Universitatea
Transilvania
din Brașov
FACULTATEA DE MATEMATICĂ
ȘI INFORMATICĂ

Master Degree

LIT

Lean Into Trading - Trading Platform

Author: Rîtan Mihai-Lucian
Coordinator: Lect. Dr. Băicoianu Alexandra

Brașov, România
2021-2022

Contents

Glossary

Commodities in the economic sense, refer to goods whose value is determined by the market. Commodities include agricultural items like grain, rice, and coffee, mining products like coal or iron ore, precious metals like silver or gold, and energy products like oil and gas. 12

Cryptocurrency is a virtual currency. It is technically impossible to counterfeit a cryptocurrency since it is protected by cryptography. However, as we have seen in reality, if you control half of the network or the coin in question is a fork of another cryptocurrency that has harmful code or vulnerabilities, it is no longer as safe. 12

Fiat currencies is a form of currency that is not backed by any commodity object, such as silver, oil or gold, and is often proclaimed legal tender by the government. The USA began using fiat currencies in 1971, and there are now no nations that do not utilize fiat currencies. [4, The Fiat Standard p. 1] . 12

Fintech is a colloquial term that combines the terms 'financial' and 'technology. It refers to the use of technology to the delivery of financial services and assets to customers.' [2, Fintech Law and Regulation p. 1]. 12

Framework is a collection of two or more libraries, complete with boilerplate code, settings, and all the necessary tinkering. [4, Neos T., Flask Web Development p. 5] . 14, 15, 17–19

Micro-framework is a term that refers to a framework that has a reduced number of libraries. [4, Neos T., Flask Web Development p. 5] . 20

Monolith application is an application in which the user interface and all of the program's functionality are contained in a single entity . 24

Share is a portion of a company, and the total number of shares represents the whole firm. Any number of shares may be issued by a firm, and ownership is determined by the percentage of shares held. The number of shares issued by a firm is not set and may fluctuate over time, however dilution of shares has no effect on the percentage of ownership. . 4

Stock , represent the proportion of a company owned by a person, institution, or other entity types. . 2, 26, 30, 31, 37, 49

Ticker , simply put, a ticker is a bundle of data that displays the stock movement of a given company identified by a symbol. Typically, this sign is an abbreviation for the company's name.

MSFT	13K @	130.13	▲	3.02
↑	↑	↑	↑	↑
TICKER SYMBOL	SHARES TRADED	PRICE TRADED	CHANGE DIRECTION	CHANGE AMOUNT

. 31

Acronyms

A Agilent Technologies. 31

AAPL Apple. 52

ABCB Ameris Bancorp. 52

AI Artificial Intelligence. 15, 21, 24–26

AOP Aspect-Oriented Programming. 19

APAM Artisan Partners Asset Management. 52

API Application Programming Interface. 15, 20, 24, 25, 29, 30, 32, 57

ARIMA Autoregressive integrated moving average. 37

ASB Associated Banc-Corp. 52

ATO.PA Atos Paris. 31

AWS Amazon Web Services. 61, 62

B Billion. 47–51

BERT Bidirectional Encoder Representations from Transformers. 2, 23, 34, 35

BSE SENSEX Bombay Stock Exchange Sensitive. 42, 48–50

CEO Chief Executive Officer. 31

CFD Contract for Difference. 12

CFO Chief Financial Officer. 31

CNN Convolutional Neural Network. 54

CNY Chinese Yuan Renminbi. 51

CORP Corporation. 48

CSI Chinese Shangai and Shenzen Index. 50

CSS Cascading Style Sheets. 17, 19

CSV Comma Separated Value. 30, 32, 33, 60

DB Database. 2, 21, 22, 25

EMA Exponential Moving Average. 44, 45

FTSE Financial Times Stock Exchange group. 42, 47, 48

GAN Generative Adversarial Networks. 2, 10, 53, 54

GBX Great Britain pence. 48

GOOG Alphabet Inc Class C. 52

GOOGL Alphabet Inc Class A. 52

GPT Generative Pre-training Transformer. 23

GRU Gated Recurrent Unit. 2, 37, 39, 41, 54–56

GUI Graphic User Interface. 23

HKD Hong Kong Dollar. 50

HSI Hang Seng Index. 42, 50

HTML HyperText Markup Language. 19, 31

IDE Integrated Development Environment. 23

INC Incorporated. 47, 49–51

INR Indian rupee. 49

IPO Initial Price Offering. 31

JDBC Java Database Connectivity. 19

JPY Japanese yen. 48

JS Javascript. 15, 20

JSON JavaScript Object Notation. 21, 32

LIT Lean Into Trading. 1, 10, 12, 13, 15, 17, 19–22, 24–26, 28, 30, 31, 34, 44, 45, 47, 50–55, 58–62

LSTM Long short-term memory. 2, 37, 39, 41, 55

LTD Limited Liability Company. 48

M Million. 49

MA Moving Average. 43

MACD Moving Average Convergence Divergence. 10, 45, 46

MAIN Main Street Capital Corporation. 10, 42, 43, 52, 53, 58

MSFT Microsoft. 52

MVC Model-View-Controller. 19

NASDAQ National Association of Securities Dealers Automated Quotations. 42

NIKKEI Nihon Keizai Shimbun. 42, 48, 50

NLP Natural Language Processing. 2, 22, 23, 33, 34

NN Neural Network. 39

NYSE New York Stock Exchange. 42

ORM Object–Relational Mapping. 19

OS Operating System. 20, 23

PaaS Platform as a Service. 23

PG Postgres. 25, 26, 57, 59, 60

PLC Private Limited Company. 48

PNG Portable Network Graphics. 58

RDBMS Relational Database Management System. 21, 26

RDS Relational Database Service. 62

RELU rectified linear activation unit. 54

REST Representational State Transfer. 57

RMSE Root Mean Square Error. 56

RNN Recurrent Neural Network. 2, 10, 38, 39, 41

S&P Standard and Poor's. 42, 47, 49–51

Sass Syntactically Awesome Style Sheets. 17

SD Standard Deviation. 44

SE Societas Europeae. 31

SIE.DE Siemens AG Deutschland. 32, 56

SMA Simple Moving Average. 44

SQL Structured Query Language. 16, 21, 22

SSE Shanghai Stock Exchange. 42, 50, 51

SUV Sports Utility Vehicle. 37

SVG Scalable Vector Graphics. 58

T trillion. 47–52

TS Type Script. 14, 15

URL Uniform Resource Locator. 31, 32

US Uniated States. 30, 43, 47, 49

USA United States of America. 3

USD United States Dollar. 42, 47, 49

VIX Volatility Index. 42, 51

WSGI Web Server Gateway Interface. 20

YML Yaml Ain't Markup Language. 62

ZX China Zenix Auto International. 31

List of Figures

Chapter 1

Introduction

The introduction serves the purpose of leading the reader into the degree's theme.

LIT, an acronym for Lean Into Trading, is a management and research platform that resembles a social networking-style platform. Apart from the fundamental component of programming, we will delve into the realm of finance in this and subsequent chapters.

1.1 Motivation

The fintech business has seen a surge in popularity across all demographics over the previous two to three years, owing to the economic uncertainties caused by the covid epidemic and recession. In these uncertain times, an increasing number of consumers seek a more secure and diversified method of storing or investing money, rather than physically storing fiat cash or depositing it in a bank.

Despite the abundance of applications that enable you to buy and sell specific assets such as stocks, CFDs, fiat currencies, commodities, and more recently, cryptocurrency, only a hanful allow you to conduct research in this area, manage your portfolio, and communicate with other investors via their apps.

1.2 LIT's position in industry

From this niche, I identified four areas of interest: social, research, management, and transaction. In this domain, LIT is positioned between social, research, and management. Other contenders include traditional social media platforms such as Facebook and Reddit, fintech applications such as Etoro, Trading212, and Revolut, and management and research platforms such as Simply Wall St and Seeking Alpha. When it comes to researching, managing, and socializing, this is where our application shines. The figure 1.1 illustrates very clearly all of the above.

