**Final Submission**

“Team Money”

Wong Chee Weng

Gary Ng

\*To run the web application, it is highly recommended to follow the .README instructions within the github repository (ie we upload the .zip file of the main branch code here, but only as a backup).

**1) Class Participation (2 points)**

* We have posted an introduction to our problem on Teams
* We commented on other’s projects where we feel we can add-value.

**2) Problem Statement (5 points)**

*Problem Statement:*

* Data is messy. Quantitative researchers working with financial data require a defined framework to preprocess, organize, and analyze multiple raw datasets so that they may present research results in an accurate, informative, and interactive manner.

*Novelty of the Problem:*

* We believe the problem of data “over-supply and under-processing” is an acute and growing trend across many industries. Not only in financial data, we observe similar trends in the following examples (and many other areas):
  + Multiple Personal Communication Platforms: Instagram, Whatsapp, Telegram, Signal, Messenger, iMessage, WeChat, Discord, Teams for Personal, and more.
  + Master’s in Computing at NUS: Teams for School, Discord, Luminus, EduRec, LeetCode, Kaggle, Steven Halim’s personal webpage (IT5002), Coursemology (IT5001), Kattis (IT5002), and more.
  + New Platforms: Straits Times, Channel News Asia, Today Online, Asia One, C.N.N, B.B.C, New York Times, Google News, Yahoo News, Sky News, and more.
* In the above examples, we illustrate a similar trend of “over-supply and under-processing” in personal communication, schooling, and the current news. In all three cases, and to our dissatisfaction, there remains multiple platforms that cumulatively provide an overabundance of data be it personal messages, school files, or simply the latest news. The bottom-line is that there is time and energy loss simply trying to aggregate data and information for simple daily tasks. This trend is similarly exhibited in the supply of financial data where there is an overabundance of data providers, but few people ready to provide useful and actionable data interpretation and analysis services.

*Challenges Involved in Solving the Problem:*

* For Gary, his challenge was to be able to contribute to the software development portion of this project as he is new to any programming language that is not Python. He will be able to contribute on a leadership front driving the broad ideas for this project, reporting, and in application of his domain knowledge in finance (any financial jargon or data that needs to be interpreted). In retrospect, the challenge for this project was to come up with the initial framework (github repository / code branch) to help Chee Weng visualize what the final desired product would be like. Though the code is not used in the final project, it was good practice for Gary to feel the time pressure and deliver a working web application with fully working components.
* For Chee Weng, his challenge was to understand the problem well as this is not his professional domain and to anticipate which portions of the project take priority over others (what to do next). His add-value was in all aspects of the technical development of this project, taking the base web application Gary drafted early on in this project lifecycle, and finally creating the github repository’s main code branch which serves as our actual code submission for this project.
* A final small challenge was that our group required a third-party graphing API and spent some time (about a week) trying out different graphing libraries, which is important to note here.
* From a collaboration perspective, the challenge for this project will be to communicate well and early with each other such that both parties have sufficient time and confidence to jointly deliver our intended solution. In retrospect, we were a good fit as the final product was complete ahead of schedule without any major project-breaking hiccups.

*Is the Problem Relevant in 2/5/10 Years?:*

* Yes, in fact we believe that this problem will grow worse if left unsolved in the coming years. Repeatedly approaching data research projects without a structured framework, data ingestion process, and processing pipeline will lead to (1) a slower overall development process (2) sub-optimal application of one’s company personnel (talent), and finally (3) the possibility of erroneous research findings and results.

*Complexity of the Solution:*

* The solution to our problem is not at all complex. On the contrary, our goal is to follow the “simple is best” approach such that an entry-level data analyst may understand the basic workflow and framework easily. Rather than complexity, we aim for a simple, scalable, and broad enough solution to not only solve this problem but provide a framework for all data projects in the future. Ideally, our solution would be to design a software architecture that provides a skeleton for a new quantitative researcher who has just joined our hypothetical company to quickly get up and running and develop in.
* It is my (Gary) experience over the past 8 years in the industry that there are about 2-3 major research projects that pop-up over the course of the calendar year. These projects either live or die based on how fast the initial results can be obtained. Sometimes, although unfortunate, there is just a loss of interest due to too much time and effort passing, and research projects get abandoned when there is still some valuable information to be discovered remaining.

