RO1

FYP Proposal

Machine Learning Based All Weather U.S. Stock Portfolio Management System

by

CHO Hangsun, YANG Wenting, TAM Ching Lung, Ferdy

RO1

Advised by

Prof. David Rossiter

Submitted in partial fulfillment of the requirements for COMP 4981

in the

Department of Computer Science

The Hong Kong University of Science and Technology

2022-2023

Date of submission: 2022-09-16

Table of Contents

1. Introduction

1.1. Overview

Ever since the emergence of the first financial institutions, the financial market has continued to evolve. Nowadays a lot of stakeholders participate in the financial market to conduct transactions. Widely ranging from basic products such as stocks and commodities to exotic derivatives, there are nearly infinite variations of financial products that are being traded in the financial market. People have always tried to make revenue in the financial market, and the strategies have continued to evolve. Before the 4th industrial revolution, traders mostly focused on fundamentals. Nowadays, traders depend highly on computers. Algorithmic trading is a method widely used by traders, which allows them to use pre-programmed rules with parameters such as time, price, and volume, to automate the trading process. Traders can exploit the reliability, robustness, and speed of computer programs when executing orders. Moreover, irrational decisions and human mistakes made by traders can be removed.

Different algorithmic trading strategies manipulate different data and technology. Arbitrage, one of the widely known strategies, takes advantage of a price difference among different markets. Different products following the same basket of securities, or even the same products, can have different values in different markets due to market inefficiencies. In this case, the profit is the market price difference. Another notable strategy is named momentum strategy. This strategy assumes that the price of a financial product tends to follow past trends.

Machine learning is used for prediction of unknown data, after training our model with training data. It has a great variety of fields of applications, like speech recognition and food classification. Some could even perform predictions better than humans given their ability to process vast amounts of data. With such ability, they have been applied for performing financial predictions in order to trade according to its expected value so as to make money.

However, there is still no golden rule in trading under volatile market conditions and macroeconomic factors, such as the COVID-19 outbreak, Russia-Ukraine War, and high inflation rates. This is because trading strategies perform totally differently under different market conditions.

We plan to use four different strategies mainly on stocks and integrate them together to provide a consistent return with risk as low as possible even under volatile market conditions. The four different strategies that we will focus on are: the multi-factor equity model, NLP-based strategy which analyzes the SEC reports, pairs trading strategy on baskets of stocks using clustering, and rank-based sentiment analysis on the internet. These four strategies will be integrated together using Harry Markowitz's modern portfolio theory in order to minimize the risk. We will also build a platform that traders can use after training the model with combined strategies.

1.2. Objectives

The goal of this project is to create a portfolio management system that beats the market to hedge our asset risks and potentially gain profits, through prediction using machine learning models and portfolio management using portfolio optimization methods. To improve upon previous projects, we will work on some newer trading strategies, using portfolio optimization, and ensemble learning, and perform more comprehensive testing.

There are several stages for developing the portfolio management system:

1. Data Collection: survey, select and retrieve relevant price, textual data
2. Data Storage: store it in a database or local cache
3. Data Preprocessing: process relevant indicators from the data storage
4. Financial Prediction: ensemble and predict the asset expected returns through different prediction strategies
5. Risk Quantification: estimate the asset expected risks through risk models

6. Portfolio Optimization: generate an optimal portfolio for multiple assets given their expected returns and risks

7. Backtesting: test and analyze portfolio management system historical returns, then compare with benchmarks

8. User Interface: a platform for users to select stocks, strategies, time periods and visualize the backtesting results

1.3. Literature Survey

1.3.1 Portfolio Management using Machine Learning

In the previous FYP proposal [1], Chih-yu and Matthew's report combined trading strategies to beat the S&P 500 Index. Compared with the market's annual return 9.85% and Maximum Drawdown (MDD) 55.1%, their machine learning algorithm beats the market index benchmark 0.14% with far less Maximum Drawdown which is only 7.8%. Our report will use a similar methodology but with different machine learning methods and more asset classes like stock, bond, commodity,etc if possible in order to achieve higher annualized return and lower MDD.

1.3.2 Pairs Trading via Unsupervised Learning

Mean reverting property is one of the core in pairs trading. Pairs trading is based on an assumption that the spread between highly correlated assets will revert to its mean eventually. Even if there exists a significant amount of difference from the mean spread, it is important when to enter and when to exit. This paper [2] took a mathematical approach to figure out the timing to start and liquidate the position subjected to transaction costs. However, this paper does not discuss which factors to consider when choosing the pair.

