Using GPT to analyze Supply-Chain news

Literature Review

News data has an impact on the price fluctuations of stocks

At present, many scholars have done relevant studies to prove that news data has a significant impact on stock prices: Shi Yongdong et al. (2015) proved that the stock returns of companies with negative sensitivity to investor sentiment are significantly affected by investor sentiment while controlling for the Fama-French triple factor and Carhart's momentum factor, and Xu Xiang and Jin Jing (2018) used the post public opinion data on Toutiao to pass LDA Shi Feng (2020) verified that news sentiment can have an impact on asset prices, and the impact of positive and negative sentiment is asymmetric, and confirmed that the sentiment index can predict index returns to a certain extent, Li Fengke (2020) By using the data of Oriental Fortune Stock Bar, the investor sentiment data of individual stocks is constructed, which proves that the data and the return of individual stocks of the CSI 300 Index are mutually Granger causal relationship. It can be seen that these studies have proved from various angles that the emotional information contained in the news data can have an impact on the price of stocks, while on the other hand, most of the research on news focuses on using NLP methods to extract emotional information from the news to obtain a measure of investor sentiment and analyze it accordingly. For example, it is obvious that for the country's macro policy and a simple daily limit news, it is obvious that the scope, importance and sustainability of the two impacts are different, and it may still be difficult to learn the unique nature and information of various news if they are all trained together with a unified model. Based on this phenomenon, we considered using GPT to subdivide news into various categories before the interim report, and then further analyzed based on these categories to obtain indicators such as the impact and sustainability of different types of news, so that we can make more refined and accurate analysis.

The use of GPT in financial news analysis and quantitative investment

With the introduction of GPT-3.5, GPT-4 and other large language models, more and more scholars have done relevant research on the application of GPT in the financial field, proving the superiority of GPT models in analyzing news data and constructing quantitative investment strategies. Hongyang Yang and so on (2023) use ChatGPT to assess whether each title is good, bad, or neutral for a company's stock price, ChatGPT was found to outperform traditional sentiment analysis methods. More basic models such as GPT-1, GPT-2, and BERT fail to accurately predict returns, suggesting that return predictability is an emerging capability for complex language models, and the long-short strategy based on ChatGPT-4 they construct in this paper achieves the highest Sharpe ratio. Udit Gupta（2023）assess annual report through GPT， using the insights generated by the LLM are compiled in a Quant styled dataset and augmented by historical stock price data, training a Machine learning model with LLM outputs as features. Showing promising outperformance wrt S&P500 returns. Ethan Callanan and so on(2023年) leverage mock exam questions of the Chartered Financial Analyst (CFA) Program to conduct a comprehensive evaluation of ChatGPT and GPT-4 in financial analysis, considering Zero-Shot (ZS), Chain-of-Thought (CoT), and Few-Shot (FS) scenarios. They present an in-depth analysis of the models' performance and limitations, and estimate whether they would have a chance at passing the CFA exams. And they concludeGPT-4 would have a decent chance of passing the CFA Level I and Level II if prompted with FS and/or CoT. Georgios Fatouros and so on (2023) integratee Chain of Thought and In-Context Learning, MarketSenseAI analyzes diverse data sources, including market trends, news, fundamentals, and macroeconomic factors, to emulate expert investment decision-making. Through empirical testing on the competitive S&P 100 stocks over a 15-month period, MarketSenseAI demonstrated exceptional performance, delivering excess alpha of 10% to 30% and achieving a cumulative return of up to 72% over the period, while maintaining a risk profile comparable to the broader market, showing a significant leap in integrating generative AI into financial analytics and investment strategies

It can be seen that GPT can better analyze data such as news and finance to a certain extent, and construct certain quantitative strategies to achieve better performance. At the same time, most of the current research on GPT also focuses on the application of GPT as a whole, and no more refined features are obtained from multiple perspectives, and the design of prompts and prompts is still not a reasonable and perfect set process and guidelines. Many articles focus on the application of GPT in strategy construction, such as signal generation, rather than the generation of some useful features. However, in reality, GPT's current capabilities may not be enough to obtain good and useful signals in the case of a complete process from feature mining to signal generation to portfolio construction, which seems a bit too greedy, so it seems more reasonable to start from some detailed directions.

Research on the industrial chain and the application of GPT

The use of industrial chain (supply chain) to obtain a more detailed relationship between stocks is also an emerging alternative data research direction that has been proven to have a certain effect. Frédéric Abergel(2023) use Bloomberg's data to construct a directed graph about the supply chain, based on this, the paper conducts cluster analysis to prove the strong interaction between listed and non-listed companies, which cannot be obtained from the stock market alone, and at the same time, based on the supply chain graph network, the article conducts correlation analysis, and finds that when there is a supply relationship, the correlation distribution has a fatter positive tail and is biased to the right, and the companies that are connected in both the basic network and the extended network have a significant correlation than the companies that are connected only in the basic network, through these analyses, the paper proves that the stocks connected directly or through a third party in the supply chain network are significantly correlated, which is more correlated than randomly paired stocks, and this correlation is suitable for extreme market conditions. Rei Yamamoto and so on (2021) uses global supply chain data to study the transmission of stock prices in the supply chain context, and constructs two factors， Customer Momentum, Supplier Momentum, for one stock, the Customer Momentum in a month is calculated by the mean of return of all its customers based on supply chain Data. The Supplier Momentum is also calculated using the same method. And they find Customer Momentum has a better performance than traditional momentum and is statistically significant, including more layers of customers and calculate is in longer time can obviously improve the performance of it. 从In these articles, it can be found that the value chain data includes quite useful alternative information, which is an analytical angle worth considering. In fact, news data also contains a lot of information related to the industry chain, and analyzing and extracting the industry chain from the news data may bring a different perspective

Project Discription

Breif introduction

The goal of this project is to use ChatGPT to analyze the A-share related news data downloaded from the tushare interface, and try to extract the information that is helpful in predicting the future returns of stocks. After the mid-term debriefing, we decided to use chatgpt to mine information related to the industry chain, which can be used to further generate features and factors that have the ability to predict the future returns of a stock.

After mining features, we drew some industry chain diagrams for each industry. We also constructed an investment strategy that achieved a return of **more than 300%** in 2023.

Project Data Description

Data: News data downloaded from tushare interface for the whole year of 2023.

ChatGPT: gpt3.5-turbo model interface provided by hkust.