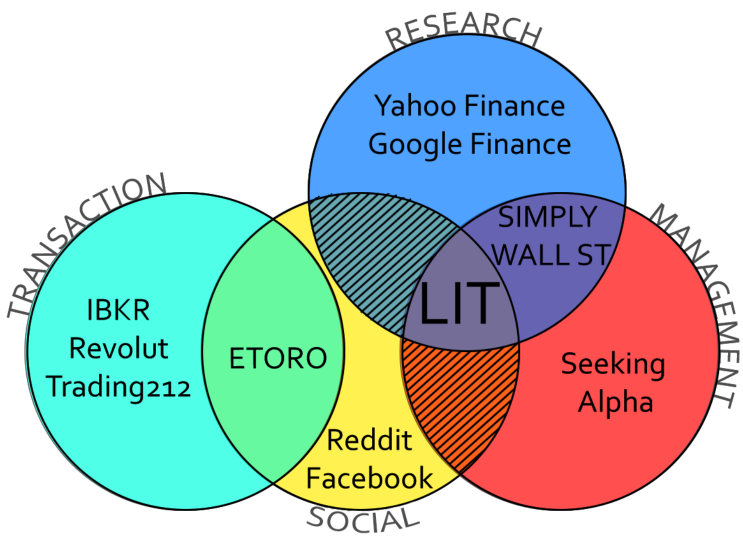


Figure 1.1: Postion of LIT in Industry

Chapter 2

Technologies

Due to the fact that we will be discussing a wide range of topics in this chapter, it has been divided into six sections: programming languages, frontend, backend, machine learning, databases and DevOps. We will explore the technologies utilized in this project in this section.

2.1 Programming Languages

2.1.1 Java

Java has several parallels with other high-level programming languages such as C++, Python, or C#, such as strong abstraction, loops, and objects. However, unlike the others, Java may operate on any system thanks to its own virtual machine. The Java programming language has resulted in a slew of sophisticated frameworks for online project development, one of which is Spring Boot, which we shall cover in further depth in subsection 2.4.1.

In summary, java is used in this project since it is a highly flexible programming language that serves as the foundation for many complex frameworks such as Spring Boot.

2.1.2 TypeScript

TypeScript augments JavaScript with new syntax to enable closer interaction with your editor. Correct problems as soon as possible in your editor. TS code is transformed to JavaScript, which will be executed in any environment that supports JS, including a browser, Node.js, or Deno, as well as in your apps, because TypeScript is a superset of JavaScript [2.1], any JavaScript programs that are already in existence are also valid TypeScript programs. TypeScript comprehends JavaScript and makes advantage of type inference to provide you with excellent tools without the need for extra code.

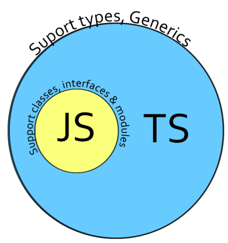


Figure 2.1: TypeScript superset of JavaScript

The blank javascript approach is preferred by many fronted developers, but when working with both the frontend and the backend, it is frequently more practical to know what kind of data we are working with. LIT use React as its froneted, and bewteen JS and TS uses TypeScript as the programming language.

2.1.3 Python

As previously discous in Java subsection 2.1.1 Python is a high-level language along with C++ and C#. Its design philosophy places a strong emphasis on code readability, which is achieved by the usage of substantial indentation. Its language features and object-oriented approach let programmers write concise, logical code for small and big projects alike.

Additionally, when it comes to Machine Learning, Python is the language of choice since it is the building block for some of the most powerful frameworks in the field, one of which is Tenserflow, which we will discuss in greater detail in subsection 2.6.1.

Python is the AI's backbone of Lean Into Trading. The AI backend of the Sentiment Analysis Service and the Flask API wrapper that provides access to it are both written in Python. The same is true for the Prediction Service.

2.1.4 SQL

SQL is a computer language that is commonly used in relational database management systems or data stream management systems.

SQL has been a continually popular option for database users throughout the years, owing to its simplicity of use and the efficacy with which it searches, manipulates, aggregates, and performs a wide variety of other tasks to transform huge quantities of structured data into useable information.

As a result, it has been included into a variety of commercial database solutions, including MySQL, Oracle, Sybase, SQL Server, and Postgres.

2.1.5 PgSQL

Due to the fact that PgSQL is a fully featured programming language, it provides significantly more procedural control than Structured Query Language, including the ability to perform loops as well as other control structures. It was designed to enable Postgre users to execute more complicated operations and calculations than they could with SQL while still being simple and easy to use.

2.1.6 NoSQL

NoSQL and SQL-flavored languages are both types of computer languages that are often used in relational database management systems or data stream management systems. The primary distinction is that SQL is a structured language, while NoSQL is not.

In summary, SQL is advantageous for data that is constant and explicit in terms of data type, while NoSQL is advantageous for unpredictable data, such as that found in environments that operate with a variety of documents or messages. They are also somewhat different in terms of syntax, the Non-SQL resemble more to a procedural language in paradigm. Both are advanced languages, making them simple to use and learn.

2.2 Frontend

This section will describe the frontend layer, which is the visual component, the layer that the user sees and interacts with. The frontend consists of components (pages, buttons, charts), their styling (css, sass), and the functionality of the components (navigation, dynamic events).

2.2.1 React

React is a highly popular frontend web framework that runs on javascript but can also be used with typescript. As seen in the graph below, fig. 2.2, this framework is quite popular among front-end developers. In fact, according to website statisa, one of the largest data analytics platforms, React was the first most popular web framework in 2021 [2.2].

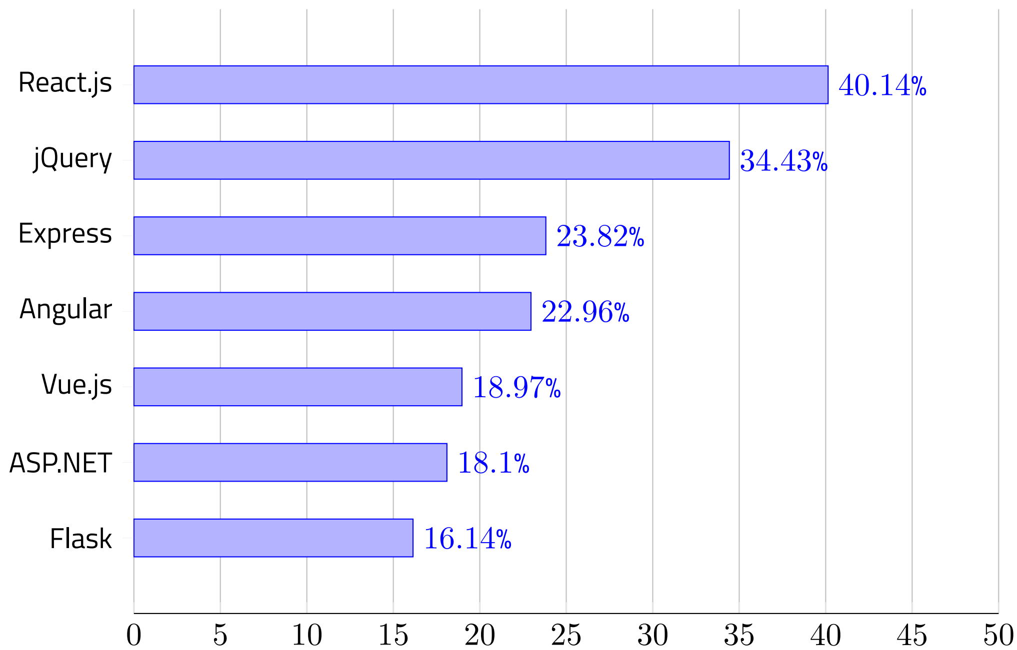


Figure 2.2: Most used web frameworks among developers worldwide, as of 2021 [7, Statista online survey on 67000+ people]

2.2.2 Sass

Sass is a stylesheet-based programming language used to style the frontend. Finally, this is translated into CSS. What Sass adds to Cascading Style Sheets is a more advanced syntax that is reminiscent of object-oriented languages; it supports inheritance, variables, and nesting, among other features.