This paper [3] uses unsupervised learning to find pairs of companies based on firm characteristics and price information. Their strategy showed annualized mean return of 24.8% and a sharpe ratio 2.69 on U.S. stocks during the period from 1980 to 2020. They used 48

price momentums as price information and 78 different firm characteristics such as beta, bid-ask spread, and return on equity. Three clustering methods: k-means clustering, DBSCAN, and agglomerative clustering, were used.

1.3.3 Multi-factor Stock Selection model with Machine Learning

Multi-factor models are one of the most important active investments in quantitative asset management, which explain returns on individual stocks with different factors. There are two types of Multi-factor models: Arbitrage Pricing Theory (APT) models which focus on the linear relationship between the expected returns on individual stocks and macroeconomic variables[4], and Fama-French type models which use companies' fundamental factors to do the return prediction[5]. The report focuses on the latter one. The multi-factor model in this report is defined in Appendix C formula (1).

Compared to using traditional statistical methods to explain Muti-factors' influence on stock expected return, machine learning methods which usually have advantages in analyzing high-dimensional non-linear data are proven to be more effective especially under current big data era[6]. Previous machine learning methods used in the Multi-factor model include Support Vector Machine (SVM), Neural Network (NN) and Gradient Boosting Decision Tree (GBDT),etc[7].

However, recent research demonstrates that in terms of the performance in stock prediction tasks, deep learning methods like LSTM are better than SVM, random forest, etc[8]. At the same time, the prediction capability of the Gated Recurrent Unit (GRU) is better than that of LSTM[9]. Besides, in order to improve the performance of deep learning methods, weight optimization algorithms are usually used to solve parameter selection problems. The essay, which proposes a multi-factor stock selection model combining a GRU and Cuckoo Search (CS) optimization, performs quite well, achieving 13.13% excessive return on Chinese market index CSI 300 with MDD -17.38%[10]. However, the essay mainly focuses on the Chinese stock market and we want to apply it to the US stock market.

1.3.4 Price Prediction by Rank-based Sentiment Analysis on the Internet

The efficient market hypothesis states that asset prices reflect all available information. So, the more information we have, the better we can predict the market prices.

Social media is highly popular recently for discussion of cryptocurrency like bitcoin and ethereum. Many papers and trading strategies started using social media posts sentiment for predicting cryptocurrency prices. This paper combines reddit post sentiment and its importance to model crypto prices [11]. One limitation of this project is that it is only tested on crypto. A past ug project aimed to find the most trending symbol in Reddit popularity symbols, then analyze its sentiment and trading them [12]. However, it is limited to only the top few symbols. News and google trends are also found to have great correlation with crypto prices [13]. However, one limitation of this project is that it is only tested on crypto.

Deep learning is one of most commonly used models for time series predictions. A previous project compared LSTM, CNN and GRU for financial prediction, where LSTM performed the best [14]. So, we could make use of LSTM for our time series model.

So, there is still much space for engineering more features along with sentiment and testing these strategies on stocks. In this paper, the internet sentiment analysis strategy will analyze sentiment from these sources to get a better view of the market for prediction.

1.3.5 Natural Language Processing (NLP)

NLP has been proven able to parse the complexities of text hundred times faster than humans [15]. It is not only limited to some general topics but could be extended to help humans analyze business and financial news, not to mention the complicated industry jargon, numbers, currencies, and product names that requires prior domain knowledge. Based on a study of forty five research papers [16], in general, financial analysts' predictions always include an optimistic bias, resulting in a higher prediction value of the prices. Using NLP will help to eliminate this optimistic bias.

