Abstract

With high-powered computer's invention, high-frequency trading has become a common trading method nowadays. It can execute orders based on market conditions in a short time. Our project can be separated into unpacking trading data and building a suitable prediction model.

Firstly, we obtain the raw trading information from Taiwan Futures Exchange. With the encoding method provided, we can obtain the status of the order book and information on each transaction, including the price and quantity. Secondly, we build a model that can predict the price of a specific product so that it can make a profit. The models we have looked into are ARIMA, LSTM and hybrid ARIMA-LSTM models.

The use of time series forecasting is to predict future values based on previously observed values. Many researchers discover that a hybrid ARIMA and LSTM model has better performance than pure ARIMA or LSTM models. Therefore, we decide to take a look at their differences and results.

After comparing, we find there is still room of improvement for the hybrid ARIMA-LSTM model. However, the ARIMA model has already reached the best performance it can be. For this reason, we decide to aim at LSTM optimization in the final part. In the end, the performance of LSTM also improved a lot compared to the former models.

1. Introduction

1-1. Background

The rapid-fire computer-based high-frequency trading developed gradually since 1983 after NASDAQ introduced a purely electronic form of trading.[1] High-frequency trading is a trading method that requires fast execution speed and an automated trading strategy. It is profitable because it grasps the opportunity when there is a sure profit. The other advantage of high-frequency trading is that it can simultaneously process large volumes of information. It may consider whether there are patterns or recurring events. It also can search certain news databases for the company name or monitor other correlation stocks. To sum up, with high-powered computers and the evolution of artificial intelligence, high-frequency trading becomes a popular trading strategy.

1-2. Purpose

1. Compare time-series forecasting between ARIMA, LSTM and hybrid model
2. Regression and Classification in the LSTM model

2. Research Methodology

2-1. Market data processing

First, we obtain the raw trading information from Taiwan Futures Exchange. Then, we follow the format in the transaction manual from Taiwan Futures Exchange to unpack the trading packet.[2] The main packets we unpack include data with message IDs I024, I081 and I083. The I024 messages provide prices and quantities of each matched transaction. The I081 and

I083 messages indicate the market data prices and quantities in five levels. Among these two, the I083 message is the data during the pre-market session, which is from 8:30 AM to 8:45 AM. Therefore, we choose to use the I081 message as our main data resource.

We aim to analyze the product with better market liquidity and larger trading volume. Taiwan Index Future(TXF) meets these features. Its trading volume takes the largest proportion in the Taiwan Futures Market, and TXF has a lower transaction cost compared to other commodity futures. In conclusion, we filter out the unpacked market data with product TXF and message ID I081.

2-2. Selection and Analysis between models

The stock market can be viewed as a combination of linear and nonlinear time series. To fit this characteristic, we aim to use a time series forecasting model to predict future values based on previously observed values. As for the linear and non-linear patterns, we create multiple models including ARIMA, LSTM and ARIMA-LSTM hybrid to deal with the hybrid pattern.

The reason why we select ARIMA and LSTM models is that ARIMA is a linear model in time series forecasting and LSTM is the nonlinear one. In the following part, we will compare the results and combine them to obtain better prediction accuracy.

2-3. ARIMA Model Introduction

The Autoregressive Integrated Moving Average(ARIMA) model characterizes time series by going from three fundamental aspects:[3]
- Autoregressive terms (AR) that model past process information.
- Integrated terms (I) that model the differences needed to make the process stationary.
- The moving average (MA) that controls the past information of noise around the process.

First, we want to break down the time series so that we can have a better understanding of the data's characteristics. We choose to use the Error-Trend-Seasonality(ETS) model to decompose the time series into error, trend and seasonality components.

Second, the data should be different until it is stationary. The reason is that ARIMA has a parameter d which means the degree of differencing. To check whether the time series is stationary, we use the augmented Dickey-Fuller (ADF) test. The ARIMA parameter can be calculated by the auto_arima() function. It will automatically run through every (p,q,d) pair and find the smallest AIC(Akaike's Information Criterion) which is useful to determine the order of an ARIMA model.