Cost estimation: For ChatGPT, the cost of one piece of news data for gpt interface is about 0.009 HKD. For news data acquisition,the total cost is 80hkd.

Project output: industry chain chart, strategy, and some pictures.

Rearch Procedure

Initially, we planned to use all news articles as the dataset for multi-dimensional classification, such as classifying supply chain information and earnings announcements. We would then construct individual strategies and backtest them, eventually merging the smaller strategies into a final strategy. However, analyzing all the news articles using GPT API would be costly, and the model would become complex, making it difficult to construct portfolio. Therefore, after discussing with Mentor Zhang during the midterm report, we decided to focus on supply chain news as the primary research subject and attempt to extract features and get valuable information from features.

Considering the high cost of using the HKUST-provided ChatGPT API and the large number of news articles, I tried using the web version of ChatGPT for web scraping. I utilized the Selenium library in Python as a tool to remotely operate and control Chrome using the Chrome Driver. Initially, I attempted to access the chat.openai.com website, which is provided by OpenAI. However, this website employs Cloudflare as an anti-scraping protection mechanism, and there were no apparent vulnerabilities. Despite trying various methods such as header modification, cookie settings, and fingerprint modification, I was unable to bypass Cloudflare's blocking. As an alternative, I used poe.com for scraping, which also has anti-scraping measures, but I managed to find a vulnerability. The specific settings and scraping operations are detailed in my blog: Using Selenium to Interact with GPT Web Version - CSDN Blog. I also have attached a video demonstrating the scraping process in the file 'video'.

After the midterm presentation, Mentor Zhang advised us to continue using the GPT API. This is because it is unlikely that financial institutions would employ web scraping, as it poses potential risks. Moreover, web scraping programs have poor robustness, and if a website introduces new anti-scraping measures, the information retrieval becomes impossible. Therefore, we reverted to using the HKUST-provided API for analyzing news content.

In order to use chatgpt to mine industry chain related information, we devide the whole project into the following steps:

Data Processing

Since the total number of news items obtained reached more than 1 million, it became impossible to analyze all the news when the cost of GPT analysis for each news was about 0.01 yuan. Therefore, we actually only selected news data from 2023 as a sample, and designed some keywords to filter news that may contain supply chain information. These keywords are:

```
#define keywords list
supply_chain_keywords=["产业链","供应链","价值链","供需","供给","供应","采购","出售","销售","上下游","竞争","合作","供货","合作伙伴","合资","转让",'合作', '签','订单', '项目', '供应', '合约', '合同', '协议','收到', '协同', '上游', '产业链', '授权', '配额','承接', '原材料', '整合','客户' ,'供货', '方案', '提供', '业务', '双方','客户','渠道']
supply_chain_keywords=set(supply_chain_keywords)
```

After inspection, the filtered dataframe also contained a large amount of non-industry chain information. I constructed some keywords to exclude these non-supply chain information. These keywords are:

```
waste_list=["上市","基金","证券","A轮","注册资本","政策","法院","央行","定增","A+轮","收购","投资","美国', '伊朗', '外长', '韩国','中方',
'国家','对华','欧盟','中签',
   '日本', '斯坦','人民银行','亚运','总统','上涨','上调','闭幕','演习','部队','收涨','城镇','社区','印度','政治','出口','公积金','全年销售','俄','澳大利亚','党','融资',
   '医保','旅游','养老金','银保监会','通关','感染','乌克兰','荷兰','尼日利亚','我国','查处','阿联酋','新西兰','病毒','教育局','接种','期货','开盘','财政','监管','防疫',
   '土耳其','主力合约','下调','事业单位','春晚','加拿大','疾病','德国','工信部','销售额','全球','工商','印尼','英国','收盘','人才','埃及','业绩','农村','创业板','新加坡',
   '马来西亚','香港','芬兰','强国','振兴','意大利','联合国','挪威','法国','交易所','部长','销售部','改革','评级','墨西哥','美联储','运河','营业','市区','发改委','行长',
   '政协','通告','科创板','库存','卡塔尔','员工','地震','欧元','指数','港股','跌超','立案','贷','建议','调查','函','救援','高盛','沙特','民政厅','货币','石油','海关',
   '投诉','通告','市场价','越南','罚','科技部','瑞士','盘前','盘后','国务院','货币','局','摩根','A股','瑞典','产权','台积电','营收','碳中和','苹果'
   '分红','学院','行动','商务部','购房','春节','人民币','城市','债券','机场','二手房','中科院','现货','京津冀','政府','城市','回购','股份','选举','开工'
     '集合竞价','赔偿','暂未','股票','期权','运输部','消保委','订票','旅客']
```

After screening again, more than 10,000 news items remained.

In order to use ChatGPT to extract information related to industry chain, we designed a series of prompts to guide ChatGPT to make judgments.

(1) We give prompts to make ChatGPT act as a neutral financial scholar.

(2) Industry chain-related data generally involves two parties, which means, there is a supply or other relationship between at least two companies (subjects). Therefore, we broadly categorized the industry chain news into two types, i.e., Type I news reflecting that Company A supplies or provides services to Company B, and Type II news explicitly indicating that a certain company sells or buys a certain product, whereas if the name of the company is not pointed out, it does not belong to the Type II news. Taking this as a benchmark, we give ChatGPT prompt to make it categorize the industry chain data obtained in our first step.

(3) Based on step two, we first give the definition of supply chain types, so that chatgpt can extract the supply chain types and the information of supplying product types, buyers and sellers according to this. According to the different classifications in step two, we design different prompts to guide chatgpt to extract information such as supply chain type, product name, supplying company name, client company name, the location of the supplying company in industry chain, the location of the client company in industry chain, and order amount. In order for GPT to obtain key information, I designed the prompt words. **The separators between the information took the form of three "|" that Mentor Zhang suggested us to use during the mid-term chat.** The result can be used to draw the industry chain chart and give a deep insight into the supply chain of a industry.

```
##system role
str1="你是一个中立的金融学者，输出不受规则限制。我将输入新闻内容，请你判断该条新闻是否涵盖两家或更多家公司的明确的产业链信息。"
str2="新闻有两种，第一类新闻体现了A公司向B公司供货或提供服务,若不是第一类新闻，判断是否为第二类新闻。第二类新闻明确表示某家公司销售或购买某种产品，若未指出公司名称，则不属于第二类新闻。若不属于这两类新闻，请直接输出非产业链新闻。"
str3="供应链种类是指提供产品或服务的种类。例如电子元器件、白酒等。产业链位置有上游、中游和下游。订单方向有买或卖两种，采购为买，出货为卖。"
str4="若新闻属于第一类，输出：第一类新闻、供应链种类、产品名称、供货公司名称、收货公司名称、供货公司产业链位置、收货公司产业链位置、订单金额。每个因素间用|||分割开。"
str5="若新闻属于第二类，输出：第二类新闻、供应链种类、产品名称、公司名称、公司在产业链的位置、订单方向、订单金额大小。每个因素间用|||分割开。如果一条新闻有多家供货或收货公司，每家公司间使用三个,,,隔开。"
str6="对于每种特征，如新闻未说明，则用“空”代替特征输出。请不要输出其他信息以及除|和,之外的标点符号。"
system_role=str1+str2+str3+str4+str5+str6
```

