MSBD 5013 Project 2: Replication Study of "(Re-)Imag(in)ing Price Trends"

Zhaoyang Deng* and Congjian Chen* {zdengao, cchenci}@connect.ust.hk
*MSc Candidate in Big Data Technology, Deptartment of Computer Science and Engineering, HKUST

1. Introduction and Task Description

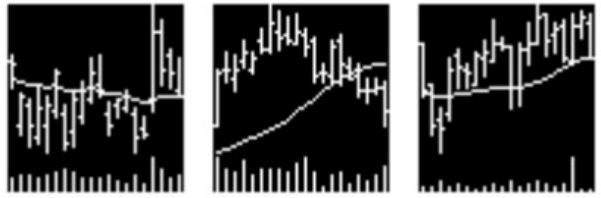
The dynamics of stock markets have always been fascinating, and stock price forecasting from a statistics or machine learning perspective has been a difficult task, because of incomplete data, external economic changes, black swan events, etc. Currently, many price trend prediction tasks utilize the idea of formulating a model of return prediction based on price trends as a test of the weak-form efficient markets null hypothesis.

In this task, we will be replicate this article "(Re-)Imag(in)ing Price Trends", which builds on machine learning to reconsider the idea of price-based return predictability from a different philosophical perspective. Specifically, this article turns the stock trend over a certain time period into an image and apply CNN (Convolutional Neural Network) on the image to predict the future trend (up or down). Essentially, this is a classification problem with output of 0 or 1.

2. Dataset

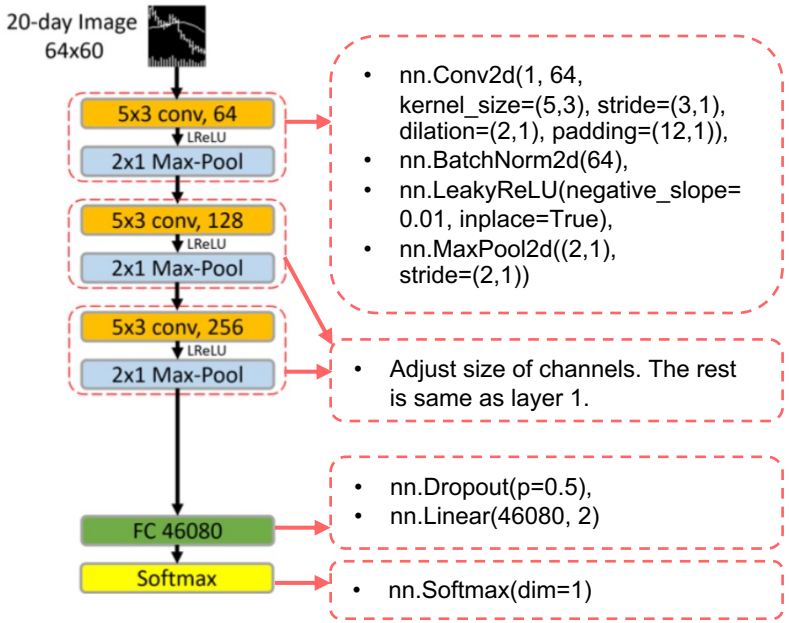
The daily stock dataset is from CRSP for all firms listed on NYSE, AMEX, and NASDAQ. The sample runs from 1993–2019 based on the fact that daily opening, high, and low prices first become available in June 1992.

The original data has into images following the same procedures introduced in the paper (Section 2). Current images have the same resolution (64 * 60) and added with moving average lines(MA) and volume bars(VB). Some example figures are shown as follows.



Also, a series of data files in terms of return rates provide labels for our task. We will use **Retx_20d_label** as our label, which takes value 1 for positive returns for the following 20 days and 0 for non-positive returns.

3. Architecture Design



4. Working Flow

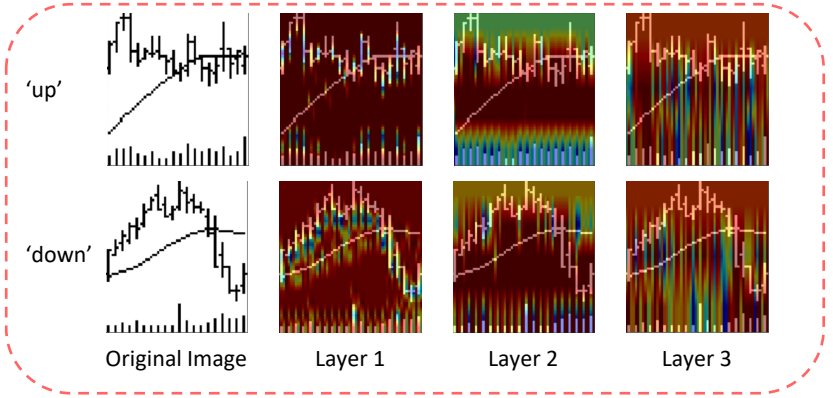
- Data Split: Due to the restriction of computing power, we are not able to use all data like the paper does. Thus, we use samples of 2017 for training and validating (70% for training, 30% for validating) and samples of 2018-2019 for testing, for the reason that during this period, the U.S. went through a normal economic cycle (neither the 2018 rate hike nor the trade war with the start of rate cuts in 2019 had a significant impact), and the exclusion of dramatic economic fluctuations favors the forecast.
- Loss and Evaluation: Cross-entropy loss and accuracy are used.
- Train and Test: Dropout, batch normalization and early stopping are used during the process.

5. Results

The training process stops at epoch 22, with training loss=0.68 and validating loss=0.70. The test loss is 0.73. The accuracy is 0.5074.

6. Interpretability of the CNN Model by Grad-CAM

The visualization by Grad-CAM helps to explain which part of image the network is pay more attention on in different layers. For a certain region of image, the lighter, the more important.



We randomly select some samples with label=1 and label=0. From the results above, we can see that in layer 1, price bars are brighter. In layer 2, the region above price bars is emphasized. In layer 3, the region between price and volume bars are more activated. The difference between 'up' and 'down' images is yet not clear.

7. References

- https://github.com/lich99/Stock_CNN
- https://github.com/jacobgil/pytorch-grad-cam

8. Contribution

Zhaoyang Deng: Data preprocessing, CNN training, validating and test
Congjian Chen: Grad-CAM, poster