Replicaiton:
(Re-)Imag(in)ing Price Trends

WANG Yufeng 21020948

Teacher: Professor Yuan YAO

Department of Mathematics

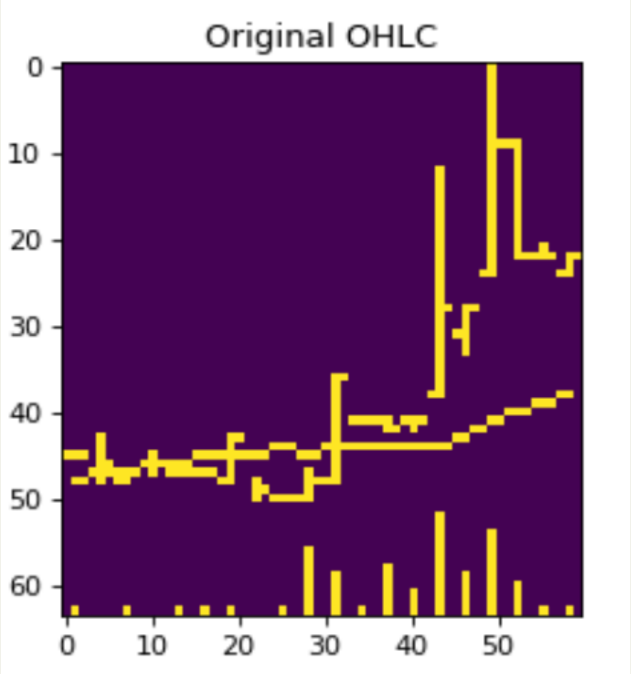
HKUST

香港科技大學
THE HONG KONG
UNIVERSITY OF SCIENCE
AND TECHNOLOGY

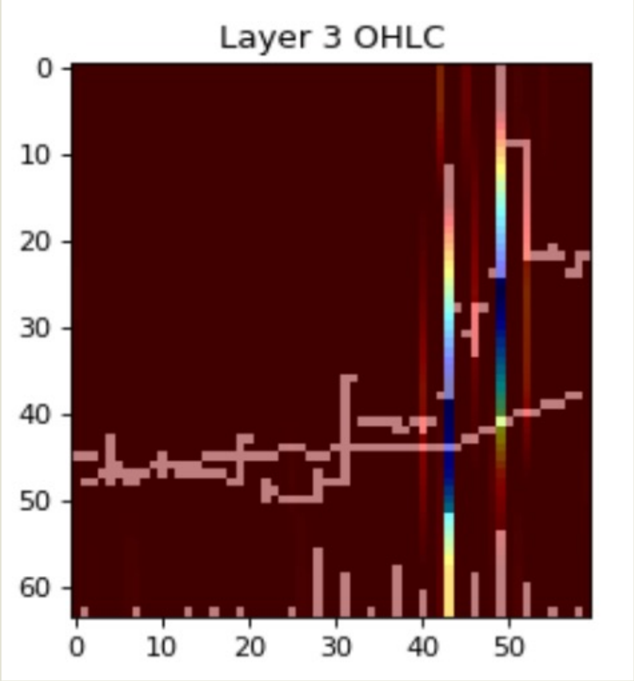
Table of contents

Problem description and motivation

Problem Description



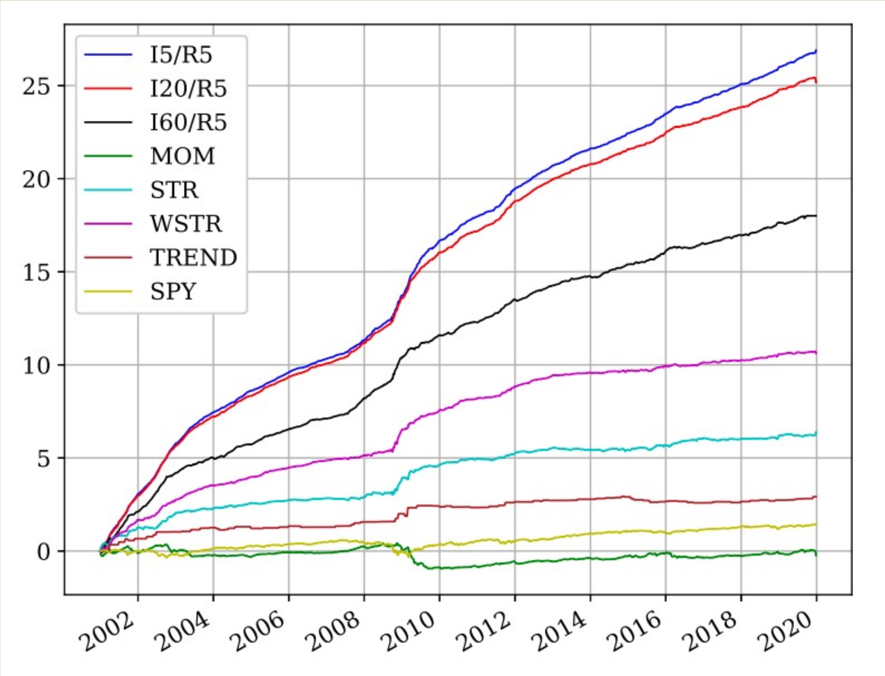
Learn price pattern