In fig. 2.4 and fig. 2.3 are simplified example of a Sass script and CSS script, from LIT project, that shows how nesting works.

```css
.app-container {
  display: flex;
  width: 100%;
  .view-container {
    flex: 19;
    padding: 1.5rem;
  }
}
```

Figure 2.3: CSS code example

```scss
.app-container {
  display: flex;
  width: 100%;
}

.app-container .view-container {
    flex: 19;
    padding: 1.5rem;
}
```

Figure 2.4: Sass code example

2.2.3 Redux

Redux is a framework that manages the application's state and is used to govern the application's flow. While some developers find redux overwhelming owing to its complexity, it has shown to be a highly reliable alternative for big projects. Additionally, Redux is a highly popular choice among React and Angular developers. Redux helps Lean Into Tradingin managing the intricate page and component structure.

2.3 Font Awesome

Font Awesome is a widely used icon, font library, and toolkit, with millions of developers, designers, and content producers. Furthermore, Font Awesome was registered as a font script on over 6,500,000 websites in 2022, making it the second most popular choice, behind only Google. [10]

With the 6.1.1 version of Font Awesome, the following features are included[11]:

- 16,083 Pro Icons
- 5 Icon Styles
- 2,009 Free Icons
- 68 Categories

2.3.1 Reactstrap

Reactstrap is a React package that encapsulates the Bootstrap library. Reactstrap eliminates the need for developers to create bespoke code on top of Bootstrap or, even worse, to write raw HTML components with bootstrap CSS. In React applications, it is best practice to develop react components as opposed to plain HTML components since react components are more flexible, contain hierarchical logic, allow parameter migration, and so on.

In this project, the most common components, such as Buttons, Text Fields, Inputs Fields, and Lists, are reactstrap components, whereas more complex components, like as Graphs, are provided by other packages or are custom-built to satisfy the expectation.

2.4 Backend

In this section we will speak about the backend component, here is where the business logic resides, without this esential layer the connecion with our databases will not be possibile. The backend additionally ensures the security of user data by encrypting the database and using Java Spring Boot Security to also guarantee a safe connection. Each microservice has its own backend, allowing the project to use many programming languages depending on its requirements. Chapter 3 provides further information on these services and how they operate independently and communicate with one another.

2.4.1 Spring Boot

Spring is a framework that enables us to construct online applications more easily owing to the many useful modules it includes. Spring has modules ORM, MVC, AOP, Security (which is a required and implemented module for LIT), Testing, and JDBC. Spring Boot is a Spring extension that enables us to write less code; standard Spring requires much more setup and requires us to manually set up the application.

2.4.2 Tomcat

Tomcat is a robust, secure, and well-managed open source web server environment. In fact, as seen on their website home page, they update frequently and consistently. [9]

Due to the fact that LIT is a web-based application, it needs a web server. Its backend is

Spring, and Tomcat is one of the web-based environment options for Spring. Additionally, Spring Boot automatically configures Tomcat.

2.4.3 Node

Node is used to execute JS outside of the browser; we need Node to run React; otherwise, we would not be able to display any visual elements. Node.js's ability to operate on a wide variety of platforms, including Windows, MacOS, OS X, Linux, and Unix-based systems, is a significant plus that simplifies the software development process and expands the range of deployment choices available.

Node.OS allows for the request of other services to be made via API calls in a highly effective and non-blocking way, which is essential for the project that is being presented.

Node.OS provides server capabilities for the frontend portion, and it this project serves as the server component of React. Tomcat, which has already been stated, is a server component as well, but it serves the java backend portion of the project.

2.4.4 Elasticsearch

Elastic Search is based on Apache Lucene [22, Elasticsearch B.V software presentation] and is quite comparable to a typical search engine in terms of functionality. Elastic Search is a one-of-a-kind search engine that indexes and searches all forms of data, including numerical, textual, geographic, unstructured, and structured data.

2.4.5 Flask

Flask is a Python merge of two libraries that is used to create backends for websites. Because it is a fusion of two libraries, one of which is a WSGI called Werkzeug and the other is Jinja a Python template engine, it is often referred to as a micro-framework ([3], p. 5).

Flask is used in LIT to provide an API for the Machine Learning code. In the case of current project, this necessitates the development and training of a Generative Adversarial Network, which will be addressed in further detail in the Implementation Details Chapter at section Stock Forecasting.

On brief with Flask, the user is able to call independently, only this section of the project, to get feature stock forecast or sentiment analysis of a news healine, tweet, comment, or any short text pertaining to financial matters.

2.5 Databases

2.5.1 MongoDB

Mongo is a NoSQL relational database management system, more on NoSQL in subsection 2.1.6. MongoDB, like PostgreSQL, is an open-source and free database management system. As a non-SQL database, it is readily scalable and adaptable to inconsistent data types such as documents. Additionally, MongoDB is an excellent choice when dealing with a large number of JSON files.

As a result of MongoDB's radical paradigm shift, developers no longer deal with tables, rows, and columns, but rather with collections, documents, and fields. JSON files may be stored directly with relative ease, despite the fact that this may seem perplexing at first. Nonetheless, some comparisons are possible, as shown in the figure 2.5.

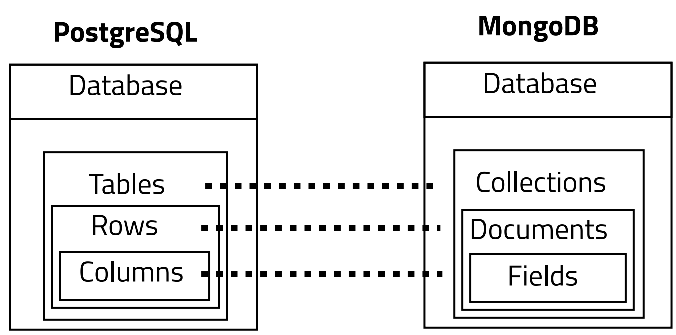


Figure 2.5: Relational vs Non-Relational

As utilised by LIT, MongoDB is used to store the data created by the AI services as illustrated in fig. 3.1. This service may produce various JSONs based on input or accessible data.

2.5.2 PostgreSQL

Postgres is a massively popular completely free open-source RDBMS. According to a recent poll conducted by stackoverflow on a sample of 53,312 users, PostgreSQL is the second most popular database, narrowly beaten by MySQL [2.6].