2. Methodology

2.1. Design

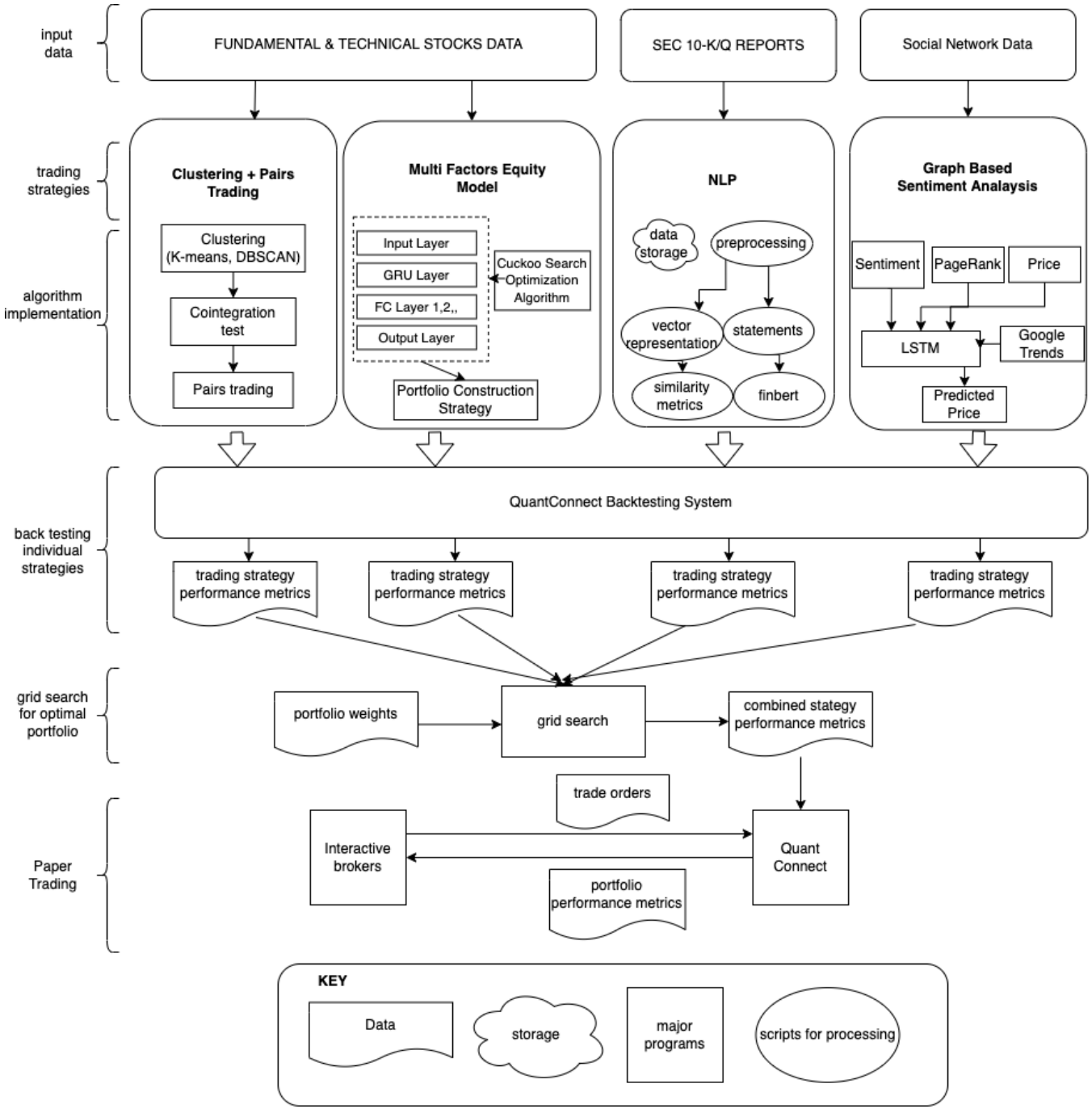

Figure 1.

2.1.1 Pairs Trading on Baskets of Stocks using Clustering

The first step will be finding a correlation between stocks and building baskets of them. Stock data, including but not limited to technical indicators, chart data, and fundamental data will be collected. After that, baskets of stocks will be built in a few different ways. For instance, stocks can be grouped by their fundamental data such as ROE and EBITDA. On the other hand, stocks can be grouped by technical indicators such as moving average and volatility. Grouping the stocks based on these factors will be done using clustering techniques such as K-Means clustering or DBSCAN. Then, cointegration and correlation tests will be done to find out which type of grouping shows the best. After we have baskets of stocks, pairs trading will be conducted.

2.1.2 Price Prediction by Rank-based Sentiment Analysis on the Internet

As discussed in the literature survey, Reddit, Twitter, and google trend data are quite useful for sentiment and price prediction. So to analyze them for price prediction, here are the steps:

1. Collect data from social media Reddit and Twitter, and google trend stock prices
2. Analyze text rank using its author rank, sentiment, and date
3. Train model to predict price using the sentiment, prices, and indicators

2.1.3 NLP Sentiment Analysis - Integrating Sentiment Analysis and Textual Similarity

Imagine we have 2 movie reviewers reviewing the same movie. Although they could use different wordings, if both of them liked the movie, then the sentiment of the meaning will be similar. Using a text similarity approach will result in low similarity for both reviews, therefore, it would be better not to use similarity measures only but leverage a sentiment analysis model that could confirm the sentiment of words contained in the reviews. Here, we are going to fine tune and use a deep learning model called Finbert, a Bidirectional Encoder Representation of Transformers (BERT) that has been trained on SEC-10K filings that can better analyze financial terms and jargons compared to the original BERT. Therefore, we

hope that this model can better classify if the word belongs to a particular sentiment class (i.e., negative, positive, uncertainty).