2-4. Hybrid ARIMA-LSTM Model Introduction

With little knowledge of machine learning, we decided to start with US daily stock market data. We choose GOOG as it has been in the market for a long period, and it is also one of the famous companies. This dataset includes open, high, low and close prices and volume of each day, and we use it to predict the opening price the next day.

There are several steps we conduct time-series forecasting by the hybrid ARIMA-LSTM model. First, we split the data into 90% for training and 10% for testing. Next, we construct ten kinds of moving averages which are stated in Talib's Moving Average(MA) indicator module. For each MA, we further set its period from 1 to 100, so that we can obtain 1000

kinds of datasets.(Fig. 2-1) The reason why we construct these datasets with different MAs and periods is to find out the one closest to normal distribution. This can be tested by the kurtosis values of each dataset. The kurtosis value of a normal distribution is 3 which is stated in the definition of the statistics. We select the MA dataset which is closest to a normal distribution by checking whose kurtosis value is closest to 3.

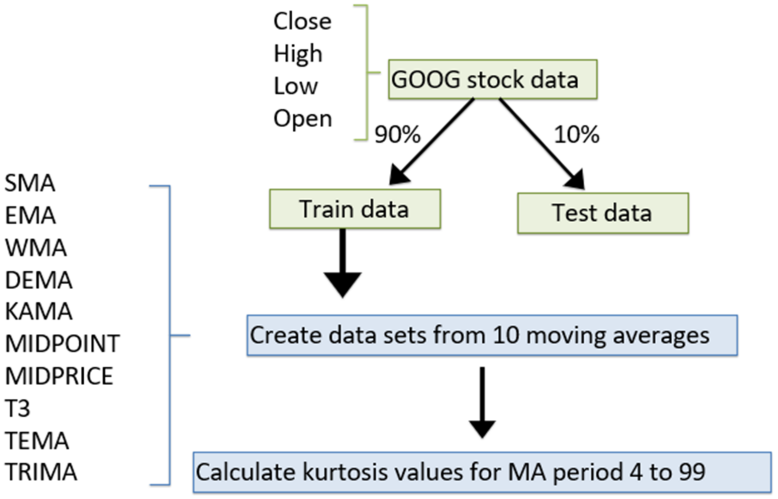

Fig. 2-1 Construction of dataset

Then, we view this MA dataset as a low-volatility dataset because MA is a linear dataset, and MA is also accounted for the volatility of a market. As for the high-volatility dataset, it is obtained by getting the real open price dataset rid of the low-volatility dataset. Then we use the low-volatility dataset as input data for the ARIMA model we mentioned in the previous section. As for the LSTM model, its input is the high-volatility dataset, and its parameter is the same as the LSTM model in the further section.(Fig. 2-2)

