Stock-loss Prevention: Mobile application with CNN-LSTM model for predicting sharp rises and falls in stock price

Borislav Pavlov, Young Oh Kim, Geo Ryang Park, Min Jae Kim

Sungkyunkwan University, 2066, SEOBU-RO, JANGAN-GU, SUWON-SI, GYEONGGI-DO, KOREA
Capstone project by prof. Hyung Joon Koo
October 3, 2021

Abstract

We face a lot of stock information every day. Among them, there is truth, there is luck, there is deceit, and there is sedition. So, quant that focuses on charts rather than information has emerged. But no one, including us, can make accurate predictions. Rather than predicting stock prices, we want to create a CNN-based alert program to protect users' wealth. As people's interest in the stock market becomes more active, issues related to stock price manipulation are frequently reported on the news. Also, there are general stock investors suffering from unpredictable forms of abnormal price fluctuations. To prevent this, we planned this project, and we predict that the results completed through this project will be of great help in preventing this phenomenon. Our project aims to predict the surge and plunge in stock prices through AI models. CNN-based AI models learn images of several abnormal stock price graph models through input. The CNN model expects to extract common features from images used in learning and predict theoretically incomprehensible exception situations. In addition, we would like to introduce XAI (explainable artificial intelligence) technology to provide a valid reason for predicting results. Among the XAI techniques, the gradient-CAM technique in which the slope is applied to the image will be used. In addition, for the convenience of users, we will provide text on the results and recommended behaviors, along with statistics to examine other users' behavior trends in the same situation.

Contents

1 Problem statement and motivation

Nowadays, the stock market is completely detached from the economic reality and depends much more on the media. A simple example is the tweets that Elon Musk wrote on the famous social media platform - "Twitter" regarding companies that he thinks are favorable for him. For instance, Elon tweeted that "Signal" is a better messaging platform that supports full end-to-end encryption and the stock price of the company rose by 5.100%.

Figure 1: Elon Musk's Signal

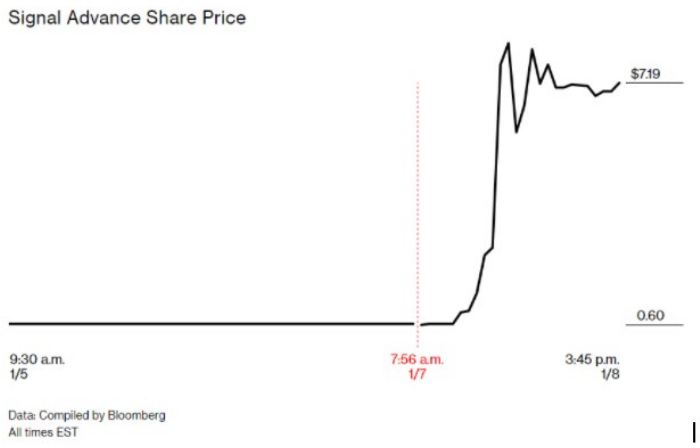


Figure 2: The example of surge. [reference 1]

Another example is the Dogecoin cryptocurrency that was by no means taken seriously before Elon Musk tweeted that he bought some of it to his son.

Figure 3: Another example [reference 1]

After that, the cryptocurrency rose by 1500%

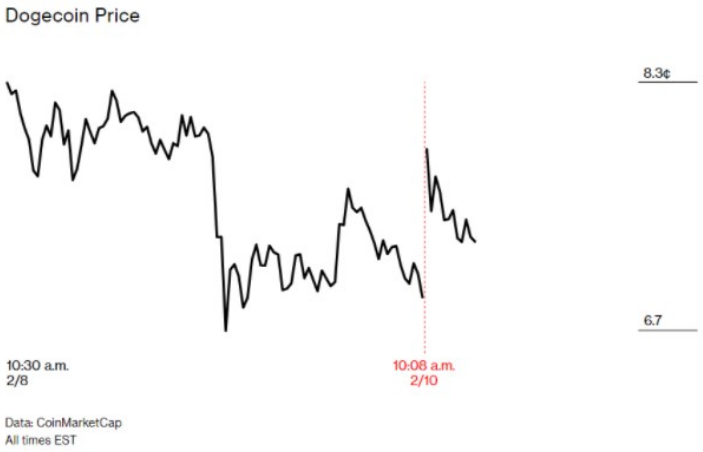


Figure 4: There's a meme I'm going to Mars.[reference 1]

It is clear that the hedge funds on Wall Street cannot accept such market changes and people that were hyped by Elon Musk quickly lost all of their investments after the market went back to normal again unexpectedly. People are often harmed by unpredictable stock surges or plunges due to unexpected causes, rather than the general stock flow.