Some examples to visualize these approaches would be like a company having financial distress. As a result, their report will be dominantly filled with negative words. Sharp changes in the similarity of the report compared to the previous report could be also used as an indication that major changes are going to affect the stock performance, and the sentiment analysis from the FinBERT model will confirm the direction of the changes.

2.1.5 Multi-factor Equity Model

For this model, there are three steps. First, we build our own factors database including both technical and fundamental data. Then, we feed our data to our stock selection model and train it, which is a Gated Recurrent Unit (GRU) neural network model optimized with the Cuckoo Search (CS) algorithm. Thirdly, we design our trading strategies based on the above multi-factor stock selection model and do the back testing.

2.1.6 Future Roadmap

Going further from the project, we are planning to add some additional features to create a seamless experience for the user to evaluate our algorithm.

1. Add User interface

   We are going to visualize the data in our portfolio management system by creating the Frontend, using React.js. and Backend, using Fast API.

2. Database for persisting user portfolio performance

   We are going to persist user portfolio performance throughout the years using MySQL as RDBMS.

2.2. Implementation

2.2.1 Pairs trading on baskets of stocks using clustering

1. Collect data from TradingView - Collect technical indicators such as moving average, stochastic, and MACD, and also fundamental data such as ROE and EBITDA.
2. Use clustering methods to build a cluster of stocks based on the collected technical data.
3. Use clustering methods to build a cluster of stocks based on the collected fundamental data
4. Perform cointegration test on clusters
5. Run pairs trading on a cluster of stocks which showed good results on the cointegration test. Trade at most n-stocks $S_{(1)}$, $S_{(2)}$, ..., $S_{(n)}$ at the same time, yielding a portfolio value $X_t = \alpha S_{(1)} + \beta S(2) + \ldots$ where time $t \geq 0$, and weights $\alpha, \beta, \ldots$ can be either positive (long) or negative (short).

2.2.2 Multi-factors Equity Model

We apply the innovative CS-GRU stock selection model to the US Stock market. Here are the implementation details:

First, we find all the stocks in the S&P 500 and build a factor database including these stocks' 16 financial factors and 9 technical factors shown in Appendix B figure 1 and figure 2 respectively.

We then construct our CS-GRU stock selection model. In order to simplify the question, we transform our stock selection model into a classification model which contains two categories: rise and fall or 1 and 0. The top 25% of individual stock's next month's excess return are classified as "rise", while the bottom 25% ones are classified as "fall". We discard the data in the middle. The excess return for an individual stock is defined in Appendix C formula (2).

We feed our data to the CS-GRU model and do the training. The output of the model should be the individual stock's next month rising possibility.

2.2.3 Price Prediction by Rank-based Sentiment Analysis on the Internet

Here are more detailed steps for implementing the strategy:

1. Collect data from news using selenium, social media using Reddit, Twitter API, google trend API, and prices from yfinance by searching its relevant keywords
2. Analyze social media influencer author rank by analyzing retweets, mentions, likes, shares, comments, and subscribers
3. Analyze recent tweet sentiment using finbert on its title and content
4. Analyze tweet rank using its author rank, sentiment, and date
5. Setup dataframe with sentiment score, price, and other indicators
6. Train and predict expected price using LSTM model

2.2.4 Stock Selection using NLP

Here are the details for implementing the strategy:

1. Indicate the tickers and central key index of the stocks that we are going to analyze.
2. Scrape all 10-K reports using beautiful soup.
3. Use requests package to download each document.
4. Preprocess each document, by removing stop words, html tags, stemming, and lowercasing the entire text.
5. Generate vector representation for each 10-K report using Term Frequency-Inverse Document Frequency (TF-IDF) and use Jaccard similarity to compute similarity between documents.

Trading Strategy

1. Everytime there is a new report, compute similarity between documents.
2. For each 10-K report, compare the similarity score for the current 10-K report with the previous year 10-K report.
3. long the stock if similarity score differ below 10 percent, otherwise sample some sentences from both reports and do inference on the Fine-Tuned FinBERT model.
4. Use the sentiment from the FinBERT model to confirm the direction of the changes and adjust our weight for each stock in the portfolio accordingly.

2.3. Testing

During the development process, we will unit test each of our prediction strategies and backtest our portfolio management system with historical data.