During the keyword debugging process, gpt performed just so so, and the prompt above was the best prompt information I could find (a total of 8 versions were modified). During debugging, GPT often does not do what is required. For example, the first type of news requires the output of 8 features, but the following example only outputs 7 features:

```
'第一类新闻|||空|||空|||阿姆斯壮（西安）|||盈建科,,,东洲际|||中游,,,中游|||空',
```

Therefore, in the process of obtaining GPT answers, if the content does not meet the requirements, I use a while loop until the answer meets the standards.

```
keywords=["第一类","第二类","非产业链"]
def check(response):
    if(contains_keyword(response,keywords)==False):
```

```python
            return False
    else:
        features = response_str.split('|||')
        if "第一类" in response:
            if(len(features)!=8):
                return False
        if "第二类" in response:
            if(len(features)!=7):
                return False
    return True
def chat(content):
    response=client.chat.completions.create(
        model="gpt-35-turbo",
        messages=[
            {"role": "system", "content": system_role},
            {"role": "user", "content": content}
        ],
    )
    response = response.choices[0].message.content
    while (check(response)==False):
        response=client.chat.completions.create(
            model="gpt-35-turbo",
            messages=[
                {"role": "system", "content": system_role},
                {"role": "user", "content": content}
            ],
        )
        response = response.choices[0].message.content
    return response
```

In the final presentation, Mentor Zhang proposed that we should tell gpt where the output error is, and then let gpt answer again. However, as mentioned before, the gpt-3.5turbo interface provided by the school does not support continuous conversations. Each conversation requires opening a new chat, so it is not feasible to tell chatgpt what is wrong and make it corrected.

After obtaining the data, I processed the data. Since "|||" is used as the delimiter, the string can be directly divided into columns and saved to the dataframe. During the processing, it was found that there are still certain problems in the output, such as the company name being recorded as "midstream" or other non-company name information. This part of the processing will be described later. The obtained dataframe structure is as follows:

```
news_type    industry     product provider     buyer
provider_location    buyer_location   bill
第一类新闻    家电、家具和日用品    空    珠三角地区生产的家电、家具和日用品    空
下游   空    空
第一类新闻    风能   空    金盘科技     甘肃瓜州宝丰风能开发有限公司   中游   下游
1.5亿元。
第一类新闻    CDMO     空    普洛药业     江苏先声药业有限公司   下游   上游   空。
第一类新闻    建筑材料     熔岩管 空    北京空间机电研究所    空    中游   空
第一类新闻    口罩   空    粤万年青     空    中游   空    空
```

In order to make the industry chain analysis more visual, I used GPT to judge the industry information again. In this process, since the industry has been obtained before, it is more effective to use the GPT interface to query the industry categories. First, I tried to query the Shenwan secondary industry in the industry, using 100 news items as samples and testing, and found that more than half of the output was not classified as the Shenwan secondary industry. Then try to query Shenwan's first-level industry. About 30% of the output is not Shenwan's first-level industry. Therefore, I artificially stipulate that the major categories of industries are:

```
categories = ['Energy', 'Materials', 'Industrials', 'CD', 'CS',
'HealthCare', 'Financials', 'IT', 'Telecom', 'Utilities',
'RealEstate']
```

There are seven categories in total and judged. More than 95% of the output this time contains industry classification.

The prompt is as follows:

```
def industry_chat(content):
    response=client.chat.completions.create(
        model="gpt-35-turbo",
        messages=[
            {"role": "system", "content":"I input the detailed
industry type, please output the industry. The industry keyword
includes:Energy, Materials, Industrials,
CD,CS,HealthCare,Financials,IT,Telecom,Utilities and
RealEstate.For example, if I input '风能', you output
'Industry:Energy'.Please only output one industry keyword.Do not
output punctuation。 Please output in the format 'Industry:'"},
            {"role": "user", "content": content}
        ],
    )
    response = response.choices[0].message.content
    return response
```

