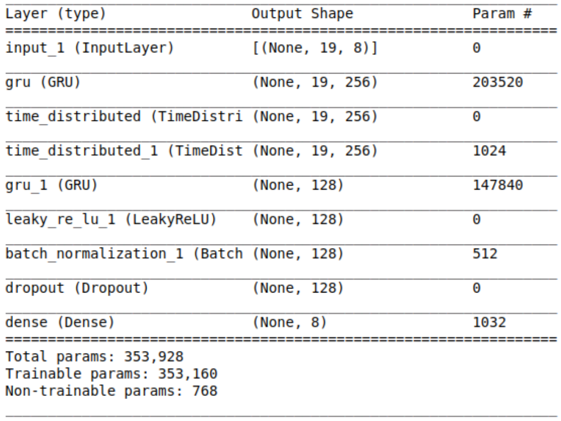
- Dowjones dataset
  `https://finance.yahoo.com/quote/%5EDJI/history?p=%5EDJI`
  Enter Time Period as you like, Choose Frequency to Daily then Press Download Data button.
- Nikkei dataset
  `https://finance.yahoo.com/quote/%5EN225/history?p=%5EN225`
  Enter Time Period as you like, Choose Frequency to Daily then Press Download Data button.

When your download is finished, you would receive a CSV file that contains basic market data (open price, high, low, close price and volume).



```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_2 (InputLayer)         [(None, 19, 4)]           0
_________________________________________________________________
bidirectional_1 (Bidirection (None, 19, 512)           534528
_________________________________________________________________
time_distributed_2 (TimeDist (None, 19, 512)           0
_________________________________________________________________
time_distributed_3 (TimeDist (None, 19, 512)           2048
_________________________________________________________________
gru_1 (GRU)                  (None, 128)               246144
_________________________________________________________________
leaky_re_lu_3 (LeakyReLU)    (None, 128)               0
_________________________________________________________________
batch_normalization_3 (Batch (None, 128)               512
_________________________________________________________________
dropout_1 (Dropout)          (None, 128)               0
_________________________________________________________________
dense_1 (Dense)              (None, 4)                 516
=================================================================
Total params: 783,748
Trainable params: 782,468
Non-trainable params: 1,280
_________________________________________________________________
```

Fig. 1. nikkei without feature engineering



```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_1 (InputLayer)         [(None, 19, 8)]           0
_________________________________________________________________
gru (GRU)                    (None, 19, 256)           203520
_________________________________________________________________
time_distributed (TimeDistri (None, 19, 256)           0
_________________________________________________________________
time_distributed_1 (TimeDist (None, 19, 256)           1024
_________________________________________________________________
gru_1 (GRU)                  (None, 128)               147840
_________________________________________________________________
leaky_re_lu_1 (LeakyReLU)    (None, 128)               0
_________________________________________________________________
batch_normalization_1 (Batch (None, 128)               512
_________________________________________________________________
dropout (Dropout)            (None, 128)               0
_________________________________________________________________
dense (Dense)                (None, 8)                 1032
=================================================================
Total params: 353,928
Trainable params: 353,160
Non-trainable params: 768
_________________________________________________________________
```

Fig. 2. nikkei combine with dowjones V1

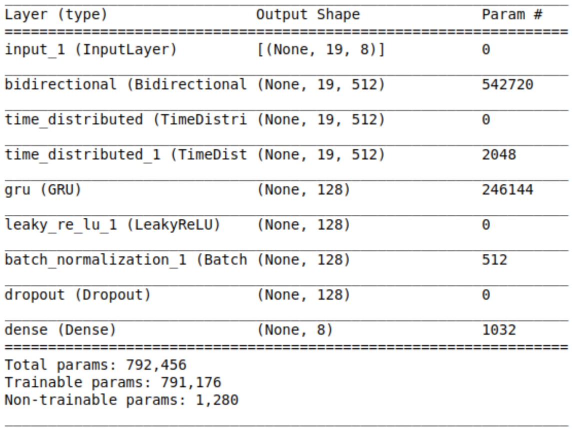
3.2. Feature engineering

After the previous part,Our team join two CSV files together by appending Nikkei dataset at day N to Dowjones dataset at day N-1 for nikkei combine with dowjones V1 model and nikkei combine with dowjones V2 model.