2.3.1 Pairs Trading on Baskets of Stocks using Clustering

After we build baskets of stocks using clustering techniques, cointegration testing will be done on each basket. Cointegration test is used to verify whether two or more non-stationary time series are integrated together, so that in the long term, they do not diverge significantly from the equilibrium. Cointegration test is crucial in pairs trading because pairs trading strategy exploits the mean-reverting property of the prices of two or more different assets. One of the well known cointegration test strategies such as Johansen test or Augmented Dickey-Fuller test will be used.

Trading strategy will be backtested against an index (i.e. S&P 500) to compare the return and the risk. One way of doing so is comparing the compound annual growth rate (CAGR) and maximum drawdown (MDD) of the portfolio following pairs trading strategy with the portfolio following S&P 500.

2.3.3 Price Prediction by Rank based Sentiment Analysis on the Internet

Before we pass the predicted prices into the portfolio optimization layer, we could perform some testing on the model first to assess its accuracy and performance. For accuracy, we will compare the actual and predicted prices and measure its error metrics like mean square error, mean absolute error, and r-squared. For performance, we will have a preliminary trading strategy using the sentiment and the predicted price as the factors. When the expected return is positive, then buy, else sell. Then, we measure the overall return by subtracting the buy and sell execution prices.

2.3.4 Stock Selection Using NLP

We will run unit testing to evaluate every preprocessing step that we use to produce the vector representation of every 10-K document. Then, we will store the vector representation

in our database, so we could compare it with future reports. For every new report, we are going to find similarity with it's previous report and test the similarity with the FinBERT model as we mention in section 2.2.4. Then, we are going to do integration testing with the quantconnect backtesting system. We will leverage the QuantConnect log system to do monitoring. Lastly, we will test that our algorithm will only run if and only if the company releases its yearly / quarterly report.

2.3.5 Multi-factor Equity Model

Our trading strategy for the multi-factor model is that we construct the portfolio by buying stocks with top 10 rise possibilities predicted by the stock selection model. During every adjustment time, we change our position to new selected 10 stocks. We can do our back testing using this trading strategy in some quant platforms like QuantConnect. The portfolio will be adjusted on a monthly basis.

2.3.6 Portfolio Management System

We will integrate our different strategies for trading using portfolio optimization methods. When each of the expected returns for the top 10 stocks is positive, add to portfolio optimization, then buy them with their optimized weights multiply the initial capital. We will perform portfolio optimization every selected period, then multiply the returns of that period minus commission and slippage for backtesting. We will evaluate the performance using metrics like sharpe ratio, calmar ratio, and max drawdown. We will try different time frames, time periods, especially crisis and high inflation periods, and stocks to thoroughly test the performance.

2.4. Evaluation

For this part, we want to answer several questions in our later stage:

1. Is the database complete and up to date?
2. How well did each strategy perform? Do we beat the previous work?
3. How comprehensive was the testing and visualization?
4. How well did the portfolio management system perform compared with benchmark?

5. How well can the system handle special periods like crises and high inflation?

3. Project Planning

3.1. Distribution of Work

Task	Hangsun	Max	Wendy	Ferdy
Do the Literature Survey	●	●	●	●
Find Data Source	●	●	●	●
Data Preprocessing	●	●	●	●
Design Pairs Trading using Clustering	●	○	○	○
Collect technical data and fundamental data	●	○	●	○
Create baskets of stocks using clustering	●	○	○	○
Perform cointegration test on baskets	●	○	○	○
Run pairs trading on the best baskets	●	○	○	○
Design the Rank-based Sentiment Strategy	○	●	○	○
Collect social media data	○	●	○	○
Develop pagerank mechanism	○	●	○	○
Train sentiment and price models	○	●	○	○
Test the rank based sentiment strategy	○	●	○	○
Design the GRU-CS algorithm	○	○	●	○
Train multi-factor stock selection model	○	○	●	○
Test trading strategies on stock data	○	○	●	○
Collect SEC 10-K reports	○	○	○	●
Develop similarity measure between 10K	○	○	○	●
Fine tune FinBERT model	○	○	○	●
Test the Platform	○	○	○	●
Write the reports	●	●	●	●
Work on Project Video	●	●	●	●
Prepare the Final Presentation	●	●	●	●
Design the Project Poster	○	○	●	●

3.2. GANTT Chart

Task	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar	Apr
Do the Literature Survey										
Find Data Source										
Data Preprocessing										
Design Pairs Trading using Clustering										
Collect technical data and fundamental data										
Create baskets of stocks using clustering										
Perform cointegration test on baskets										
Run pairs trading on the best baskets										
Design the Rank-based Sentiment Strategy										
Collect social media data										
Develop pagerank mechanism										
Train sentiment and price models										
Test the rank based sentiment strategy										
Design the GRU-CS algorithm										
Train multi-factor stock selection model										
Test trading strategies on stock data										
Collect SEC 10-K reports										
Develop similarity measure between 10K										
Fine tune FinBERT model										
Test the Platform										
Write the reports										
Work on Project Video										
Prepare the Final Presentation										
Design the Project Poster										