When it comes to brute force, PostgreSQL is unquestionably the winner, outperforming MySQL in many circumstances. MySQL is still somewhat more popular, but if MySQL does not catch up quickly, it will be dethroned. [12]

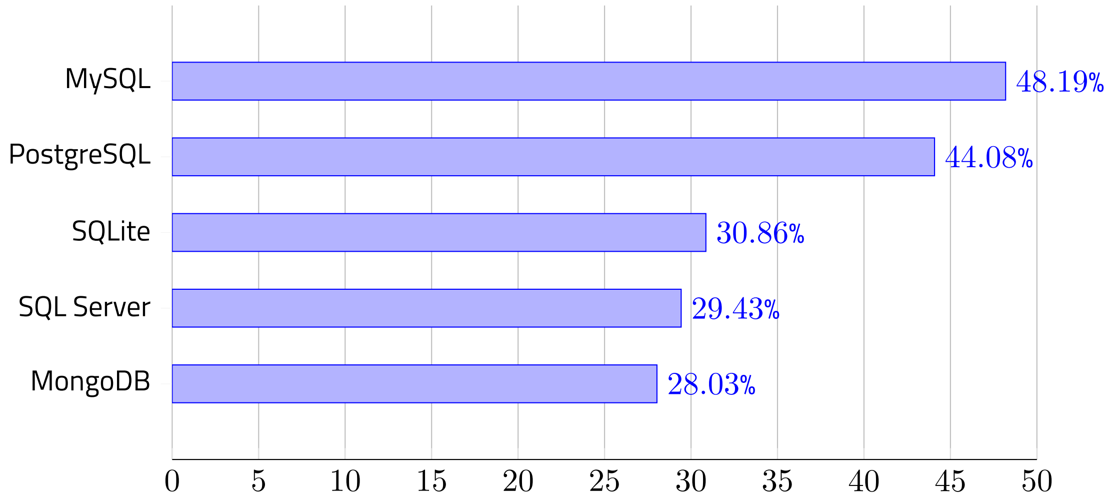


Figure 2.6: Database popularity survey
[8, Stack overflow online survey]

2.6 Machine learning & NLP

During this chapter, we will discuss the Machine Learning and Natural Language Processing aspects of the project and their implications. In order to assist the user in making better investment decisions, one of the project's services is devoted to predicting the future performance of certain enterprises. This service is powered by a Generative adversarial network, while another service focuses on natural language interpretation. On a scale from -1 to 1, an NLP service may determine whether a financial statement is negative or positive. This AI solution saves the user both time and the requirement for technical analysis, which would otherwise be necessary for a deeper understanding of the market.

2.6.1 Tensorflow

Tensorflow Google-backed Tensorflow is a massive artificial intelligence and machine learning library. It supports a variety of programming languages, including Java, C++, Javascript, and Python. LIT makes use of Python, which is the most common choice for Tensorflow.

2.6.2 Pandas

Pandas is a Python library that offers quick, versatile, and expressive data structures that make it simple and natural to deal with ”relational” or ”labeled” data. It is intended to serve as the foundational high-level building block for doing pragmatic, real-world data analysis in Python. Additionally, it aims to be the most powerful and versatile open source data analysis and manipulation tool accessible in any language.[13, Pandas: powerful Python data analysis toolkit]

2.6.3 BERT

BERT is a language representation model. In a nutshell, BERT is used to forecast future words in a phrase or to complete a sentence’s missing words. [1]

LIT uses this Google-provided NLP approach since it is open-source and free, a more powerful alternative would be the GPT3, but this transformer is only accessible through a specific API, which you must request access to.

2.6.4 Numpy

Enchants the mathematical capabilities of python, one notable benefit is that it provides support for basic matrix and multidimensional matrices.

2.7 DevOps

2.7.1 Docker

Docker is a series of PaaS products that encapsulates a software product in a virtualized environment, allowing you to execute a local program anywhere, such as on other personal computers or servers, unrestricted by the operating system or software dependencies that the application may need.

Docker supports Windows, macOS, and Linux, so a software product can be developed on a complex OS such as Windows, which has multiple IDEs and GUI tools for development such as Intellij IDEA, Github Desktop, and Visual Studio Code, and then deployed on a Linux server that does not require any of the development software. In this way, a product may be constructed quickly on a powerful, sophisticated system, but it can be easily launched on a machine that prioritizes resource efficiency.

Chapter 3

Solution Architecture

This chapter will describe the project's architecture. The primary application is a microservices-based application that uses two additional parallel services. Microservices are developed using Java Spring Boot, whereas parallel services are developed with the Python package Flask. Each microservice and service possesses its own database and communicates through API calls. The overall architecture is seen in fig. 3.1.

litis based on services and microservices, a fairly recent architecture. The conventional solution is a monolith application. LIT is a complex project with numerous components doing a variety of tasks, and the ideal approach is to keep these components segregated and under tight control. Each component must be scalable, simple to use, simple to alter, and even removable without causing faults in the project. All of the aforementioned are obstacles in monolithic applications. The components are tightly coupled and difficult to remove, the resources are allotted to the whole program; in other words, a component must share its resources equally with the others, which might create bottlenecks.

The communication between all of the services and microservices that are placed on different ports and that may scale to numerous additional ports is one of the service oriented architecture's disadvantages. Section 3.2 addresses this issue.

Conceptually, the project may very well be divided into two parts: the Java Web Application and the AI application. Additional information is provided in the following sections.

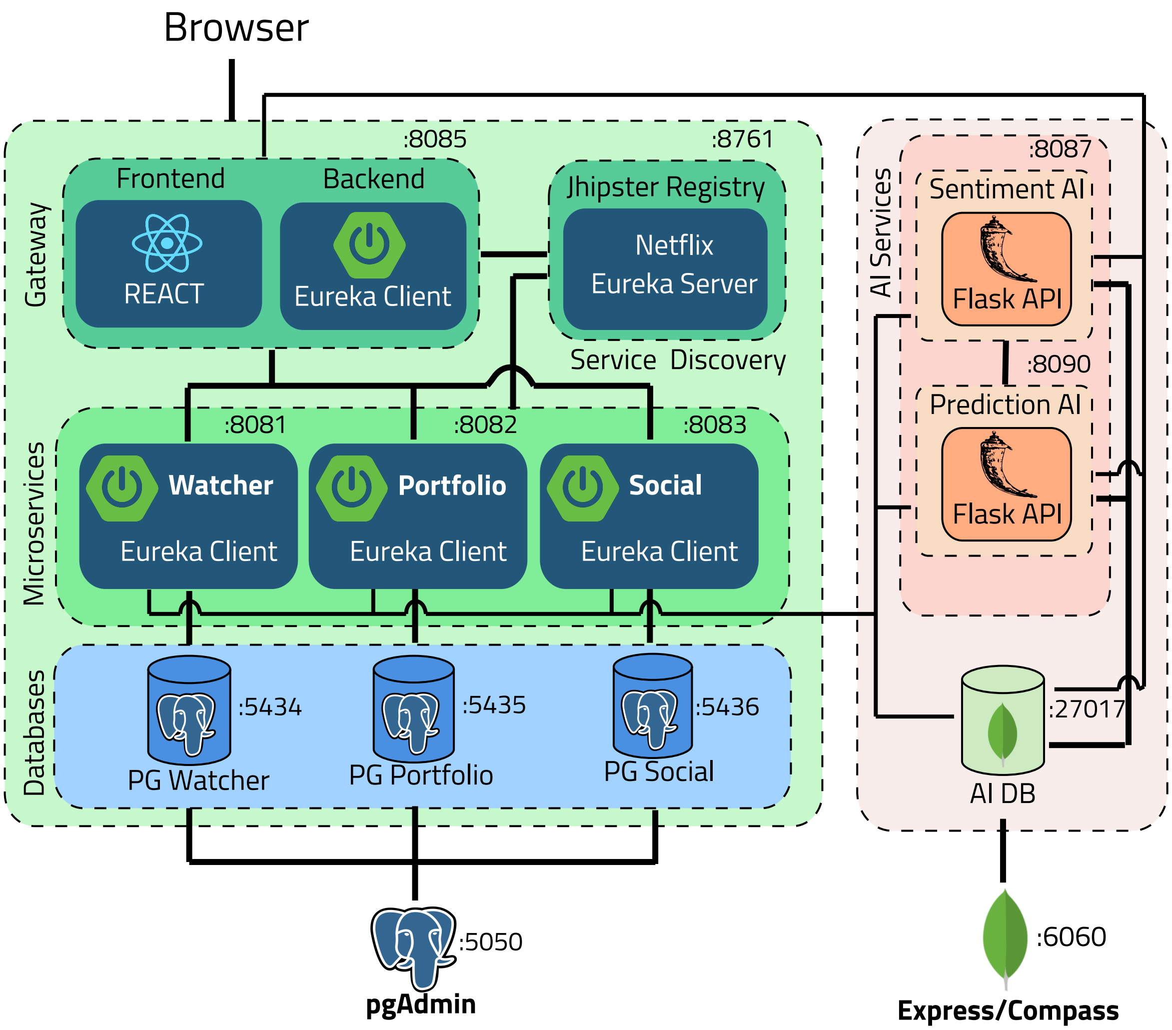


Figure 3.1: LIT architecture

3.1 Flask Services

These services are devoted to the AI portion of the project. As their names imply, one can provide a sentiment analysis of a given text, while the other may train and return a range of predicted prices for a certain period. Each service and database execute on their own ports 3.1.

port	8087	8090	27017	6060
name	Sentiment AI	Prediction AI	AI DB	Express
	Services		**Database**	

Table 3.1: Port allocation for AI application

3.2 Java Spring Boot Microservices

The main point of access is the gateway, which is a react application with a eureka client backend.