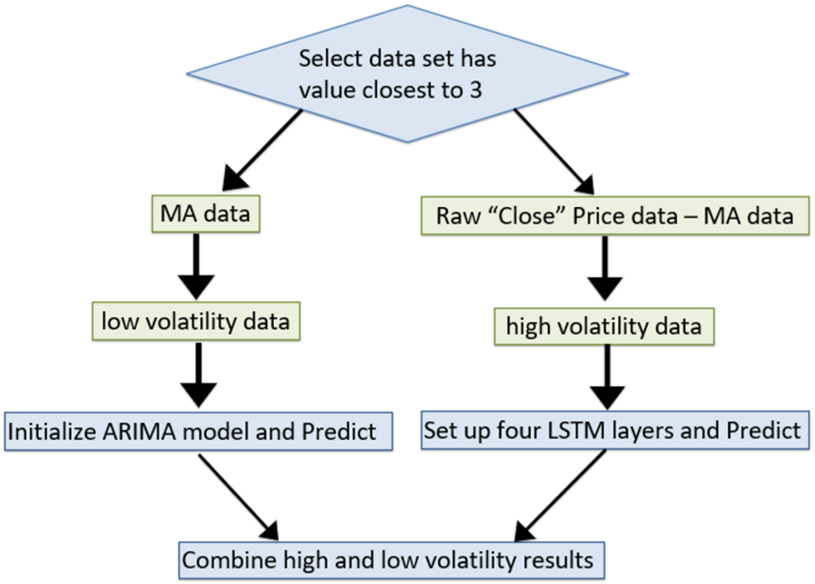

Fig. 2-2 Hybrid ARIMA-LSTM prediction flow chart

2-5. Results of ARIMA and Hybrid Model

After combining the two models' results, we can obtain a final result including linear and nonlinear prediction. Next, we decide to compare them in two ways: error rate and accuracy.

In the error aspect, we use MSE(Mean Squared Error) and RMSE(Root Mean Square Error). It is used to calculate the amount of error which means the distance from our prediction to the correct value.

As for the accuracy part, we can measure it by the price trend. For example, if today's open price is higher than yesterday, and the prediction of today's open price is also higher than the prediction of yesterday's open price, then it stands for our prediction trend is the same as the actual trend. The other possibility is the prediction of today's open price is also higher than yesterday's actual open price, not the prediction one, then it also means our prediction trend is correct. In the end, we obtain the below results(Table 2-1). Since the ARIMA model is determined by the smallest AIC between all the possible parameters, the model reaches its best condition and is viewed as fixed. To make the hybrid model perform better, we aim to improve our LSTM model part without changing the ARIMA parameter.

Table 2-1
Results of ARIMA and Hybrid ARIMA-LSTM models

	Prediction vs Actual:	Prediction vs Prediction:	MSE	RMSE
ARIMA	54.34%	62.81%	336	18
Hybrid	55.68%	62.14%	2209	47

2-6. LSTM Model Introduction

2-6-1. LSTM model - Regression

The features we use here are open, high, low, and close prices and volume of each day, and predict the opening price the next day.

We chose to predict the stock price the next day's open price (regression model). However, there is always a delay in the predicted price, which leads to no use of the result.

We looked into the actual and predicted data, there are lots of crossing or opposite trends. (Fig. 2-4, indicated in the green box)

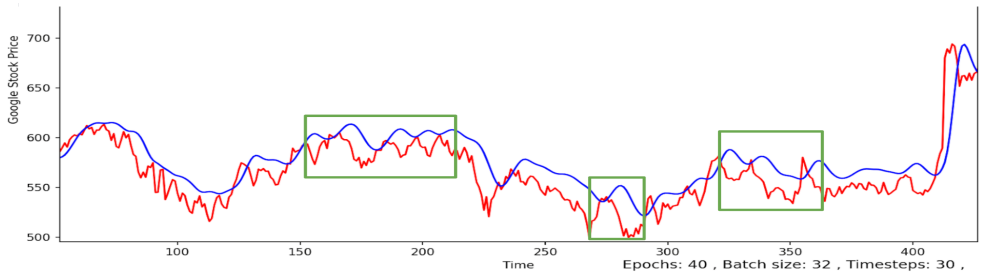

Fig. 2-3 The actual and predicted data from LSTM Regression model

We figure the Mean Square Error is not penalizing the prediction with the opposite trend of the predicted result. The modification of the loss function was made to double the loss if the opposite trend prediction is over half in a batch. However, the result was worse than the standard MSE (Fig. 2-5). We presume this method would make it less likely to find the optima point.