The goal of this project is to create an application that will give warning notifications after detecting a surge or plunge in stocks that are tracked by the people using the application.

The main research question is: "How to help people handle unexpected profit losses due to unpredictable stock market plunges and surges using software development tools and technologies with a combination of AI and machine learning?"

2 Methodology and related works

There is no application with the same idea and goal, however, there are similar solutions that focus on predicting stock prices.

1. Forecasting stock prices with a feature fusion LSTM-CNN model using different representations of the same data

Figure 5: Forecasting stock prices using LSTM-CNN model [reference 2]

- CNN-LSTM combined model to predict stock prices

- It gave better results than prediction with a single model (CNN, LSTM, RNN, etc.).

- Input data: SPDR S&P 500 ETF data (Time series data and stock graph images) → The S&P 500 is data that contains the commonality of stocks rather than single stocks. Because it includes stocks of 500 American companies, making it a bit more universally predictable.

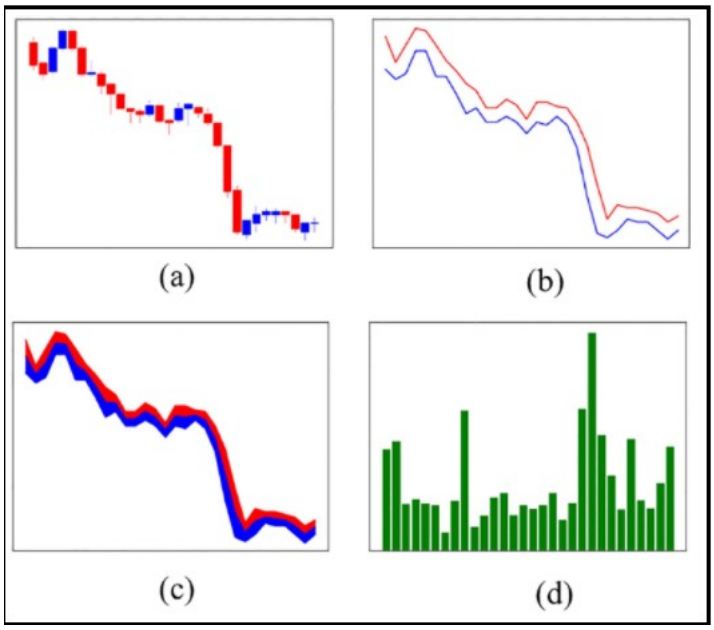


Figure 6: (a)Candlestick chart (b)Line chart (c)F-line chart (d)Bar chart [reference 2]

- Result : Good prediction with graph (a), bad prediction with graph (b).

- It can be helpful with some indicator that can show the shape of surges and plunges. Give indicator to AI predict model then, the model can use indicator at the learning process.

- https://github.com/luanft/lstm-cnn-model

2. Grad-CAM : Visual Explanations from Deep Networks via Gradient-based Localization

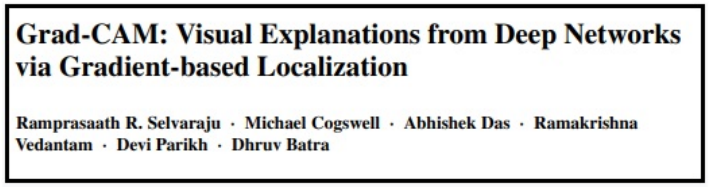


Figure 7: Grad-CAM 1 [reference 3]

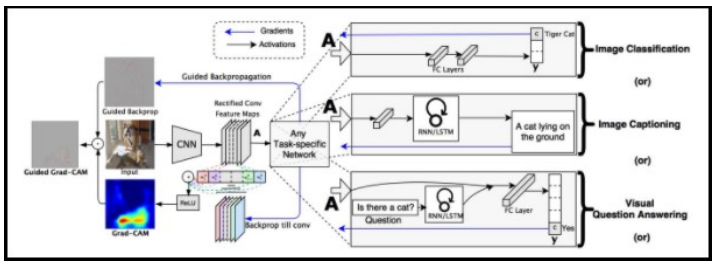


Figure 8: Grad-CAM 2 [reference 3]

- It is a type of XAI visualization technique that is often used in image.

- CAM method uses GAP (Global Average Pooling) instead of using the Fully-Connected layer immediately after going through the Convolution layer.

- Grad-CAM uses the gradient of the convolution layer in the AI model and information of output that passes through that convolution layer.

- Result : Grad-CAM method can explain the reason which convolution model predicts with input images. Below the images are shown.