The gateway enables the user to interact with the microservices functionality and AI services. The user never interacts directly with the services or microservices, all requests go via the gateway.

This kind of communication may get problematic due to the fact that these microservices are often referred to by their address, for example: localhost::8080/stock. When a microservice is relocated to a different port or when the same microservice is hosted on numerous ports, the implementation of communication becomes quite complex. One approach is to employ another service that solves this communication issue; one of the top services in the business is Eureka, a Netlifx-provided open source project. With Eureka, the old localhost::8081/stock may now be written as watcher/stock; this introduces an additional degree of abstaractization.

Each microservice has its own database (3.1); each database is containerized and operates on their own ports (3.2) on a private network devoted to postgres databases. Furthermore, a docker of pgAdmin is used for development. PgAdmin is a RDBMS that assists developers in better visualizing the cluster of databases. It does not give features to end users, however, it accelerates the process of development tremendously.

port	8085	8081	8082	8083	5050	5433	5434	5435	5436	8761
name	LIT	Watcher	Portfolio	Social	pgAdmin	PG LIT	PG Watcher	PG Portfolio	PG Social	Eureka
	gateway	**Microservices**			**Databases**					**Service Discovery**

Table 3.2: Port allocation for core application

3.3 Docker

To make the application runable anywhere and simply configurable, LIT employs a docker architecture in which each microservice and service uses docker containers. The content in figure 3.2 is given in broad terms for clarity and simplicity.

Each service and microservice operates on an internal port, the ports shown in Figure 3.1 are the exposed ports. In this project, each service has its own internal network, with the exception of the databases. Databases link with one another through bridging on the same network, which makes it possible for pgAdmin to do basic manipulations on databases from the outside easliy.

Because each database is on a docker, each postgres database runs on its default port, 5432. Because the operating system does not permit port allocation on the same port, they must be exposed on different ports. This allows for very flexible port allocation, since the source is on a stable port, and on demand, this container may be duplicated and exposed on available ports, hence achieving horizontal scalability with relative ease.

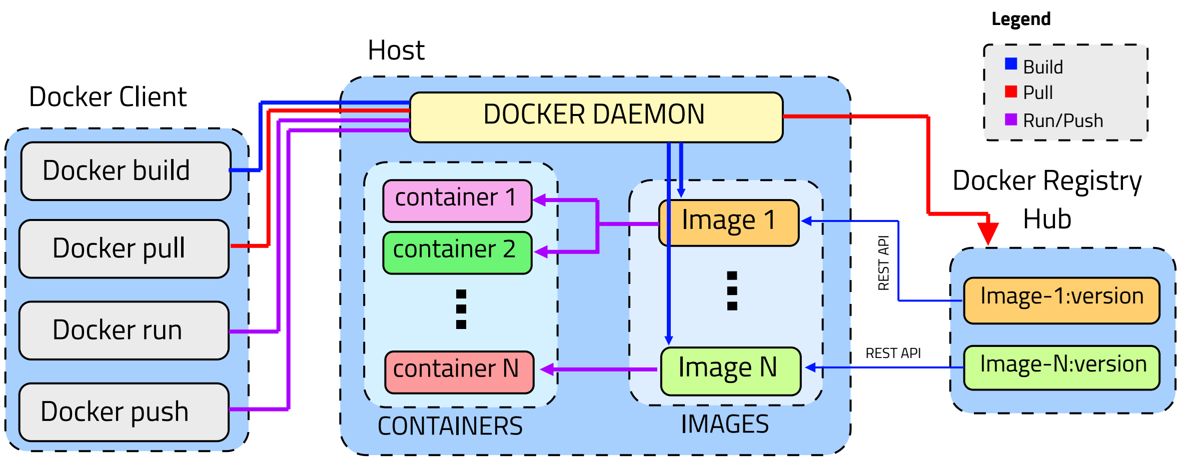


Figure 3.2: Docker arhitecture

Docker client is a the way that the developer intereact with Docker. The Docker client can connect to multiple Docker Daemon, so we are not limited at one application at a time. The following is a list of commands that developers are able to send through the Docker Client:

- Docker build
- Docker pull
- Docker run
- Docker push

"A Docker registry stores Docker images. Docker Hub is a public registry that anyone can use, and Docker is configured to look for images on Docker Hub by default. You can even run your own private registry." [23, Docker handbook p.1].

Docker Hub provides official images and plugins, as well as certified and open source ones, and organizes them by operating systems and architectures. The list of default architecture offered by Docker Hub are:

- ARM
- ARM 64
- IBM POWER
- IBM Z
- PowerPC 64 LE
- x86
- x86-64

LIT use x86-64 images for its images. One of the images used is postgres, that can be manually download with the command `docker pull postgres`, or can be added in docker file to be downloaded at run. The desired image version is specified after the image name, for instance, in LIT, the postgres image is postgres:14.2, therefore the version is 14.2. If a developer want always the most recent version, the developer might use the keyword latest, on previous example would be postgres:latest.

"When you use the docker pull or docker run commands, the required images are pulled from your configured registry. When you use the docker push command, your image is pushed to your configured registry." [23, Docker handbook p.1]

Chapter 4

Implementation Details

4.1 Data Gathering and Processing

This section describes the techniques used to acquire data and how to transform it into a usable format.

There are two types of data in Lean Into Trading: those obtained by web scraping from financial sources and those derived from public databases. Yahoo Finance is the scraped site for financial data, whereas Kaggle is the source for public datasets. Implementation details are provided in sections 4.1.2 and 4.1.1.

The received data is processed using the pandas package. Pandas are discussed in further depth in the Technology Chapter - Section Machine Learning - Subsection Pandas 2.6.2

In addition to web scraping and public databases, third-party APIs may be used. Webscraping and datasets are used due to cost, since the bulk of third-party APIs are prohibitively expensive and those that are free or have free plans are quite restricted. The cost factors are shown in table 4.1.

4.1.1 Kaggale datasets

Prior to this point, we've just discussed financial APIs, but when it comes to news, a specific API becomes necessary. As financial APIs, news APIs may be highly costly, ranging from hundreds to thousands of dollars each month [19, Newsapi Pricing for API], [20,

API	Free Plan	Full Plan	
	Non-Comercial	**Non-Comercial**	**Comercial**
polygon.io	5 API Calls / Minute 2 Years Historical Data Just US market	200$ / month	$3000 / month
finnhub.io	60 API Calls / Minute 1 Year of Historical Data Just US market Real Time Updates	$1000/month	Unspecified
financial modeling prep	250 API Calls / Day 1 Year of Historical Data Just US market	$49/month	Unspecified
IEX cloud	No details	$19/month	$2699/month
Intrinio	Does not have	No details	$1000/month

Table 4.1: Financial API providers prices [14],[15],[16], [17], [18]

Media stack News Data Subscriptions].

Because of the expense, the restricted access to data, and the fact that it is more difficult to obtain data. As a result of that high-quality data is scattered throughout a large number of websites, it is difficult to web-scrape the data. Since each site has its unique implementation and structure, the litapplication should have various scrapers to collect data from different sites.

litutilizes the reputable public datasets provided by Kaggle. Kaggle is one of the largest platforms of its sort, and is used globally by the scientific community for research purposes.