Predict

Future return

Construct



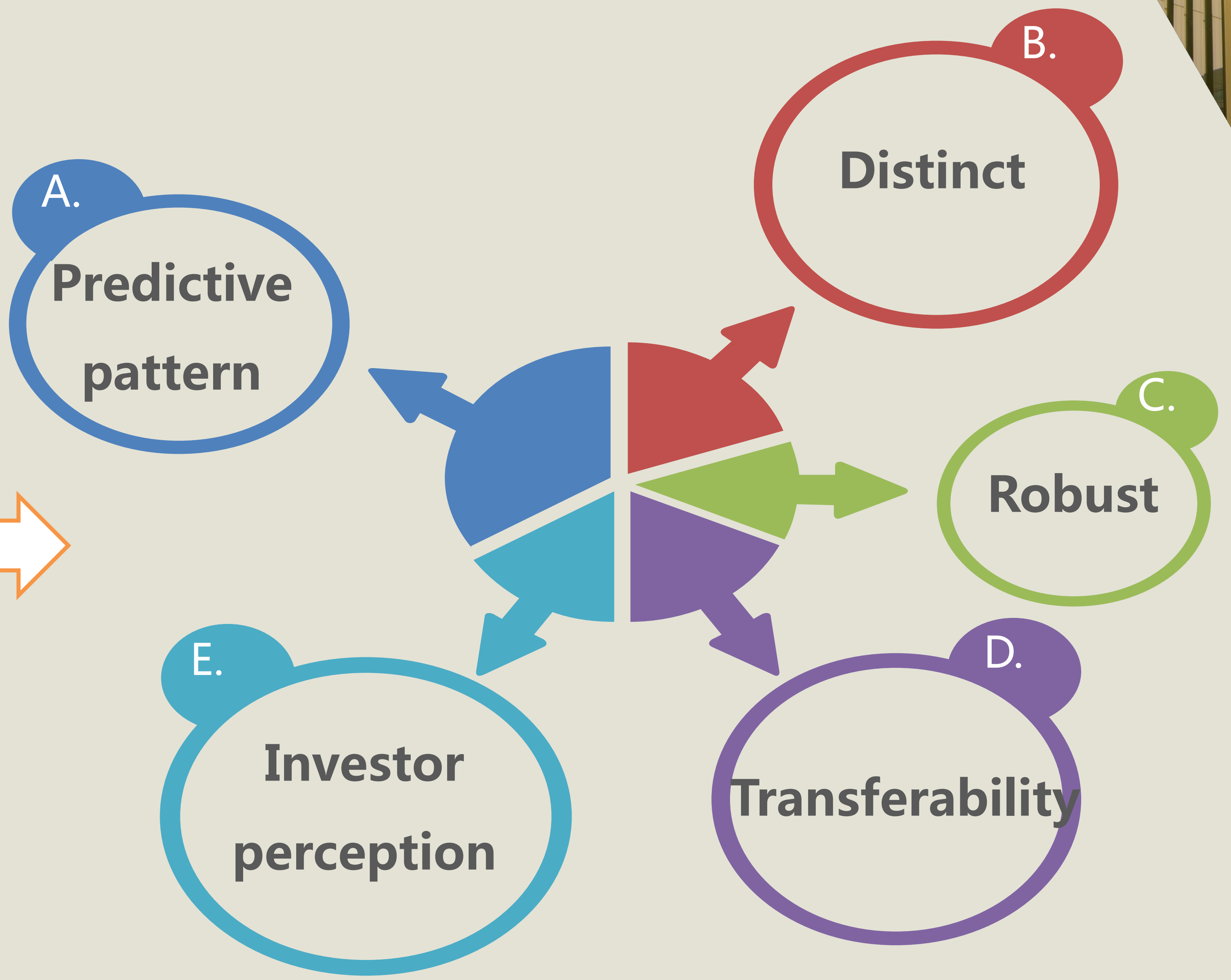
Problem description and motivation

Motivation - Predict Future Stock Return

Common Way

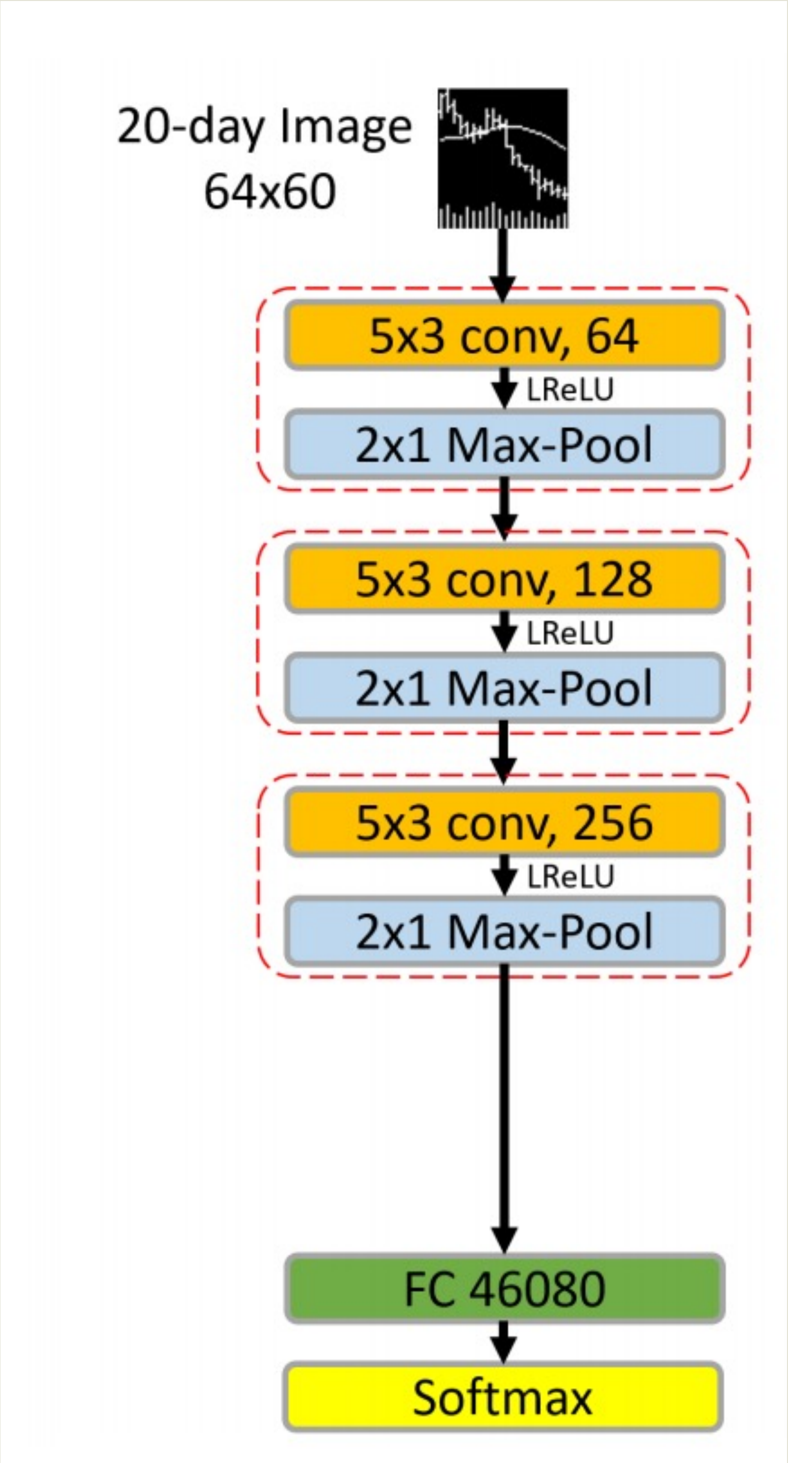
- Using hypothesized or pre-specified patterns
 (e.g., momentum and reversal)

- Directly from one-dimensional time series stock price
(e.g., ARIMA, GARCH and LSTM)



Model

CNN



Introducing CNN method to extract trading signals from 2-D stock images

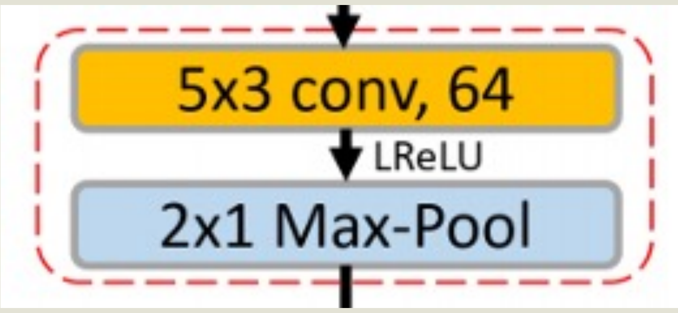
CNN doesn’t require pre-specified patterns,
It automatically extract more predictive and in line with investors’ perception patterns, but also these patterns are distinct from those come form standard methods of empirical finance

2-D images,
help the convolutional filter to obtain some nonlinear spatial relationships of prices