4. Required Hardware & Software

4.1. Hardware

Item	Specification (Minimum)
RAM	8GB
HDD	128 GB
Processor	4 x 1.6 GHz CPU
GPU	2x Nvidia RTX 2080Ti
Web Server	2 x 1.6 GHz CPU, 3.5 GB RAM, 40GB HDD

4.2. Software

Item	Version	Specification
Development OS	MacOS Catalina 10.15.6	Environment for development
Git	2.23.0 or after	Version control
Miniconda	4.8.4 or after	Package control
Finbert	ProsusAI	base model for nlp fine tuning
QuantConnect	4.0.1 or after	backtesting platfrom
Python	3.9 or after	programming language
Tensorflow	2.0 or after	machine learning library

Keras	2.10 or after	machine learning library
VSCode	latest	IDE
ta-lib	0.4.25	Technical Analysis Library

5. References

[1] C. Lee, M. J. UY, D. Rossiter, "Using machine learning and algorithmic trading to beat the U.s. stock market index," *Edu.hk*, 2021. [Online]. Available: https://cse.hkust.edu.hk/~rossiter/fyp/RO4_FYP_final_report_202021.pdf. [Accessed: 16-Sep-2022].

[2] X. Li, and T. Leung, "Optional Mean Reversion Trading with Transaction Costs and Stop-Loss Exit", *International Journal of Theoretical and Applied Finance*, vol. 18, no. 3, Apr. 2015. [Online]. Available: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2222196. [Accessed: 15-Sep-2022].

[3] Z. He, C. Han, and A. J. W. Toh, "Pairs trading via unsupervised learning," *SSRN Electron. J.*, 2021. [Online]. Available: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3835692. [Accessed: 15-Sep-2022].

[4] S. A. Ross, "The arbitrage theory of capital asset pricing," *J. Econ. Theory*, vol. 13, no. 3, pp. 341–360, 1976. Available: https://doi.org/10.1016/0022-0531(76)90046-6. [Accessed: 16-Sep-2022].

[5] E. F. Fama and K. R. French, "The cross-section of expected stock returns," *J. Finance*, vol. 47, no. 2, pp. 427–465, 1992. Available: https://doi.org/10.1111/j.1540-6261.1992.tb04398.x. [Accessed: 16-Sep-2022].

[6] Q. Feng, X. Sun, J. Hao, and J. Li, "Predictability dynamics of multifactor-influenced installed capacity: A perspective of country clustering," *Energy (Oxf.)*, vol. 214, no. 118831, p. 118831, 2021. [Accessed: 16-Sep-2022].

[7] S. Sugitomo and M. Shotaro, "Fundamental factor models using machine learning," *SSRN Electron. J.*, 2018. Available at SSRN: https://ssrn.com/abstract=3322187 or http://dx.doi.org/10.2139/ssrn.3322187. [Accessed: 16-Sep-2022].

[8] K. Sai and J. Lakshminarayanan, “A comparative study of SVM and LSTM deep learning algorithms for stock market prediction,” *Ceur-ws.org*. [Online]. Available: http://ceur-ws.org/Vol-2563/aics_41.pdf. [Accessed: 16-Sep-2022].

[9] Lawi A. Implementation of Long Short-Term Memory and Gated Recurrent Units on Grouped Time-Series Data to Predict Stock Prices Accurately, 2021. Available: https://doi.org/10.21203/rs.3.rs-1057875/v1. [Accessed: 16-Sep-2022].

[10] A. Lawi, H. Mesra, and S. Amir, “Implementation of Long Short-Term Memory and gated recurrent units on grouped time-series data to predict stock prices accurately,” *Research Square*, 2021. Available:https://doi.org/10.3390/math10040566. [Accessed: 16-Sep-2022].

[11] S. Wooley, A. Edmonds, A. Bagavathi, and S. Krishnan, “Extracting cryptocurrency price movements from the reddit network sentiment,” *18th IEEE International Conference On Machine Learning And Applications (ICMLA)*, 2019. [Accessed: 16-Sep-2022].

[12] K. Ngan and D. Rossiter, “Popularity-based trading strategy from reddit posts,” *Edu.hk*, 2022. [Online]. Available: https://cse.hkust.edu.hk/~rossiter/independent_studies_projects/reddit_trading/reddit_trading.pdf. [Accessed: 11-Sep-2022].