3.3. Design of the model

We created three models that have different layers design.

1. nikkei without feature engineering
2. nikkei combine with dowjones V1
3. nikkei combine with dowjones V2 (Bidirectional)



```
Layer (type)                 Output Shape              Param #
=================================================================
input_1 (InputLayer)         [(None, 19, 8)]           0

bidirectional (Bidirectional (None, 19, 512)           542720

time_distributed (TimeDistri (None, 19, 512)           0

time_distributed_1 (TimeDist (None, 19, 512)           2048

gru (GRU)                    (None, 128)               246144

leaky_re_lu_1 (LeakyReLU)    (None, 128)               0

batch_normalization_1 (Batch (None, 128)               512

dropout (Dropout)            (None, 128)               0

dense (Dense)                (None, 8)                 1032
=================================================================
Total params: 792,456
Trainable params: 791,176
Non-trainable params: 1,280
_________________________________________________________________
```

Fig. 3. nikkei combine with dowjones V2

3.4. Model training

First, we use factor=0.8, patience=4, min lr=0.0001, batch size=64,epochs=300 and optimizer Adam(0.005) for nikkei without feature engineering model.
Second, we use factor=0.9, patience=3, min lr=1e-9, batch size=1000, epochs=500 and Adam(lr=1e-2, decay=1e-2) for nikkei combine with dowjones V1 model.
Last model, we use factor=0.9, patience=3, min lr=1e-9, batch size=1000, epochs=500 and Adam(lr=1e-5, decay=1e-4) for nikkei combine with dowjones V2 model.

4. EXPERIMENTAL RESULTS

Testing method for time series data has two ways

1. Expanded window

This method is used for testing the data in the past.

2. Sliding window

This method is used for tesing new data.

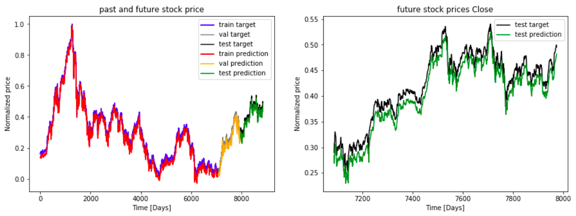


Fig. 4. Backtesting nikkei without feature engineering model by sliding window and expanded window

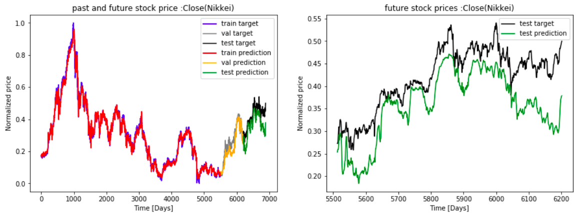


Fig. 5. Backtesting nikkei combine with dowjones V1 model by sliding window and expanded window

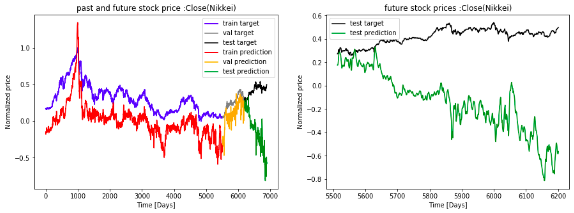


Fig. 6. Backtesting nikkei combine with dowjones V2 model by sliding window and expanded window

Models	train	test	validate
nikkei without feature engineering	0.00064	0.00049	0.00068
nikkei combine with dowjones V1	0.00030	0.07512	0.01022
nikkei combine with dowjones V2	0.08058	12.39508	0.78669

Table 1. Loss of the model

5. CONCLUSIONS

Every markets has it own flactuation.When we try to combine the data of both markets and the data is not denoised before model training, the created model is worse than the nikkei without feature engineering model.So, the nikkei without feature engineering model has the best result.In the real world, EA can be created by using machine learning model or rule-based programming but now, rule-based programming still better than machine learning model because of many reasons such as the number of data or knowledge about data.