Model

CNN

```
----------------------------------------------------------------
        Layer (type)               Output Shape         Param #
================================================================
            Conv2d-1           [-1, 64, 24, 60]           1,024
       BatchNorm2d-2           [-1, 64, 24, 60]             128
         LeakyReLU-3           [-1, 64, 24, 60]               0
         MaxPool2d-4           [-1, 64, 12, 60]               0
            Conv2d-5          [-1, 128, 12, 60]         123,008
       BatchNorm2d-6          [-1, 128, 12, 60]             256
         LeakyReLU-7          [-1, 128, 12, 60]               0
         MaxPool2d-8           [-1, 128, 6, 60]               0
            Conv2d-9           [-1, 256, 6, 60]         491,776
      BatchNorm2d-10           [-1, 256, 6, 60]             512
        LeakyReLU-11           [-1, 256, 6, 60]               0
        MaxPool2d-12           [-1, 256, 3, 60]               0
          Dropout-13               [-1, 46080]                0
           Linear-14                   [-1, 2]           92,162
================================================================
Total params: 708,866
Trainable params: 708,866
Non-trainable params: 0
----------------------------------------------------------------
Input size (MB): 0.01
Forward/backward pass size (MB): 7.73
Params size (MB): 2.70
Estimated Total Size (MB): 10.45
----------------------------------------------------------------
```



First core block of CNN

Core Block

- Convolution Layer
  - 5×3 convolutional filters and
  - vertical stride 3 and vertical dilation 2 on first layer
  - Suitable padding
- Activation Layer
  - Leakly Relu
- Max pooling Layer
  - 2×1 max-pooling filters

Other Layer

- Fully connected Layer
- Softmax Layer

Numbers of Neuron and Parameters

- Neuron: 46080
- Total parameters: 708866

Model

CNN - Training

Dataset

- Image(64*60) contain OHLC, MA, Trading Volumn
- Train dataset: 1993-1999 (70%)
- validation dataset: 1993-1999(30%)
- test dataset: 2000-2019

Training Goal and Optimizer

- Goal: Minimize Cross entropy loss
- Optimizer: Adam

Hyperparameter

- Batch size: 128
- Learning rate: 1*e^-4
- Dropout: 50% on final layer
- Early stopping: No improvement in the validation set in two consecutive epochs

Other technique

- Batch Normalization
- Xaiver Initializer

Appropriate hyperparameter and techniques can alleviating overfitting problems

Main results and analysis

Main results – Baseline Results

		5d-test-accuracy	20d-test-accuracy
Baseline		0.5285	0.5252
Filter	32	0.5318	0.5140
Filter	128	0.5325	0.5188
Dropout	0.75	0.5327	0.5240
Dropout	0.25	0.5066	0.5186
Dropout	0	0.5269	0.5144
Batch Normalization	No	0.5248	0.5246
Xaiver Initialization	No	0.5300	0.5198
Activation	ReLU	0.5246	0.5236
Pooling	2×2	0.5210	0.5006
Learning Rate	$1{*}e^{-5}$	0.5191	0.5225
Learning Rate	$5{*}e^{-5}$	0.5216	0.4851