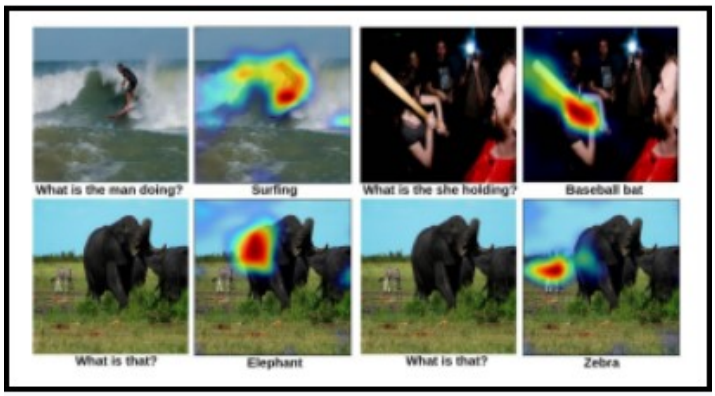


Figure 9: Grad-CAM 3 [reference3]

3 Benefits

- Increase user profits through prediction and notification of stock price surges and sharp declines
- Unexpected profit losses are prevented
- Giving justification of the prediction results through XAI
- Open source

4 Team Roles

The team leader for this project will be Borislav Pavlov.

[AI production]

Main Responsible : Kim Min Jae, Kim Young Oh
Support : Borislav Pavlov, Park Geo Ryang

[Mobile Application and Backend API]

Main Responsible : Borislav Pavlov, Park Geo Ryang
Support : Kim Min Jae, Kim Young Oh

5 Planning

1. AI Development
   - Reimplementation of CNN-LSTM combined model in existing papers (4weeks)
   - Implementation of XAI Grad-CAM model and application to CNN layer (4weeks)
   - Performance improvement : Create indicators and use them for learning

2. Mobile Application Development

   Research & Design
   - Research a framework for developing native mobile applications for both iOS and Android systems
   - Research Websocket communication implementation for mobile devices
   - Create initial wireframes for the mobile application

   Implementation & Testing
   - Implement the initial design view of the application and state management - components for rendering stock items, selecting stock items, and removing selected stock items
   - Implement Websocket service communication and test it with the backend service

- If there is time left, implement an open-source broker provider API for getting actual stock investments of the user

3. Backend API Development

   Responsible for processing stock information and insights by the AI and delivering them to the client application

   Research & Design

   - Research which language is the better for the use case - Node.js or Python

   - Research Websocket communication implementation for mobile devices

   - Research how to integrate the open-source yahoo-finance API for getting real-time stock data, or find a better API

   -Research how to integrate the AI model into the service

   Implementation & Testing

   - Create all client and server messages needed for the Websocket communication

   - Implement a Websocket server and test it with mobile application clients

   - Implement any open-source API for getting real-time stock data

   - Integrate the AI model and test it with the mobile application

   - If there is time left, set up a Redis database for keeping all client messages sent - Needed for the proper scaling of the service due to Websocket drawbacks

4. Infrastructure Overview

   On the figure below is illustrated the initial infrastructure design. Separate services will be implemented for the mobile application, the stocks service, the AI model, and the identity service. The mobile application will be from where all the stock alerts will be used by the users.

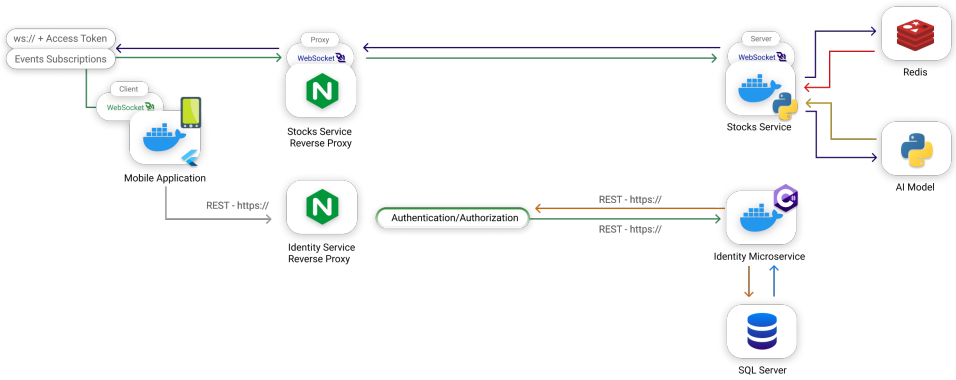


Figure 10: Initial Infrastructure Design

The stocks service wil be responsible for getting stock data, as well as real time updates and serve them to the mobile application through a web socket connection. Furthermore, the stocks service will evaluate in a pre-defined time frame any expected surges or plunges (using the AI model) regarding all of the alerts set by the user. Moreover, the stocks service will be responsible for delivering notifications to the users whose alerts were triggered by the AI. All of the notifications and alerts will be stored in a Redis database.