Dataset used by LIT is "Daily Financial News for 6000+ Stocks" which contains over 840000 news title for more than 6000 stocks.

The document is in CSV format, and in table 4.2 we can see a sample data from the file.

index	ticker symbol	title	date
1	A	Stocks That Hit 52-Week Highs On Friday	2020-06-05 10:30
2	A	Stocks That Hit 52-Week Highs On Wednesday	2020-06-03 10:45
• • •			
1413848	ZX	China Zenix Auto International Opens For Trading at $6.00; IPO Price Set at $6.00	2011-05-12 09:36

Table 4.2: Raw Kaggale dataset

As seen in table 4.2, the raw data lacks sentiment analysis for each news headline. This will be accomplished with the assistance of the Pandas package and the LIT Sentiment Analysis Service.

4.1.2 Yahoo Finance Gathering

Yahoo finance is a financial website that contains essential information for the LIT project. Some esential details include historical data, profile, financials, holders.

The historical data for a certain stock covers tickers from the date of its IPO to the present. This data includes the date of the ticker followed by its price at the opening of the stock exchange, its highest price, its lowest price, its closing price, transaction volume, and a price adjustment depending on corporate activities **??**.

Date	Open	High	Low	Close	Adj. Close	Volume
May 26, 2022	24.15	24.99	23.95	24.88	24.88	526664

Table 4.3: The hitorical data of Atos SE (ATO.PA) on May 26, 2022 [21]

Profile information includes the company's location, website, phone number, sector, industry, number of employees, current chairman of the Executive Board, CEO, CFO, etc.

Number of major holdings, insider roster and insider transaction are listed in Holders section.

The whole dataset is gathered using web scraping. Using HTML elements, id, header, and hyperlink manipulation, a Python script retrieves data. An example of a frequently seen page on by LIT is the history page, which can be accessed through the URL https://query2.finance.yahoo.com/v8/finance/download/X, where X is the symbol of a certain stock. This is one of the simplest methods to download Yahoo Fiance data since it works like an API request, but instead of providing a JSON, it returns a CSV, which is better because it is simpler to parse the CSV in a pandas Dataframe container for further processing.

By default, the historical download link contains the company's data since its first public offering. In certain instances, data must be filtered, link modification is required in order to accomplish this.

The following example 4.1 demonstrates how to modify a hyperlink to receive the whole data folowed by 4.2 that filter data for the stock symbol SIE.DE.

$$base\_url = https://query2.finance.yahoo.com/v8/finance/download/SIE.DE \tag{4.1}$$

Following the base URL are two parameters denoting the beginning and end of the interval as "period1" and "period2", respectively. Each period is formatted in Unix. In 4.2 "period1" is 27 May 2021 in Unix and "period2" is 27 May 2022.

$$period\_data = base\_URL/period1?period1{=}1622073600\&period2{=}1653609600 \tag{4.2}$$

Optionally we can select the frequency of the data with the help of the parameter "interval" as seen in 4.3. The options are:

- **1d** - Daily
- **1wk** - Weekly
- **1mo** - Monthly

$$frequency = period\_data\&interval{=}1wk \tag{4.3}$$

The default frequency is 1d, which corresponds to the daily value of tickers.

4.1.3 Pandas processing

Pandas is used to manipulate previously obtained data, with its primary advantage being the ability to natively parse CSV files into a DataFrame type.

Pandas efficiently remove unnecessary data by removing rows and columns and, perhaps most critically, enable developers in writing code that is simple to comprehend. In figure 4.1, the technical indicator moving average is written in a single line of code and is straightforward to comprehend. Initial data consists just of a stock's ticker and news headlines; all other forms of data are obtained via processing.

```
data['\acrshort{ma}7'] = data.rolling(window=7).mean()
```

Figure 4.1: 7 Days Moving Average with Pandas

4.2 Sentiment Analysis

4.2.1 NLP

The automated manipulation of natural language by software, such as voice and text, is an example of natural language processing. This is a wide definition of the term. The method in which we, as humans, interact with one another is referred to as natural language.

The phone assistant is a real-world software that uses natural language processing (NLP). Users communicate with the phone assistant in a natural manner, not using predetermined sentences to obtain a response, but rather complex ones such as "Find a free 30-minute window in my schedule to set up a meeting with Alexandra." In the following sentences, the assistant will comprehend not only the context, which is to schedule a phone appointment, but also that Alexandra is a person, most likely a contact phone, and that it should not interfere with other current plans.

Use cases of NLP:

- Information Retrieval
- Sentiment Analysis
- Translations
- Question Answering

- Classifying
- Spell checking
- Predict next word in a sentence or in a empty given space

LIT uses BERT to generate a sentiment analysis of a given headline; subsection 4.2.2 provides further information on BERT's inner workings.

4.2.2 BERT

Bidirectional Encoder Representations from Transformers is a cutting-edge methoNLP that makes use of a multi-headed attention in order to provide a highly contextualized result based on the input that is provided.

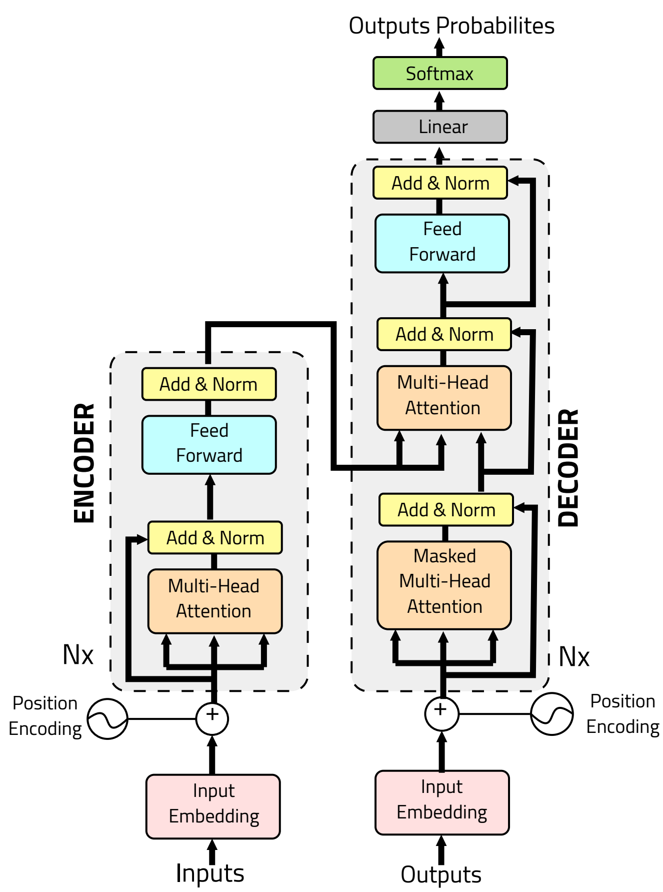


Figure 4.2: BERT Model [1, Bert paper]

The encoder is made up of a stack that has $N = 6$ layers that are all the same. Each of the layers consists of two sublayers. The first is a multi-head self-attention mechanism, while the second is a straightforward, position-wise completely linked feed-forward network. After establishing a layer normalization, we begin by establishing a residual link around each of the two sublayers.

In the same way as the encoder, the decoder is built up of a stack of $N = 6$ layers that are all the same. This will be in addition to the two sub-layers that are already present within every encoder layer. In a manner similar to that of the encoder, we make use of residual connections all around each of the sublayers, and we then normalize the layers. In addition, we make adjustments to the decoder stack's self-attention sub-layer so that positions won't be able to pay attention to those that come after them.

One way to think of an attention function is as a mapping between a query and a collection of key-value pairs to an output, with the query, the key-value pairs, the result, and everything else being vectors. The result is determined by computing a weighted total of the values, with the weight allocated to each value being determined by a compatibility function of the query with the key that corresponds to it.