After 12 epoch, the model trigger the stopping time

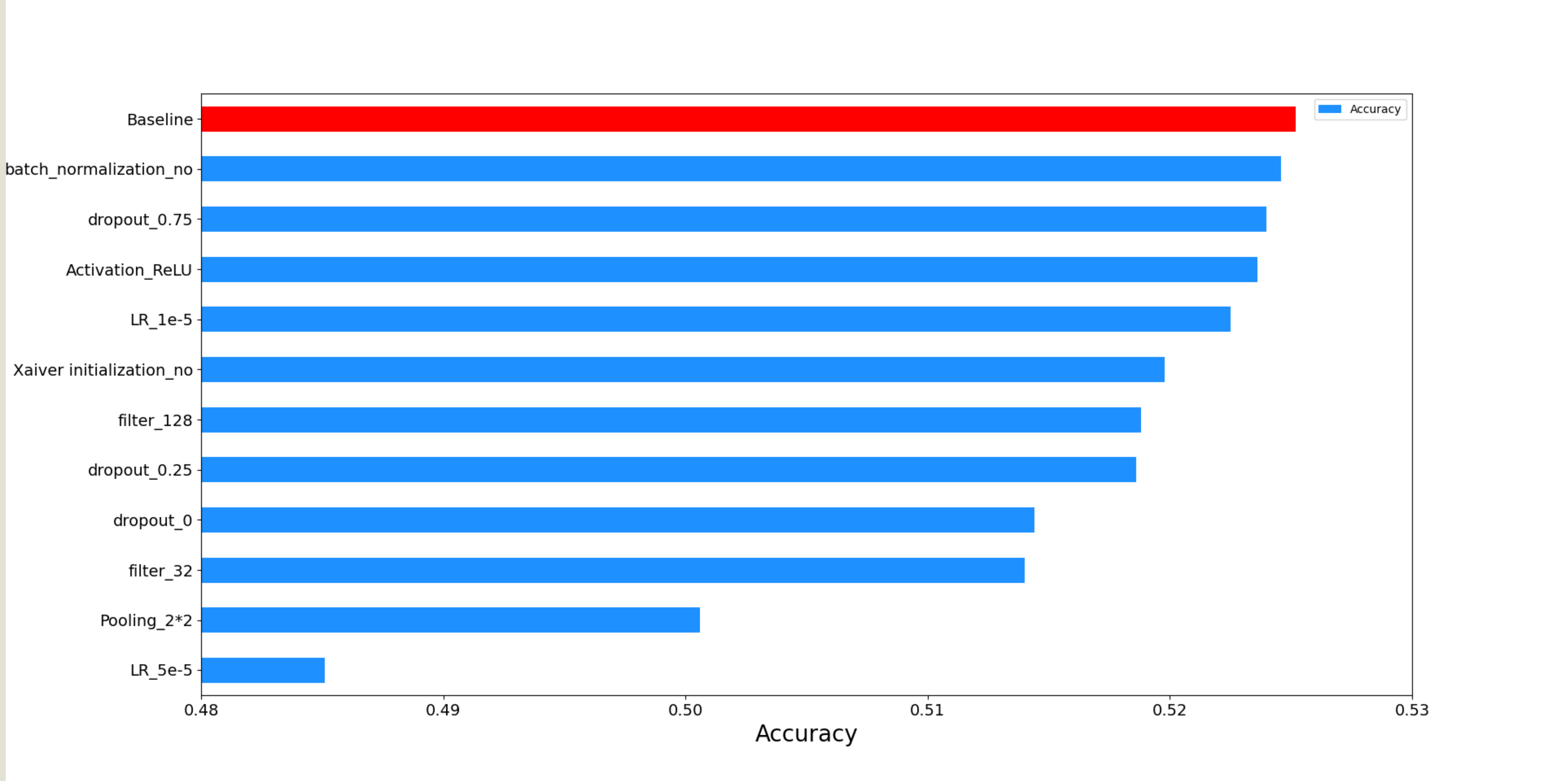
Replication of Baseline result

Accuracy: 0.525

Main results and analysis

Analysis – Sensitivity Test

Predict 20 day return direction accuracy



Slightly decrease accuracy

no batch normalization
increase dropout to 75%
change activation function to Relu
decrease learning rate 1/10