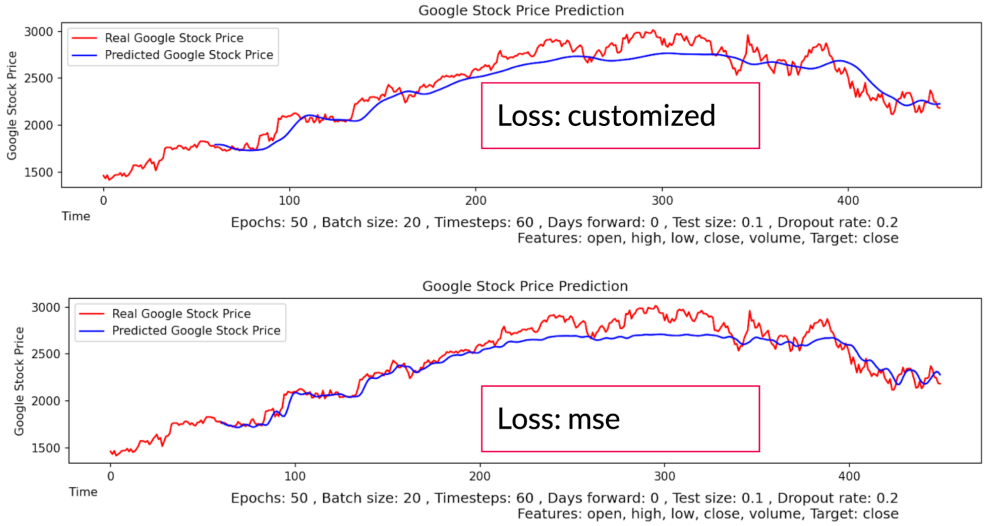

Fig. 2-4 Different loss functions' results

With the best model we have, we used it for training on TXF. Setting the threshold for triggering the buy/sell is harder than we thought. The range is very small and it differs every day.

2-6-2. LSTM model - Classification

With the threshold and the prediction delay, we switched to the classification model. Predicting if the price trend is going up, down_or_flat between current sell, buy price and the next 30 order book sell, buy price.

With the purpose of predicting the market, we have to make sure the data we used for training and predicting have some similarities but not from the aspect afterwards. For TXF, each contract expires on the third Wednesday of the month. TXFF2 is the June 2022 contract, which ends on June 15. We used the data from 6/13~6/15 for training, and 7/18~7/20 for testing

The features we use here are: time gap from last order book update, 1st ~ 3rd buy/sell price, quantity, MACD, EMA.

With the best model we can perform okay in testing (using the same day, but different time of the day). (Fig. 2-6)

However, to predict the data of other days, we get more predictions on actual flat_or_down, but predicted trending up, which would not be what we want. (Fig. 2-7)