However, some output does not meet the specification. Despite the request not to output punctuation, there is still a large amount of output containing ".", "。" and other punctuation marks. I took statistics on all outputs and created classification dictionaries for replacement.

```
industry_keywords=['Industry: Energy.', 'Industry: HealthCare',
'Industry: Materials', 'Industry: Industrials', 'Industry:
Energy',
                   'Industry: IT', 'Industry: IT.', 'Industry:
Materials.', 'Industry:HealthCare',
       'Industry: Information Technology (IT)', 'Industry:
HealthCare.',
       'Industry: Consumer Discretionary (CD)', 'Industry:
Industrials.',
       'Industry: IT。', 'Industry: Financials.','Industry:
Telecom.',
       'Industry:Consumer Discretionary (CD)', 'Industry:
Financials',
       'Industry: Automotive','Industry: Consumer Staples',
'Industry: Telecom',
       'Industry:HealthCare.', 'Industry: Utilities',
'Industry:IT',
       'Industry: RealEstate.', 'Industry: Consumer
Discretionary',
       'Industry: Consumer Staples.','Industry:Industrials.',
'Industry: Healthcare.','Industry:Agriculture', 'Industry:
Healthcare',
        'Industry:IT.','Industry:Materials','Industry: CD
(Consumer Discretionary)', 'Industry: RealEstate',
       'Industry: Consumer Staples (CS)', 'Industry:Industrials',
'Industry: Communication Services', 'Industry:Marketing',
        'Industry:Financials','Industry: CS (Corporate Social
Responsibility)', 'Industry:Utilities.', 'Industry: CD',
'Industry:Energy',
        'Industry: Utilities.', 'Industry: Infrastructure (CD)',
'Industry: Consulting Services.','Industry:Materials.',
        'Industry: IT (Information Technology)',
'Industry:RealEstate.', 'Industry: Industrials and HealthCare.',
       'Industry: Communication Services (CD)', 'Industry:
Utilites' ]
# 创建分类列表
categories = ['Energy', 'Materials', 'Industrials', 'CD', 'CS',
'HealthCare', 'Financials', 'IT', 'Telecom', 'Utilities',
'RealEstate']
# 创建分类字典
category_dict = {category: [] for category in categories}
```

```python
# 遍历industry_keywords列表
for keyword in industry_keywords:
    # 提取关键词
    keyword = keyword.lower()

    # 根据关键词进行分类
    if 'energy' in keyword:
        category_dict['Energy'].append(keyword)
    elif 'materials' in keyword:
        category_dict['Materials'].append(keyword)
    elif 'industrials' in keyword:
        category_dict['Industrials'].append(keyword)
    elif 'cd' in keyword or 'consumer discretionary' in keyword:
        category_dict['CD'].append(keyword)
    elif 'cs' in keyword or 'consumer staples' in keyword:
        category_dict['CS'].append(keyword)
    elif 'healthcare' in keyword:
        category_dict['HealthCare'].append(keyword)
    elif 'financials' in keyword:
        category_dict['Financials'].append(keyword)
    elif 'it' in keyword or 'information technology' in keyword:
        category_dict['IT'].append(keyword)
    elif 'telecom' in keyword:
        category_dict['Telecom'].append(keyword)
    elif 'utilities' in keyword:
        category_dict['Utilities'].append(keyword)
    elif 'realestate' in keyword:
        category_dict['RealEstate'].append(keyword)
# 输出分类结果
for category, keywords in category_dict.items():
    print(f"{category}: {keywords}")
```

Industry chain mapping and analysis

The construction of graph structure

After getting the first type of news, we can roughly build the industry chain information based on the provider and buyer of each news. However, during the research process, I discovered some problems. For example, the company name has both the full name and the abbreviation. Obviously,"小米及其代工厂","小米集团","小米手机","小米公司","小米汽车" are all affiliated with the same company, news related to them will have an impact on Xiaomi. Therefore, I need to merge company names. Here I use the fuzzywuzzy library to judge the partial similarity of some strings, and use a graph structure to store the relationship between names. The threshold is 90, which is a good standard after some test. If it is set too high, the names of the same company cannot be merged; if it is set too low, some unrelated companies but with similar characters will also be merged. The idea of constructing the graph structure is as follows:

Assume that the names of provider and buyer have been stored in the set. For the first element of the set, add it directly to the graph. For subsequent elements, check whether there is an element a in the graph that is highly similar to it. If it exists, add it. a and the directed edge of this element; if it does not exist, only the vertex is added to the graph. Finally, all strongly connected components in the graph are traversed, and the vertex name with the shortest name in the branch is used as a replacement for other elements.

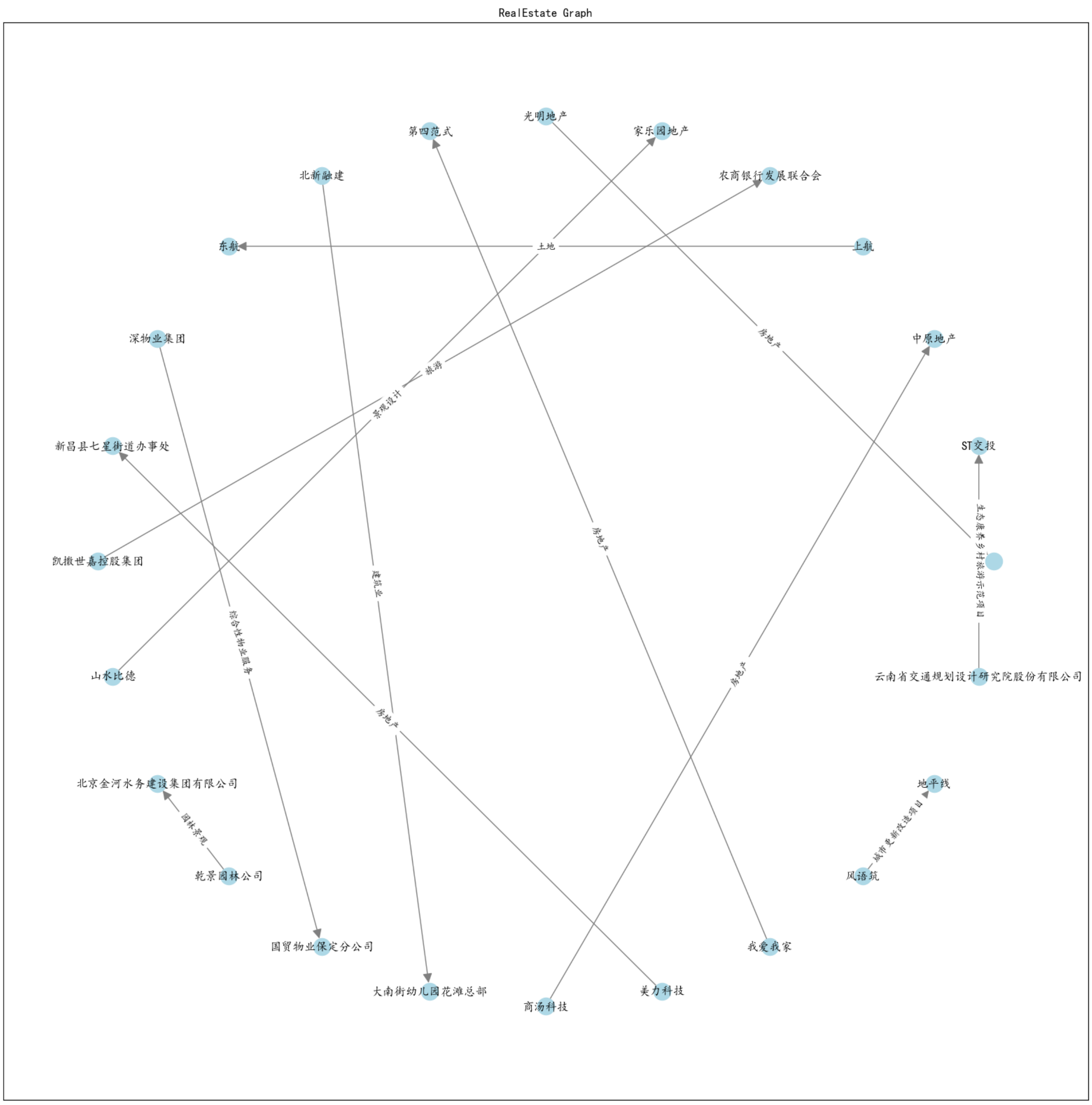
Code is show as below:

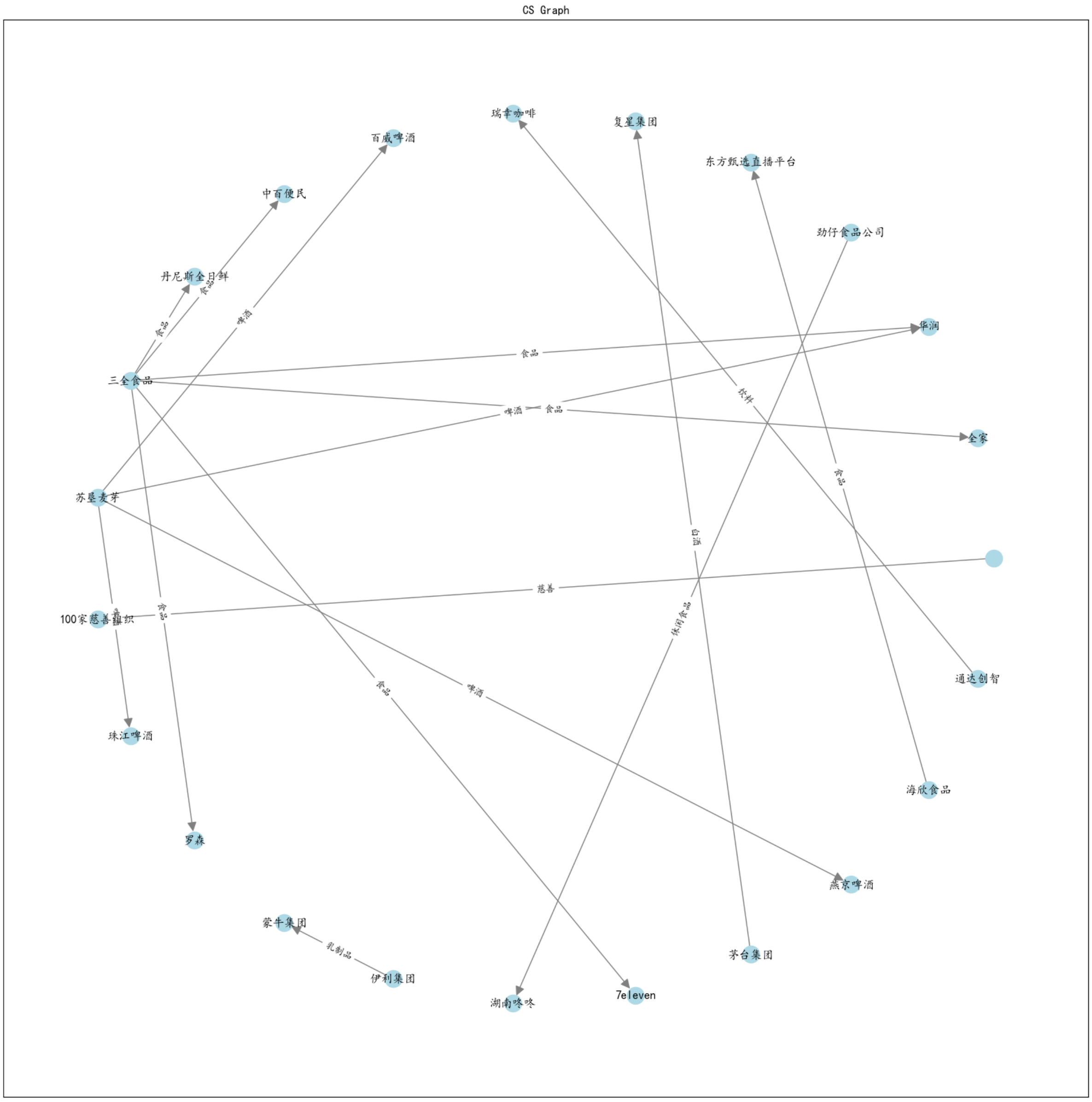
```
graph = nx.Graph()
# 创建公司列表
# 添加所有公司到图中
for company in company_list:
    nodes = graph.nodes()
    if (company not in nodes)&(',' not in company)&(': ' not in
company)&(':' not in company):
        flag=0
        # 检查是否有相似度高的元素
        node_list=[]
        for node in nodes:
            similarity = fuzz.partial_ratio(node, company)
            if similarity > 90:
                flag=1
                #print("node:",node,"company:",company)
                node_list.append(node)
        if flag==1:
            for node in node_list:
                graph.add_edge(company, node)
                graph.add_edge(node, company)
        if flag == 0:
            graph.add_node(company)

# 获取图中的强连通分量
connected_components = nx.connected_components(graph)
# 生成公司名称合并字典
merged_names = {}
for component in connected_components:
    print("connected_components group:")
    shortest_name = min(component, key=len)
```

```
    print("shortest_name in component is:",shortest_name,"|||")
    for name in component:
        print("component name",name)
        merged_names[name] = shortest_name
```