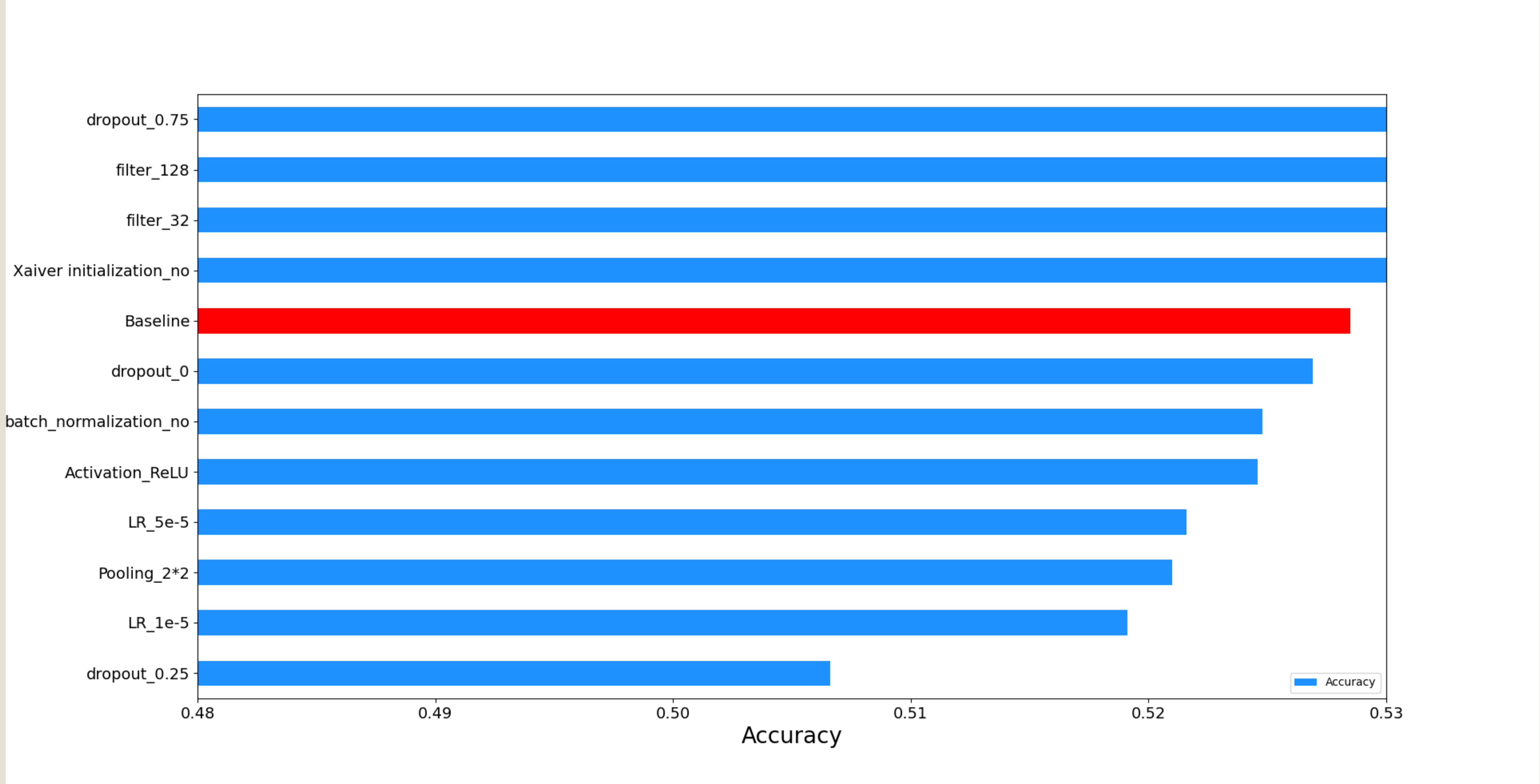
dramatically decrease accuracy

no Xaiver normalization
change filter size
change pooling filter size
decrease dropout to 0, 25%

Main results and analysis

Analysis – Sensitivity Test

Predict 5 day return direction accuracy



Still find the following inferior parameter

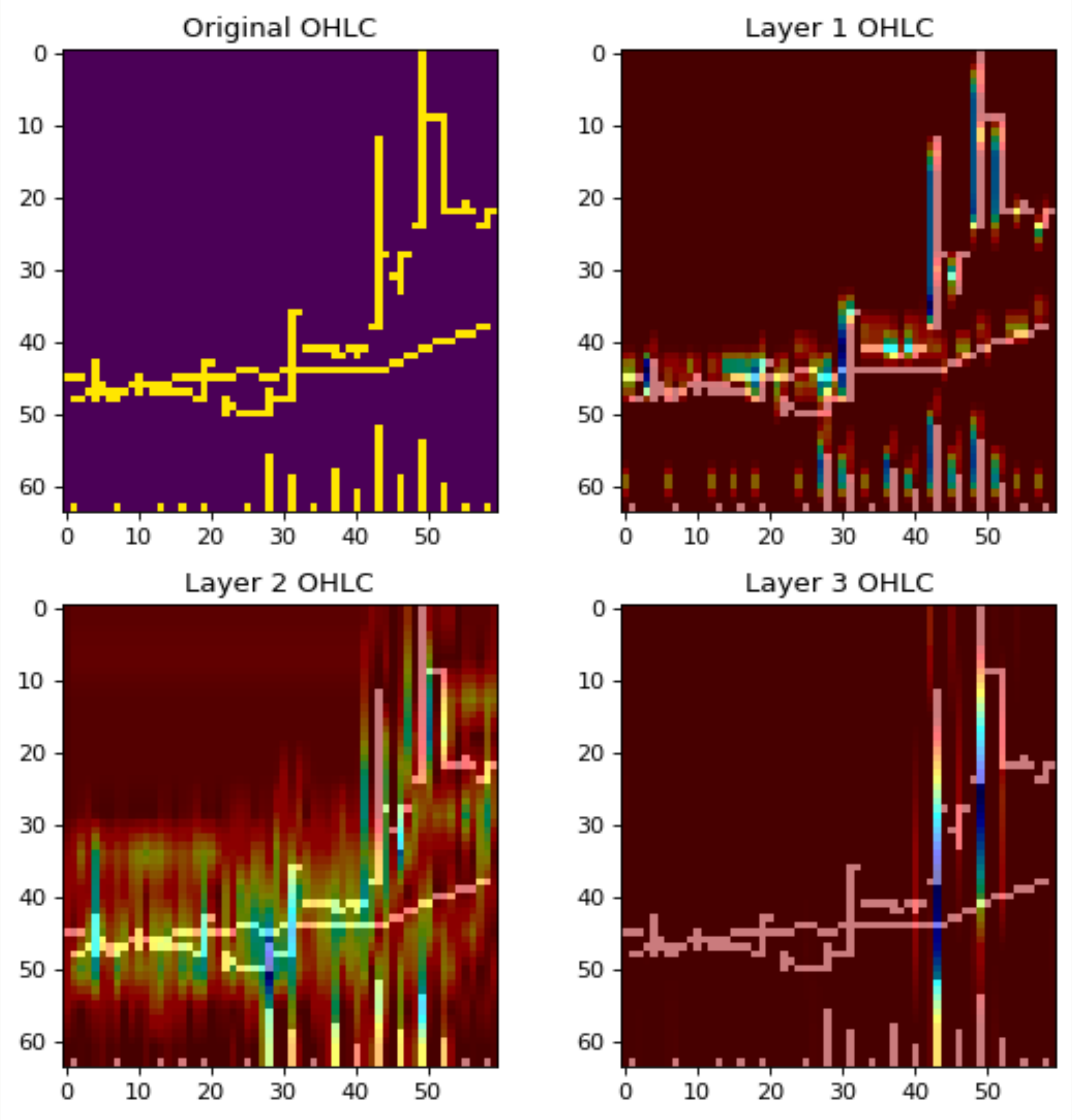
Lower dropout,
Change pooling filter size,
Chage activation function,
no batch normalization