Each user must identify themselves before using the application, and this would be the role of the identity service. All of the user data will be stored in a SQL Server database.

Finally, two reverse proxies will be set if the time allows in front of the identity and stocks services. The reverse proxies is much more urgent and the focus would be on it because all of the workload is expected to go through the stocks service.

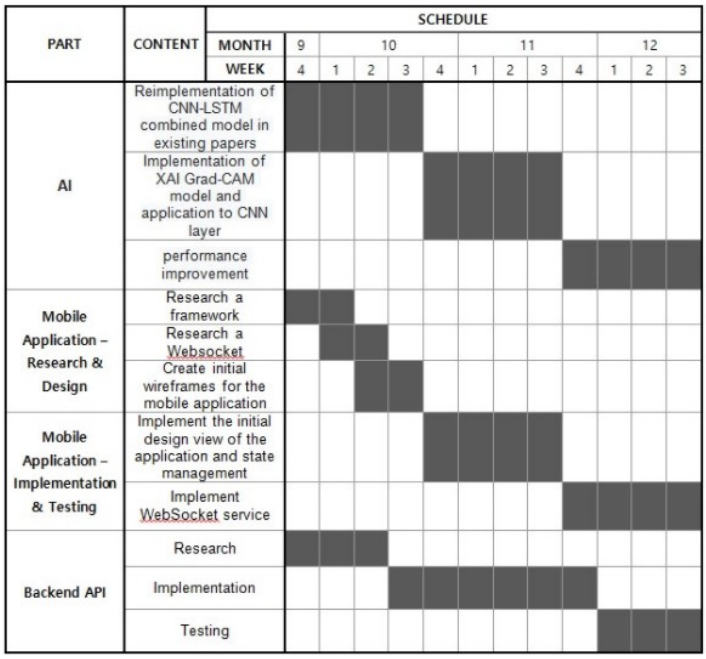


Figure 11: A summary of the schedule

6 AI Model Justification

1. Supervised or unsupervised?

   - Supervised learning

2. Dataset you want to utilize for training?

   - Dataset : minute-by-minute SPDR S&P 500 ETF Trust (SPY) ticker data _from_ Thom-son Reuter Database

   - Ratio (training/validation/test)

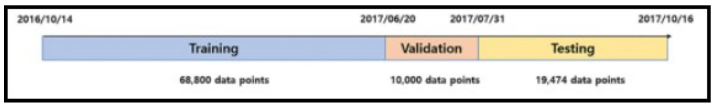


Figure 12: Ratio of training, validation and testing [reference 2]

CNN or CNN-LSTM model (not choose yet)

Input formation : There is Kaggle dataset that has Daily stock market prices. Date, Volume, High, Low, and Closing Price (for all NASDAQ, S&P500, and NYSE listed companies). Updated weekly. [reference 4]

Output formation : predicted prices for some period(*ex. 1month)

XAI : Grad-CAM for image

3. Goal

   [First Priority] Research the existing implementation.

   [Subordinate] Integrate the existing solution to be compatible with heatmap prediction.

   [Subordinate] Performance improvement is attempted by introducing indicators.

4. Algorithm or model

   - CNN or CNN-LSTM fusion model

5. Existing techniques to take advantage of

   - Open-source (MIT License)

   - There is a paper along with the published code.

6. Evaluation

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_{1,i} - x_{2,i})^2}$$

$$RMAE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}|x_{1,i} - x_{2,i}|}$$

Figure 13: Equation of RMSE and RMAE [reference 2]

- RMSE or RMAE

The root mean square error (RMSE), the root mean absolute error (RMAE) we use RMSE as a loss function, which means that the model is trained to reduce RMSE. RMSE is a good measure for revealing relatively large forecast errors RMAE is useful for revealing the systematic bias of the model. N is the number of data points, x1,i is a predicted value, and x2,i is a real value.[reference 2]

- relative comparison formula = (original error - our error)/(original error)*100%