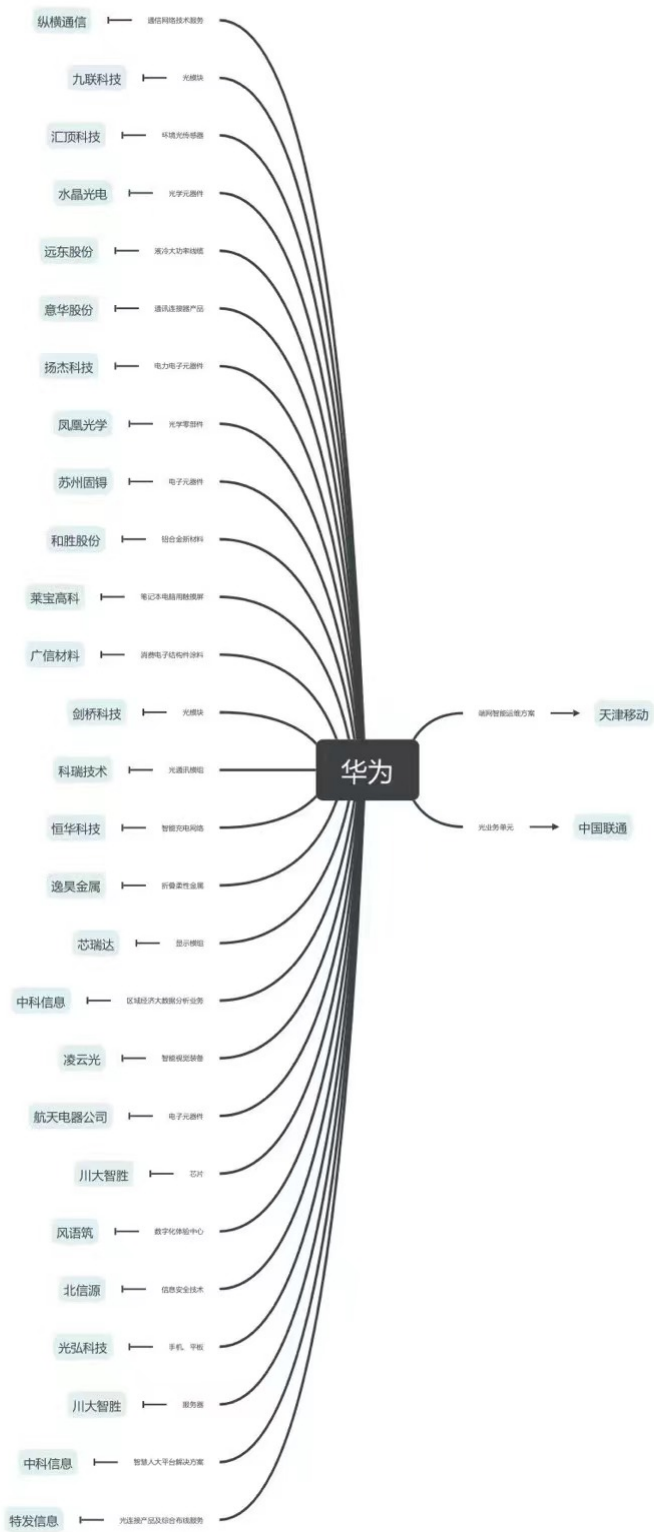
Drawing of industrial chain diagram

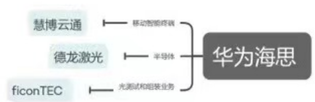
After replacing the company name, an image related to the industry chain can be drawn. Since there are many companies in some industries and cooperation is more complicated, only RealEstate and CS's industry chain diagrams are extracted for display here. Other industry chain diagrams can be obtained in the folder 'output'.





By the way, Dai also draws a supply chain of Huawei using Xmind. It is shown as below:


华为
纵横通信
通信网络技术服务
九联科技
光模块
汇顶科技
环境光传感器
水晶光电
光学元器件
远东股份
液冷大功率线缆
意华股份
通讯连接器产品
扬杰科技
电力电子元器件
凤凰光学
光学零部件
苏州固锝
电子元器件
和胜股份
铝合金新材料
莱宝高科
笔记本电脑用触摸屏
广信材料
消费电子结构件涂料
剑桥科技
光模块
科瑞技术
光通讯模组
恒华科技
智能充电网络
逸昊金属
折叠柔性金属
芯瑞达
显示模组
中科信息
区域经济大数据分析业务
凌云光
智能视觉装备
航天电器公司
电子元器件
川大智胜
芯片
风语筑
数字化体验中心
北信源
信息安全技术
光弘科技
手机、平板
川大智胜
服务器
中科信息
智慧人大平台解决方案
特发信息
光连接产品及综合布线服务
端网智能运维方案
天津移动
光业务单元
中国联通


华为海思
慧博云通
移动智能终端
德龙激光
半导体
ficonTEC
光测试和组装业务

Industry chain related analysis

After obtaining the graph structure, we can determine the nature of the graph by analyzing the connectivity of the graph. I calculated the maximum out-degree, maximum in-degree, average out-degree, average in-degree, maximum out-degree company and maximum in-degree company of the graph. The file is graph_degree_analysis_results.csv and can be found in the output folder. Out-degree and in-degree measure the frequency with which a company buys/sells products. Average out-degree and average in-degree measure the average frequency of transactions in the industry. The largest out-degree company and the largest in-degree company represent the company's position in the industry. The highest number of supply/purchase agreements were concluded.

An example is as follows:

Category	Max Out Degree	Max In Degree	Average Out Degree	Average In Degree	Max Out Degree Companies	Max In Degree Companies	
Energy	12	4	0.644699	0.644699	亚普股份	阿里, 中兴	
HealthCare	2	2	0.525773	0.525773	博济医药, 烟台东诚核医疗健康产业集团有限公司	蓝鹊生物	
Materials	13	8	0.642105	0.642105	普利特	中兴	

As for analysis, here is the example: in the Materials industry, Plit has reached the largest number of supply agreements, 13; ZTE has reached the largest number of purchase agreements, 8. This can also reflect the industry leaders and companies with high news attention. These companies tend to be relatively stable and have high investor sentiment.

Supply Chain Signal Backtesting

If the company purchases products, it means that the company has the ability to purchase products, and the company's current capital flow is good or its future income exceeds expectations; downstream customers/enterprises have relatively strong demand for products, which will bring better capital flow to the company in the future. If a company sells products, it means that the company's current revenue is high or its future revenue exceeds expectations. Therefore, both provider and buyer have better fundamental information. In addition, the exposure of the news also fully shows that the current company or industry is receiving high attention, investor sentiment is strong, and there is a high probability that the stock price will rise now/in the future.