One of the most important aspects of BERT is the Multi-Head Attention, which can be understood as a collection of Attention functions offered to a single Input embedding. Because of this, each head is able to calculate specific things; for example, one head could determine the location of the proposition in a sentence, while another head could determine the subject and a verb.

Each atention is implemented by folowing this steps in vanilla BERT:

- Tokenization

  The first step in proecessing text, is to cut it into piece called token, there are many variations on how to do it, but bert use WordPiece tokenazation, this means that tokens roughly to words and punctuation, altough a word can also be split into sevral token if it contains a common prefix or suffix. Words can even be spelled out if they have never been seen before.

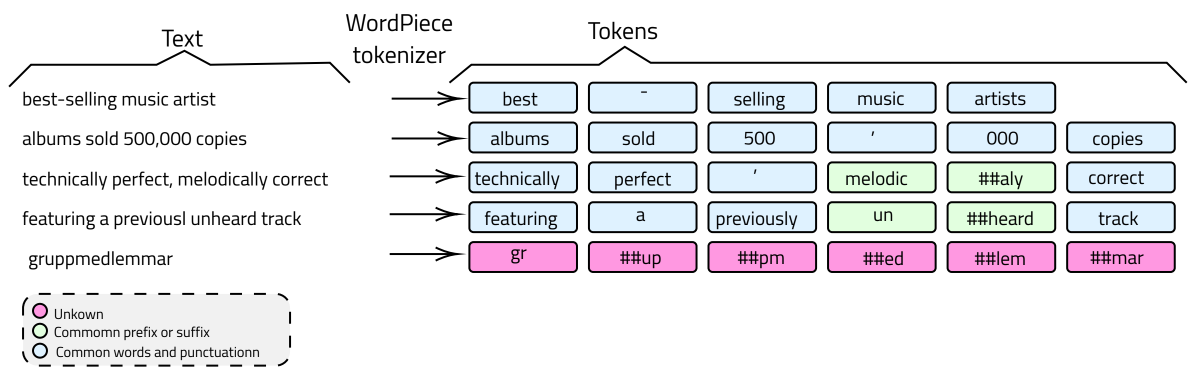


Figure 4.3: Example of word tokenization

- Embedding

  The second step is to associate each token with an embedding, which is nothing more than a vector of real numbers. There are many ways to create from zero this embeddings, fortunetly there are already trained embeddings provided by researchers.

  The emeding of tokens into vectors is an achivment in itself. The value inside an embedding carry information about the meaning of the token, but they are also arranged in such a way that one can perform mathematical operationson them, which corespond to semantic changes, like changeing the gender of a noun 4.4, or the tense of a verb 4.5, or even the homeland of a city 4.6.

$$king + woman - man \approx queen \tag{4.4}$$

$$singng + yesterday - today \approx sang \tag{4.5}$$

$$\begin{array}{ccccccc} italy & + & paris & - & rome & \approx & france \\ \begin{bmatrix} -0.04 \\ \vdots \\ 0.01 \end{bmatrix} & + & \begin{bmatrix} -0.02 \\ \vdots \\ -0.04 \end{bmatrix} & - & \begin{bmatrix} -0.03 \\ \vdots \\ -0.01 \end{bmatrix} & \approx & \begin{bmatrix} -0.04 \\ \vdots \\ 0.01 \end{bmatrix} \end{array} \tag{4.6}$$

- Context

However, embeddings are associated with tokens by a straight dictionary lookup, which means the same tokens always gets the same embeddings, regardless of its context. Here attention come into place, and transforms the default embeddings by analyzing the whole sequence of tokens, so that the values are more representative of the token they represent in the context of the sentence.

- Model

LIT use a highly proficient model on the finance sentences called finBERT. FinBERT is pre-trained on financial data using a huge financial corpus to fine-tune the categorization of financial sentiment. Table 4.4 contains an excerpt from the dataset; the sentences shown there were chosen on the basis of their sentiment score, with one example taken from each of the categories. During this stage of the process, the sentences are categorized as either positive, negative, or neutral. In a subsequent phase, the feature sentences will also be outputted with a decimal number.

Sentence	**Sentiment**
The agreement strengthens our long-term partnership with Nokia Siemens Networks	positive
Ford is struggling in the face of slowing truck and SUV sales and a surfeit of up-to-date , gotta-have cars	negative
Tielinja generated net sales of 7.5 mln euro $ 9.6 mln in 2005	neutral

Table 4.4: Excerpt from the annotated sentences dataset
[ResearchGate - FinancialPhraseBank-v1.0]

This dataset contains over 10000 annotated sentences and can be found on ResearchGate as FinancialPhraseBank-v1.0

4.3 Stock Forecasting

This section is dedicated to the Stock Forecasting part of the project. The stock forecasting is done with the help of a generative adversarial network, which is relatively new approach in the field, the traditional approach on series of data were baseline LSTM, GRU and ARIMA. Even more old than baseline recurent neural networks LSTM and GRU is the manual procedure in which one or more technical analysts analyze corporate technical data and make forecasts. When analysts are specialized in a certain industry or sector (e.g. the Tech industry), this technique may be highly accurate, but it is time consuming and costly.

Before delving deeper into the specifics, let's define several terminology that appear multiple times in this section.

Feed-Forword

The Feed-Forward represents the flow of data through the layers of the neural network, with each layer modifying the data, represented by gates in fig. 4.5 and fig. 4.6, for the subsequent layer until the data reaches an output layer.

Bias vector

Assuming we have a weight w_1, an output y, and a condition $y = w_1 \times x$, the bias vector serves as a collection of intermediate values that facilitate the measurement of network changes. If we wish to measure the changes for this particular y on the previously described condition with $x = 0$, the outcome will always be 0. To be able to assess the changes on different weights, we create a bias: $y = w_1 \times x + w \times bias$. The bias may be arbitrary, but cannot be 0.

Sigmoid layer σ

The sigmoid layer generates values between 0 and 1 that describe the percentage of each component that should be let through. A value of zero indicates "permit nothing to pass through," whereas a value of one indicates "let everything to pass through!"

4.3.1 RNN

Humans don't begin each new thought process from scratch, this is how traditional neural networks do. As you read, each word is interpreted in light of the preceding word's meaning. You don't have to start again from scratch and toss everything away. Thoughts persist in the mind of a human being. RNN overcome this limitation by iterating t, is basically a loop, times and passing a processed input from one network to the next [4.4], which is impossible for standard neural networks to do.

RNNs are used for sequential data, including tickers shown in the LIT project, natural language processing, voice recognition, text summarization, face detection, and video tagging, among others.

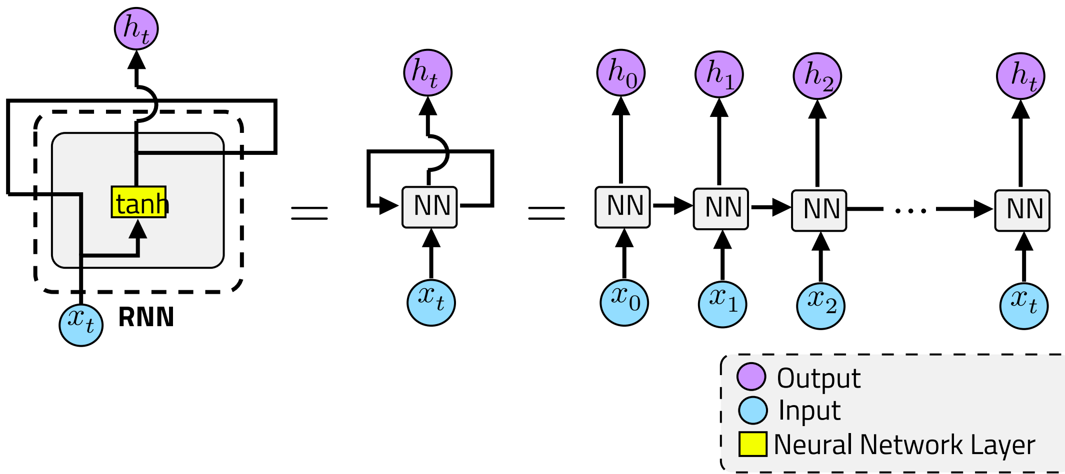


Figure 4.4: Unrolled RNN

As seen in figure 4.4, a section of the NN neural network receives the input x_t and returns the value h_t. A loop permits the transmission of data from one network step to the next. Multiple copies of the same network, each transmitting a message to its successor, constitute a recurrent neural network. The issue with them is that they are incapable of long-term data persistence. The next parts of this section handle this issue using LSTM and GRU.