*Features:*

The Data Table

* This is the core of the web application that presents all the details from 3 major data inputs (Brian Freitas, Intropic, and Mizuho Bank). The columns are not too important, however the jist is that each line is a financial security that is either being added or deleted from a major index in the coming future.

Graphical user interface, application

Description automatically generated

The Charts

* Clicking on a financial security (Ticker Column), the charting area will populate with three charts.
* Chart 1: A basic price (line) and volume (bar) chart history of recent prices.
  + Intuition: “How has the stock been performing?”
* Chart 2: A basic price % change (blue) and % change versus a pre-defined benchmark index (red)
  + Intuition: “The stock did good/bad, but how good/bad?”
* Chart 3: This is an example of the added-value analysis conducted by the quantitative researcher, in this chart we demonstrate what is known as “excess volume,” which is the volume traded above or below a pre-defined lookback period average volume. For example, how did today’s volume trade relative to the average of 3-months, 6 months ago?

Chart, histogram

Description automatically generated

The Filters

* We may filter all three input sources (the list can be quite long as you can see) for specific events of interest, either by event name, ticker name, or the analyst who contributed the data.

Graphical user interface, application

Description automatically generated

Export to CSV

* Finally, based on the three charts, personal judgement, and discussion with team members, we are able to shortlist and export the tickers that we find interesting. The CSV format is an example, but can be customized to include number of shares and other useful metrics that we can upload directly to a trading software to execute in financial markets.

Graphical user interface, text, application

Description automatically generated

**3) Solution Architecture (5 points)**

* Back-End Solution: Dataset Ingestion
  + Data Source 1: Market Data (From Bloomberg)
  + Data Source 2: Research Data from Analyst #1 (Intropic.io Data Provider)
  + Data Source 3: Research Data from Analyst #2 (SmartKarma Data Provider)
  + Data Source 4: Research Data from Analyst #2 (Mizuho Bank Data Provider)
* Middle: Quantitative Researcher’s Workspace
  + We would like a part of the pipeline to be an environment where the research may query data, apply basic data analysis techniques, and output his research results
* Front-End: Informative & Interactive
  + The final product will be a dashboard that draws upon the raw data and research results to display the relevant data (both useful raw and processed data) in an insightful and interactive manner with the ability to “export to csv” a shortlist of financial securities.
* Note: The above data sources and quantitative researcher add-value analyses are just for illustration, i.e. not supposed to be novel in anyway. We are more interested in the data “pipeline,” i.e. to demonstrate how four data sources can be used together to present coherent and useful information.

**4) Legal/Other Aspects (1 point)**

The processed data, web application, and it’s back-end components may be shared with our classmates and others in private settings but are not meant for large scale distribution to the public. We should refrain from distributing the raw data (MongoDB contents) in its raw form, but feel free to display the web application for others to interact with and ask questions.

* For example, Chee Weng may share this web application with his potential new employers during his upcoming internship with a financial services provider specializing in Crypto. Gary has already shared the web application with his teammates and will follow-up should this be a tool they wish to add to their daily “watchlist.”

**5) Competition Analysis (2 points)**

* There is a list of about 100 financial data providers, yet what we are proposing is to develop this framework as skillset for our personal toolkit. There are a few financial data aggregation and interpretation services out there, but again, if one were to sign up for even 10 of these providers, there would be an overwhelming amount of data. For these service providers, there is also the legal issue of (re)-distributing their data, which again obfuscates the underlying raw data (often companies can only show you the result and not share the raw data with you). Hence this project is important for us to take ownership of the entire front-to-end research process and improve our hire-ability in the job market.

**6) Git Repository (0 points)**

[https://github.com/cheewengg/it5007-project/](https://github.com/cheewengg/it5007-project/tree/report)

* Main branch is for our official web application and final submission
* Code branch is for Gary’s initial web application beta idea and visualization
* Report branch is for any general documentation such as this document