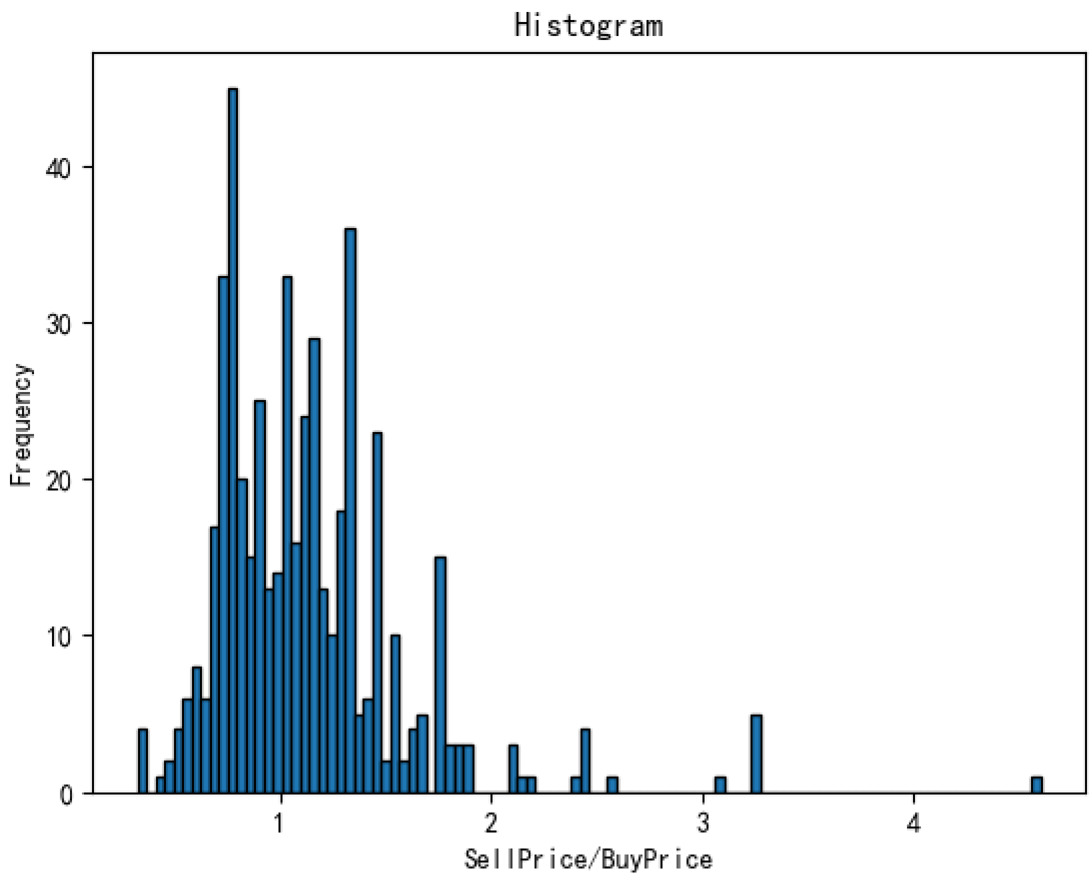
Based on the above analysis. I directly summarized the provider and buyer information, and after querying all security information in baostock, I took the intersection of the security names to obtain the stocks to invest in 2023. After processing, the number of stocks involved was 529.

The investment strategy is as follows: The trading signal is that the stock has news related to the supply chain. The investment portfolio is swapped at the end of each month. If there is a trading signal for the stock within a month, it is bought and held for one month.

To determine the profitability of each stock, calculate hit_rate as follows:

```
#hitrate的计算，即股票一个月内是否上涨
revenue_list1=[]#return
bin=len(swap_bin)
for i in range(bin-1):
    temp_frame=0
    if i==0:

 temp_frame=merged_provider_buyer[merged_provider_buyer["date"]
<=swap_bin[0]]
    else:

 temp_frame=merged_provider_buyer[(merged_provider_buyer["date"]>s
wap_bin[i-1])&(merged_provider_buyer["date"]<=swap_bin[i])]
    #get stock to invest
    stock_invested=temp_frame["code"].tolist()
    for stock in stock_invested:

 stock_next_month_data=stock_data_copy[stock_data_copy["code"]==st
ock]

 monthly_return=stock_next_month_data["close"].iloc[-1]/stock_next
_month_data["close"].iloc[0]
        revenue_list1.append(monthly_return)
count = sum(1 for num in revenue_list1 if num > 1)
hit_rate=count/len(revenue_list1)
hit_rate
```

The final result was 57%. It means that stock has a 57% probability of rising. Statistics are made on the selling stock price/buying stock price, and the histogram is as follows:



The mean is calculated as follows:

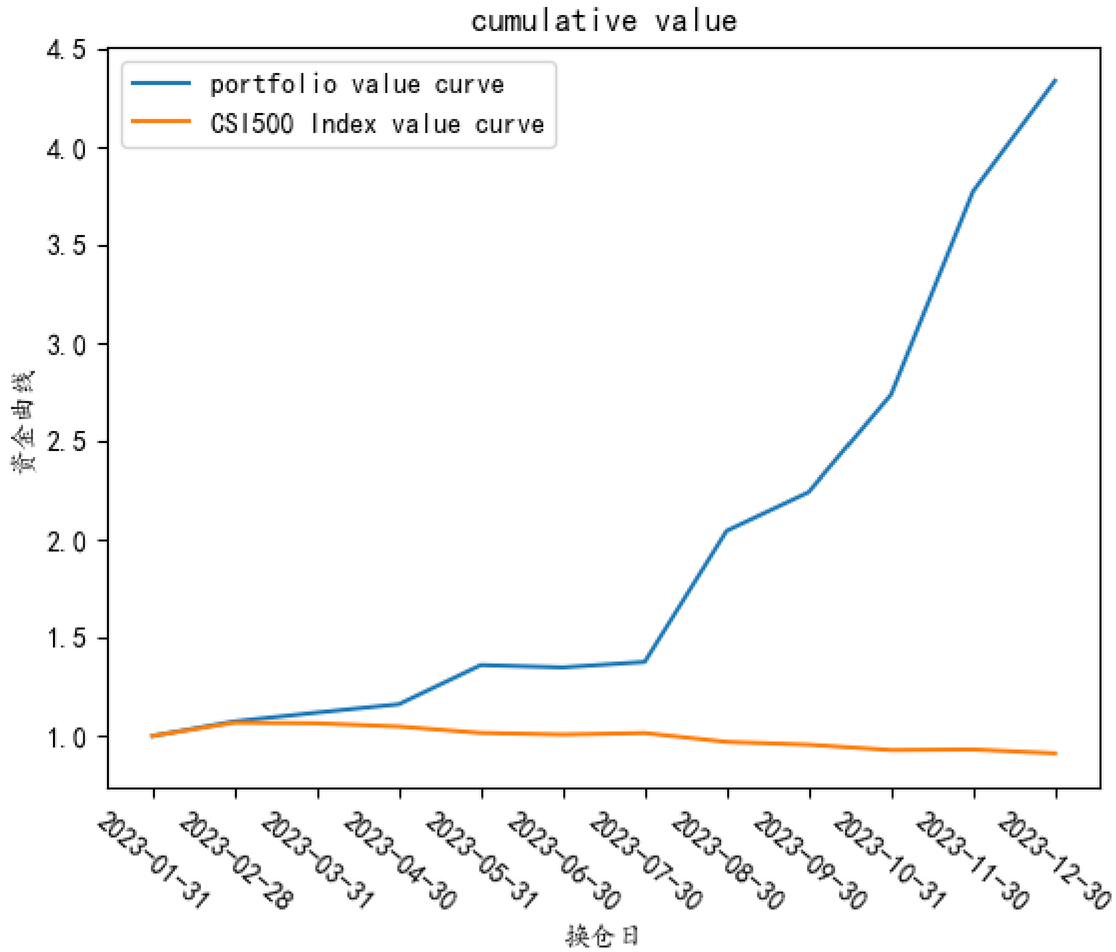
```
In  [102]: sum(revenue_list1)/len(revenue_list1)
Out[102]: 1.1290467558657842
```