[13] J. Abraham, D. Higdon, J. Nelson, J. Ibarra, and J. Nelson, “Cryptocurrency price prediction using tweet volumes and cryptocurrency price prediction using tweet volumes and sentiment analysis sentiment analysis,” *Smu.edu*, 2018. [Online]. Available: https://scholar.smu.edu/cgi/viewcontent.cgi?article=1039&context=datasciencereview . [Accessed: 11-Sep-2022].

[14] H. Kung-Hsiang, K. Ziwon, and H. Yao-Chieh, “Bitcoin Price Prediction with Deep Learning and Social Trend,” *hkust.edu.hk*, 2018. [Online]. Available: http://www.cs.ust.hk/~dlee/fyp/Password_Only/2017/27_DL1_Final-R.docx . [Accessed: 15-Sep-2022].

[15] T. Mayor, “Why finance is deploying natural language processing,” *MIT Sloan*, 2020. [Online]. Available: https://mitsloan.mit.edu/ideas-made-to-matter/why-finance-deploying-natural-language-processing. [Accessed: 16-Sep-2022].

[16] A. Stotz, “Are financial analysts’ earnings forecasts accurate?,” *Become a Better Investor*, 2016. [Online]. Available: https://becomeabetterinvestor.net/are-financial-analysts-earnings-forecasts-accurate/. [Accessed: 16-Sep-2022].

6. Appendix A: Meeting Minutes

6.1 Minutes of the 1st Project Meeting

1. Arrangement
    - Date: 2022-04-23
    - Time: 17:00-18:00
    - Place: Zoom
    - Present: All team members
    - Absent: None
    - Recorder: Hangsun
2. Discussion items
    - Finalized the priority project title
    - Discussed some direction on which market and strategy to use
    - Discussed on whether we should focus more on research (strategy and backtesting) or building a tangible platform that the users can use.
    - Suggested to send an email to professor to narrow down the topic.
3. To do list
    - Send an email to supervisor (Dr.Rossiter) as we will be proposing our own topic related to “Trading Systems for Financial Gain”
    - Research more on which strategy and asset we will use
4. Next meeting
    - The next meeting will be after we contact the supervisor

6.2 Minutes of the 2nd Project Meeting

5. Arrangement
    - Date: 2022-07-11
    - Time: 21:00-22:30
    - Place: Zoom
    - Present: All team members
    - Absent: None
    - Recorder: Ferdy
6. Discussion items
    - Data sources
    - Useful indicators and quant
    - Baseline benchmark
    - Backtesting tools
    - Our strategies
        i. Creditable financial report text mining
        ii. Social media pagerank sentiment
        iii. Multi factor model
        iv. Reinforcement learning
    - Combine strategies by portfolio optimization
7. To do list
    - Continue to read research papers on our strategies
8. Next meeting
    - The next meeting will be 8pm on August 22, 2022 via Zoom

6.3 Minutes of the 3rd Project Meeting

1. Arrangement
    - Date: 2022-08-22
    - Time: 20:00-21:00
    - Place: Zoom
    - Present: all and professor
    - Absent: none
    - Recorder: TAM Ching Lung

2. Discussion items
    - Introduction of our strategies to professor
    - Data
        - Should mention the data source
        - Can try looking at different sources like HKUST Bloomberg terminal, TradingView, business school, interactive broker
    - Design
        - Suggest system strategy to fit different assets
    - Testing
        - Compare with benchmarks like risk-free rate and all cash
        - Better presentation of backtesting metrics and comparison
        - Test with custom random data
    - Suggested ug, msc, fyp projects paper by professor
    - Budget 3000 shared between 3 fyp groups at 1k per group
    - Learning from previous groups
        - Too ambitious objectives
        - Time management
        - Meeting and making progress every week
        - Start early
        - Produce original content
        - Well defined task distribution
        - Reference negative knowledge

3. To do list
    - Read professor suggested paper readings
    - Start planning the work division of writing the proposal report

4. Next meeting
    - The next meeting will be 10pm on September 3, 2022 via Zoom

6.4 Minutes of the 4th Project Meeting

1. Arrangement
    - Date: 2022-09-03
    - Time: 22:00-23:00
    - Place: Zoom
    - Present: all team members
    - Absent: none
    - Recorder: Yang Wenting

2. Discussion items
    - Distribute work write proposal
    - Confirm our directions and discuss our findings
        - First focused on one asset class and then we explore others
        - For Multi-factor model, we can have a discussion with other groups with similar topics