7. Computation resource

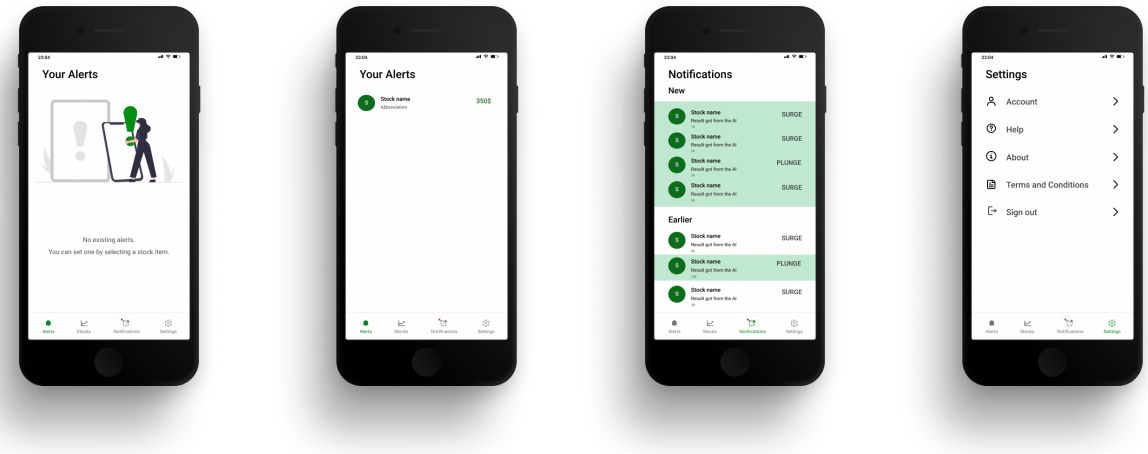
- Cloud or personal laptop

8. Limitations

- Code with high accuracy may be hard to implement.

7 Mobile Application

The mobile application will be available on both iOS and Android. The users would need to provide information regarding what stocks that they want to receive warnings. Furthermore, in order to not have bounded context and prevent unnecessary spamming of the application notifications, the users need to specify the percentage of change in the stock price for which they want to receive notifications. The initial design of the application is illustrated on the figure below.



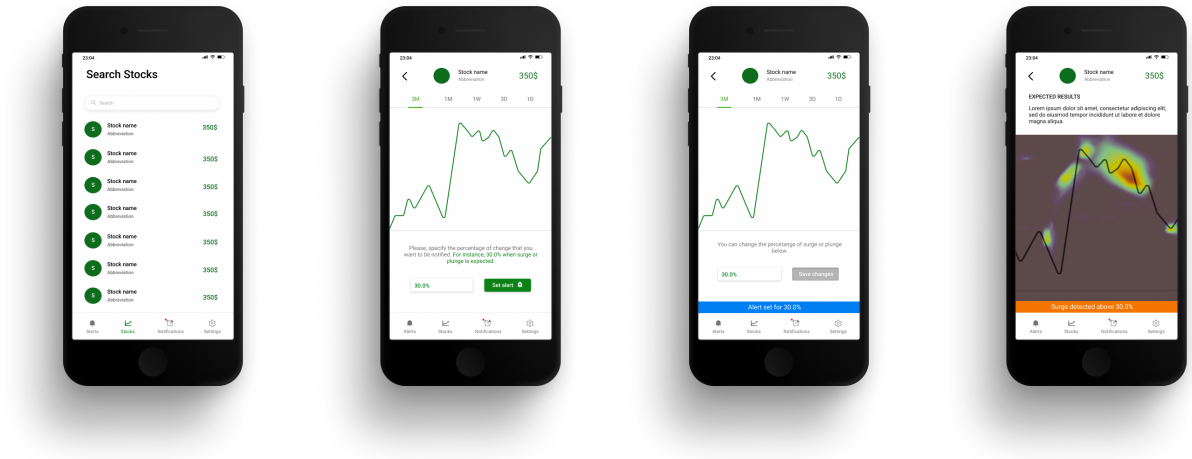


Figure 14: Mobile Application Initial Mockups

8 References

1. https://www.bloomberg.com/news/articles/2021-02-11/how-elon-musk-s-tweets-moved-gamestop-gme-bitcoin-dogecoin-and-other-stocks

2. Kim T, Kim HY (2019) Forecasting stock prices with a feature fusion LSTM-CNN model using different representations of the same data. PLoS ONE 14(2): e0212320. https://doi.org/ 10.1371/journal.pone.0212320

3. Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., & Batra, D. (2019). Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. International Journal of Computer Vision, 128(2), 336–359. https://doi.org/10.1007/s11263-019-01228-7

4. Paul Mooney, Stock Market Data (NASDAQ, NYSE, S&P500) Date, Volume, High, Low, Close (updated weekly), https://www.kaggle.com/paultimothymooney/stock-market-data