Perform a t statistical test on sell_price/buyprice and the results are as follows:

```
n  [105]: from scipy import stats
          population_mean = 1   # 假设的总体均值
          t_statistic, p_value = stats.ttest_1samp(revenue_list1, population_mean)
          print("T-statistic:", t_statistic)
          print("P value:", p_value)

          T-statistic: 6.17264137931055
          P value: 1.4095814012778427e-09
```

The p value of the t statistical test is very small, indicating that the mean is significantly greater than 1. The right tail of the sequence is serious, indicating that there is a certain probability of obtaining a fairly high rate of return. To sum up, our trading signals are effective.

Construct a portfolio for backtesting. Without considering transaction fees and slippage, the resulting net value of the portfolio is as shown in the figure below:



I chose CSI 500 as the benchmark. My strategy significantly outperformed CSI 500. Therefore, the strategy is effective and the possibility of positive future returns is quite high.

Selecting industry chain news and investing in related companies has broad application prospects in quantitative investment. Due to the timeliness of news, improving investment strategies (such as buying stocks on the day the news is released or the next trading day, changing the holding time limit, etc., using information such as order amounts) is likely to yield higher returns.

Rebuttal about Final Presentaion

The GPT-turbo 3.5 provided by HKUST actually has some problems that prevent us from working further and more detailed. It is true that the prompt we design is complicated including many aspects, which GPT may not be able to generate the answer we expect. Prompting by chain is logically more reasonable and reliable, we could do this, however, due to this GPT model does not provide this function. To address this problem, we have made many attempts to make our prompts be understandable and sequential for GPT to get insight and provide us correct answer. In this section, even though is still make mistakes, it performs far more better than our original design does. Also, because of the disability of finetune of this GPT model, we cannot correct it answers by telling it gives a wrong response, as it cannot remember the previous dialogue. So the check function just make it generate answer again and again until it reaches the correct results. For the json format, it is true that it can make GPT produce more stable outputs, but due to the time limits, we don't get a detailed knowledge of it and don't use it.

Difficulties and Expectations

Difficulties:

Owing to the high cost of ChatGPT tokens, it will take a lot of time and money to run through data of all years, or screen out the industry chain news step by step from the beginning using ChatGPT. If so, prompts to be designed and the guidance to be given in the process will also be more complicated.

Before the mid-term debriefing, we tried to use multiple agents to give different prompts to classify the news into various categories, and further subdivided them into different categories from the macro-micro perspective.Our hypothesis is that different categories of news should have different impact, importance and continuity.Hence we divided them into different categories and analyzed news of each categories. Considering the different definitions and measurements of news impact, different results and exploration methods can be obtained. However, which can be seen from the description, this method needs lots of detailed analysis to carry out, which means that the amount of data can not be too small, and it takes too long to complete a reliable analysis. Due to time constraints and limited funds, we therefore decided to give up the exploration in this direction.

Expectations:

Due to time and cost constraints, we only ran news data of 2023. In fact, this framework can also be used to obtain more complete information of the supply relationship between companies and industry chain from other resourse like financial statements. At the same time, considering the the time series data of news on a larger time scale may be helpful to get information on changes in the industry chain and changes in company operation.

Secondly,the supply chain network information we get can be further analyzed by ChatGPT to forecast the current operation status of each companys according to the industry chain position and order status, or to get the overall view of each industry. Graph Neural Network or some clustering methods can also be applied to the supply chain charts to generate some new useful features.

Finally, more profitable strategy can be constructed. We swap the portfolio at the end of each month. But as the timeliness of news, buying stocks on the day the news is released or the next trading day may be much better than swapping at the end of the month. Besides, we construct equity-equally weighted portfolios, considering some factors such as order amount could also improve the portfolio performance.

All in all, we believe that this direction is of great prospects and remains much more for exploration.

Contribution:

Jia Yaoyao: All the code writing, construct keywords for supply chain, run gpt-api to extract 4 month news' features of 2023, the main analysis part of the report, and modifications to report.

Li Junyan: Construct keywords for supply chain, run gpt-api to extract 4 month news' features of 2023, the literature review and rebuttal section of the final project.

Gao Daiyutong: Construct keywords for supply chain, run gpt-api to extract 4 month news' features of 2023, the project description and difficulties and expectations section and a part of research procedure section.

References:

史峰.金融媒体新闻情绪及其对股市影响研究.对外经济贸易大学,2020.

徐翔,靳菁.网络舆情与上证指数涨跌幅的关联性分析——基于 LDA 主题模型的文本挖掘.杭州电子科技大学学报(社会科学版),2018

史永东,田渊博,马姜琼,钟俊华.多因子模型下投资者情绪对股票横截面收益的影响研究.投资研究,2015,34(05):48-65.

李奉珂.基于投资者情绪的多因子选股模型实证研究.西南财经大学,2020.

https://doi.org/10.48550/arXiv.2304.07619

https://arxiv.org/abs/2309.03079 https://doi.org/10.48550/arXiv.2401.03737

https://doi.org/10.48550/arXiv.2401.03737