3. To do list
    - Hangsun - Overview, 1st minute, pairs-trading & portfolio theory
    - Wendy - Literature review, 4th minute, Multi-factor model
    - Ferdy - Hardware and Software, 2nd minute, NLP and System & application design
    - Max - objective, 3rd minute, graph popularity-based sentiment analysis
    - Shared - Project planning, Methodology
    - Personal DDL for proposal: Sep 11

4. Next meeting
    - The next meeting will be 8pm on September 12, 2022 via Zoom

7. Appendix B: List of Figures

Factor Name	Factor Abbreviation	Factor Description
Circulation market value	CMC	closing price of individual shares * number of circulating shares of individual shares
P/E ratio	PE	(closing price of individual shares on the specified trading date * total share capital of the company as of that day)/net profit attributable to shareholders of the parent company
Price to book ratio	PB	(closing price of individual shares on the specified trading date * total share capital of the company as of that day)/equity attributable to shareholders of the parent company
Price to sales ratio	PS	(closing price of individual shares on the specified trading date * total share capital of the company as of that day)/total operating revenue
Increase rate of business revenue annulus	IRA	((current operating income value—previous operating income value)/previous operating income value) * 100%
Increase rate of net profit attributable to shareholders of the parent company annulus	INPTSA	((net profit attributable to shareholders of the parent company in the current period—net profit attributable to shareholders of the parent company in the previous period)/net profit attributable to shareholders of the parent company in the previous period) * 100%
Increase rate of net profit annulus	INPA	((net profit value of the current period—net profit value of the previous period)/net profit value of the previous period) * 100%
Return on equity	ROE	(net profit attributable to shareholders of the parent company * 2)/(net assets attributable to shareholders of the parent company at the beginning of the period and net assets attributable to shareholders of the parent company at the end of the period)
Return on assets	ROA	(net profit * 2)/(total assets at the beginning and total assets at the end of the period)
Net profit margin on sales	NPR	net profit/operating income
Earnings per share	EPS	net profit/closing share capital
Retained earnings per share	RPPS	retained profit/closing share capital
Net asset value per share	NAPS	net assets/closing share capital attributable to shareholders of the parent company
Capital reserve per share	CRFPS	capital reserve/closing share capital
Net cash flow per share	CPS	net cash flow/ending equity
Quick ratio	QR	(total current assets—inventory)/total current liabilities

Figure 2. Chosen Financial Factors

Factor Name	Factor Abbreviation	Factor Description
20 day annualized return variance	V20	Variance of annualized returns of individual stocks in the last 20 days
20 day return kurtosis	K20	20 day kurtosis of individual stock returns
10 day average turnover rate	VOL10	Average turnover rate of individual stocks in the last 10 days
Moving average of 12 day trading volume	VEMA12	Moving average of 12 day trading volume of individual stocks
Standard deviation of 20 day trading volume	VSTD20	Standard deviation of trading volume of individual stocks in the last 20 days
Buying and selling momentum rate	AR	(sum of 26 days (highest price of the day—opening price of the day)/sum of 26 days (opening price of the day—lowest price of the day) * 100%
Willingness to buy rate	BR	(sum of 26 days (the highest price of the day - yesterday's closing price)/sum of 26 days (yesterday's closing price—the lowest price of the day) * 100%
Arron up	AU	((calculation period days—days after the highest price)/calculation period days) * 100%
Arron down	AD	((calculation period days—days after the lowest price)/calculation period days) * 100%

Figure 3. Chosen Technical Factors

8. Appendix C: List of Formulas

$$R_{it} = \sum_{n=1}^{k} X_{int} f_{nt} + \epsilon_{it} \quad \text{— (1)}$$

in which R_{it} means company i's equity return in t period, X_{int} is risk exposure of factor n for company i in t period, f_{nt} is factor n return in t period, and ϵ_{it} is the special return.

$$r_{i,t} = \frac{P^{l}_{i,t+1} - P^{f}_{i,t+1}}{P^{f}_{i,t+1}} - \frac{P^{l}_{500,t+1} - P^{f}_{500,t+1}}{P^{f}_{500,t+1}} \quad \text{— (2)}$$

where $r_{i,t}$ is the excess return of next month of month t for individual stock i, $P^{l}_{i,t+1}$ is the the last trading day price of month t+1 for individual stock i, $P^{f}_{i,t+1}$ is the the first trading day price of month t+1 for individual stock, $P^{l}_{500,t+1}$ is the the last trading day price of month t+1 for S&P 500, $P^{f}_{500,t+1}$ is the first trading day price of month t+1 for S&P 500.