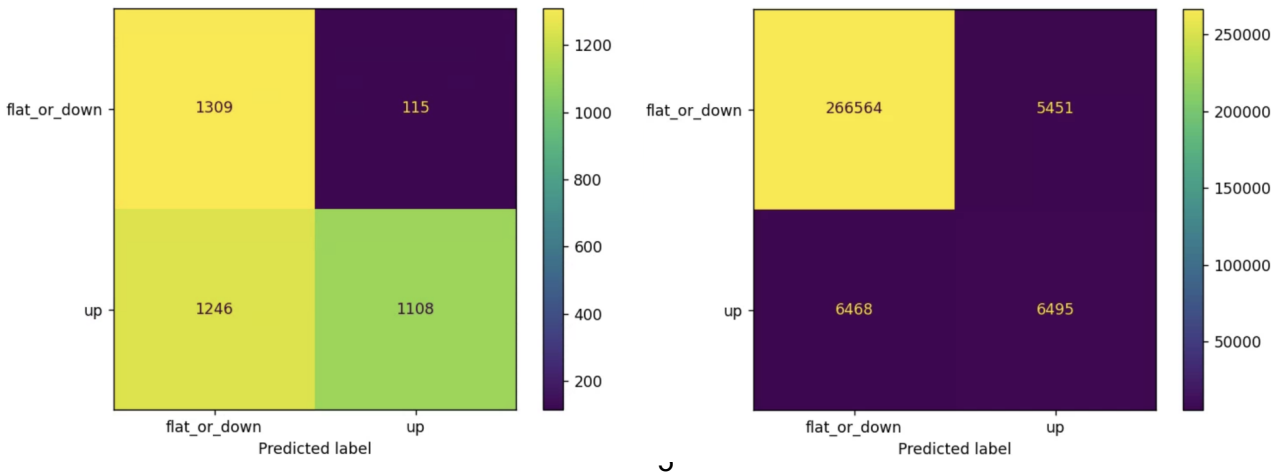

Fig. 2-5 Model performance with different testing items

3. Conclusion

The trading information takes time to unpack and need to follow specific decoding rule. After obtaining the data, we start with GOOG stock data which is easier to check the long-term prediction. Then, we try to compare three models - ARIMA, LSTM and hybrid ARIMA-LSTM model. Next, we decide to improve the LSTM model and test with TXF data. The result obviously improves a lot and will be put together into hybrid model in the future. After this individual project, we found that machine learning is not just a magic box as we thought. There are a lot of parameters to adjust and the choice of features also plays a big part. We also look into several technical indicators and other financial knowledge. This also plays an important role.

4. Reference

[1]Aldridge, I., 2013. High-Frequency Trading: A Practical Guide to Algorithmic Strategies and Trading Systems, 2nd edition. Hoboken: Wiley. ISBN 978-1-118-34350-0
[2]逐筆行情資訊傳輸作業手冊(行情資訊傳輸網路), 版本1.2.0, 臺灣期貨交易所
[3]Oussama FATHI,Velvet Consulting,64, Rue la Bo´etie, 75008, Time series forecasting using a hybrid ARIMA and LSTM model

Gao, Qiyuan. *Stock market forecasting using recurrent neural network*. Diss. University of Missouri--Columbia, 2016.

Pai, Ping-Feng, and Chih-Sheng Lin. "A hybrid ARIMA and support vector machines model in stock price forecasting." *Omega* 33.6 (2005): 497-505.

Zedric Che, Customize loss function to make LSTM model more applicable in stock price prediction
https://towardsdatascience.com/customize-loss-function-to-make-lstm-model-more-applicable-in-stock-price-prediction-b1c50e50b16c

5. 計劃管理與團隊合作

5-1. 計劃管理

在寒假開始前和指導教授聯絡，在寒假期間自學計算機網路概論，學期間與暑假每周進行一次會議，報告內容包含本周進度、遇到的困難以及下周規劃， 過程中也可以聽其他組的創意，指導老師也會在報告過程或結尾給予意見和想法，實驗室的學長則給我們封包資料以及交易成本等細部資訊。

5-2. 團隊合作

5-2-1. 鄭佳渝同學**:**

負責解封包與建立委託簿，以及ARIMA模型與混和模型製作，對應報告則是2-6以外的所有部分，以及撰寫2-5以外的所有部分

5-2-2. 廖庭輝同學**:**

負責LSTM模型與混和模型製作，對應報告則是2-6及協助2-4，以及撰寫2-6的報告內容。

6. 心得感想

首先，我想感謝馬席彬教授這一年的指導與幫助，在剛加入時，我們沒有修習過計算機網路概論和機器學習，於是教授鼓勵我們利用寒假時間上網透過開放式課程自學，並在開學時給予我們時間嘗試解封包，暑假期間也願意花時間聽我們報告並給我們方向，學期間每周報告都會給予我們一些想法和未來需要考慮的地方，讓我們下周可以繼續進行，另一方面，感謝實驗室的學長們花時間幫忙調封包的資料，對於我們不懂的地方也在短時間內回覆、給予詳細的資訊。

此外，謝謝我的組員廖庭輝同學，這一年期間不管是在討論交易策略或是深度學習模型，他都很願意傾聽我的想法，每當我需要找人討論或是懷意自己的研究方向時，他總是能很快回覆，這一路以來，我們不僅並肩作戰，更是一同學習與前進的夥伴。最後也謝謝另一組做同門領域的專題同學，過程中我們不僅互相討論與協助，也給予對方鼓勵堅持到最後。

做研究並非想象中容易，從上台報告到解決問題，都沒有一個規律的解法，只能在誤打誤撞的過程中慢慢成長與學習，不論是計算機網路概論還是機器學習，現在對我們而言不僅是課本上的內容，更可以實作於很多方面，對於學術寫作與閱讀學術文獻也有很大的進步。