Combine these two sensitivity tests,

1. **Some certain hyperparameters do alleviate overfitting, and the model is robust to specification variations**
2. **Although the short term predict accuracy is higher than the long term, the model also has pretty good long term predict ability**

Main results and analysis

Analysis – Visualization



Grad-CAM
illustrate the regions of the input most important for predicting a class.

Using a sample classified as ’Down’
First Layer

successfully captured the volume and price information of the stocks in the image, but it was relatively broad

Second Layer

gives more weight to high trading volume

Third Layer

focuses on the two longest shadow lines, that is, high intraday fluctuations day, which is consistent with huamn technical analysis.

Conclusion and future direction

Conclusion

We've verified CNN has these property

Predictive patterns

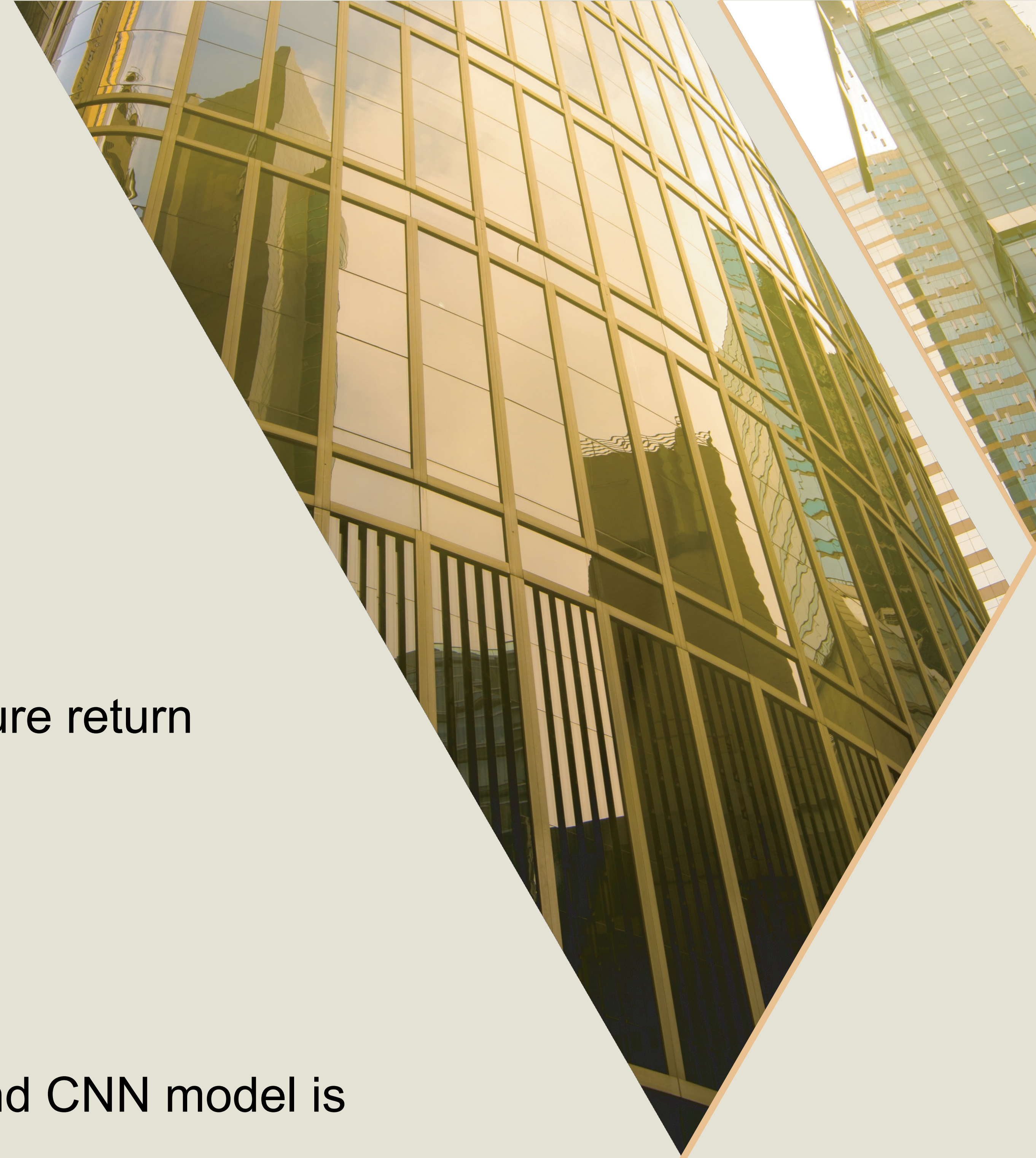
CNN is successful to extract predictive features and predict future return

Robust

Certain parameters and techniques help alleviate overfitting, and CNN model is robust to specification variations

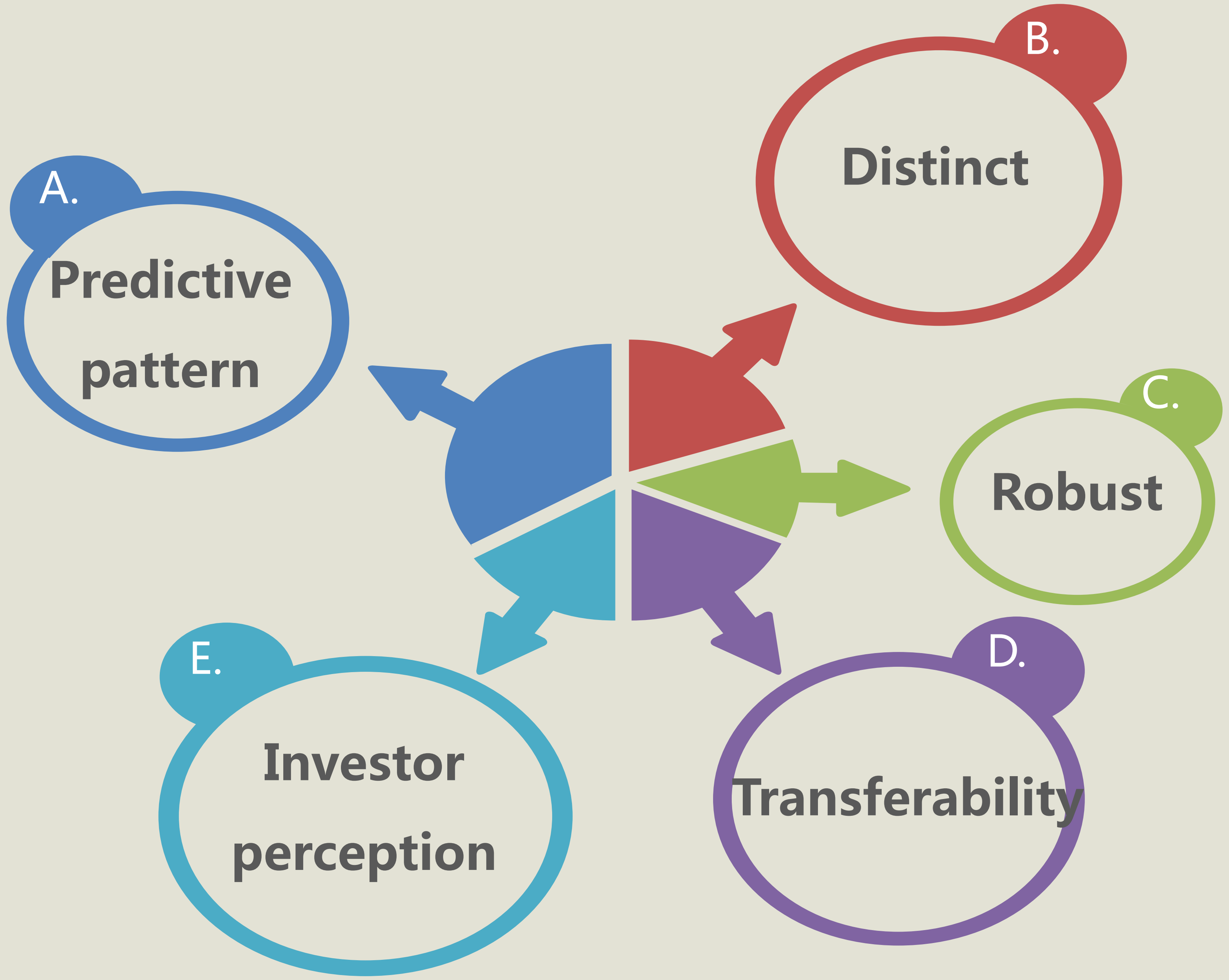
Investor perception

From Grad-CAM, the CNN can deal with the 2-D image like human



Conclusion and future direction

Future Direction



Reconsider our motivation, we still need to verify

1. Whether the prediction of CNN is similar to other factors or models
2. Whether the CNN trained from certain dataset can be transfer to another dataset
3. The accuracy is not enough to measure the predict ability of CNN, we need to construct the investment portfolio and calculate the Sharpe Ratio.

用LSTM比较，提取最后一层layer的feature 去看最后一个维度60

Thank you