4.3.2 LSTM

LSTM feature a structure that allows for an excellent data flow. The cell state is analogous to a data highway in that it travels directly across the chain with just two intersections, known as pointwise operations.

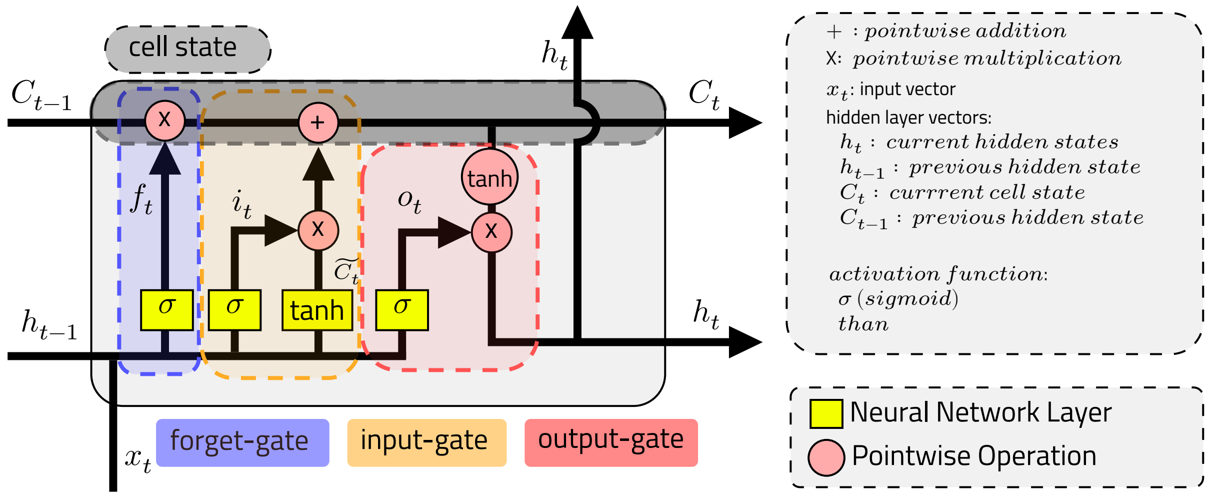


Figure 4.5: LSTM structure

Gates allows the insertion or removal of data from a cell's state, and they may optionally block data that is deemed unnecessary. As an example related to the project's topic, the cell state might contain the market size of a given company symbol and the subject itself, which is the company symbol, in order to employ the proper terminology. When we find new symbol, we wish to forget the old company's market size. The input gate can add up values to a new candidate after we forget the previous candidate by adding $i_t \times \widetilde{C_t}$. Therefore, gates may add, retrieve, and even alter the subject.

Feed-Forward

W_o, W_c, W_i, W_f : *parameter matrices*

b_o, b_f, b_i, b_c : *bias vector*

$$
\begin{aligned}
f_t &= \sigma\,(W_f \times [h_{t-1},\, x_t] + b_f) \\
i_t &= \sigma(W_i \times [h_{t-1},\, x_t] + b_i) \\
o_t &= \sigma\,(W_o \times [h_{t-1}, x_t] + b_o) \\
\widetilde{C_t} &= tanh(W_c \times [h_{t-1},\, x_t]\ b_c) \\
Ct &= f_t \odot C_{t-1} + i_t \odot \widetilde{C_t} \\
h_t &= o_t \odot tanh(C_t)
\end{aligned}
$$

4.3.3 GRU

There are several variants of LSTM, the most of which are minor changes of the one being described, which is the vanilla kind. GRU and peephole LSTM are two of the most common ones. The peephole connections add two more connections to the state cell; these connections are directly linked to all three gates of the vanilla LSTM, allowing the gates to see the cell states.

GRU differs from LSTM in that it merges the input gate and the forget gate into a single gate called the update gate. Also, it is evident that GRU no longer has two outputs, since the hidden state and cell state have been unified. GRU is one of the most unique interpretations of LSTM and is simpler than LSTM itself. Currently, (2022) is one of the most popular RNNs in the research community.

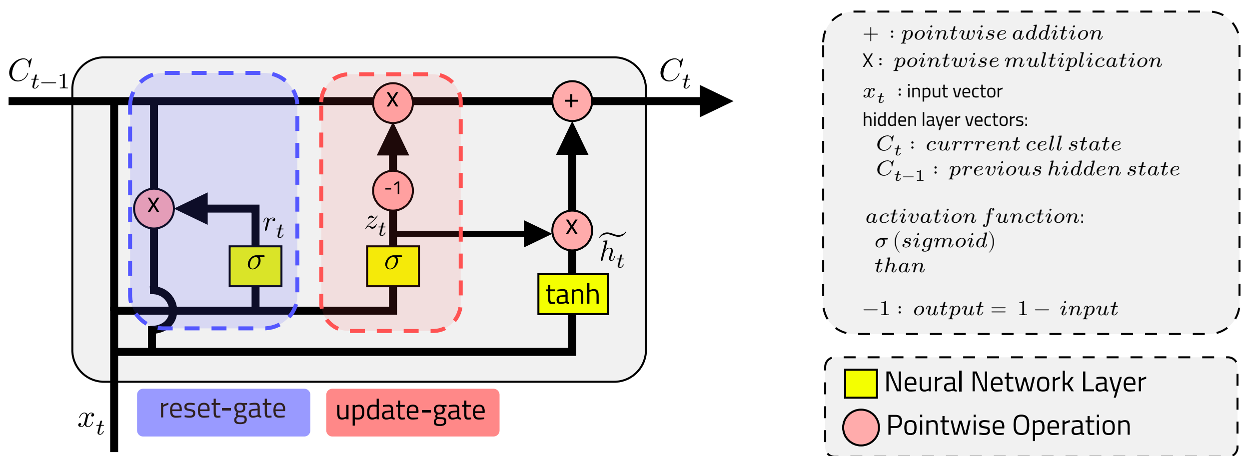


Figure 4.6: GRU structure

Feed-Forward

W_z, W_r, W_h : $parameter\ matrices$

b_h, b_z, b_r : $bias\ vector$

$$
\begin{aligned}
z_t &= \sigma(W_z \times [h_{t-1}, x_t] + b_z) \\
r_t &= \sigma(W_r \times [h_{t-1}, x_t] + b_r) \\
\tilde{h}_t & tanh(W_h \times [r_t \odot h_{t-1}, x_t] + b_h) \\
h_t &= (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t
\end{aligned}
$$

4.3.4 Training data

This subsection is dedicated to the trading data. The data utilized for trading and forecasting is:

- Indexes Closing Price
  - S&P 500
  - FTSE 100
  - NIKKEI 225
  - BSE SENSEX
  - Russell 2000
  - HSI
  - SSE Composite Index
  - VIX
- News sentiment Analysis
- Commodities
  - Gold
  - Crude Oil
- Stock Exchanges Closing Price
  - NASDAQ
  - NYSE
- USD index
- Technical Indicators
  - 7 Days Moving Average
  - 21 Days Moving Average
  - Moving Average Convergence Divergence
  - Exponential Moving Average
  - Bollinger bands
  - Momentum
- 3 similar companies on the stock market
- Ticker data: Date, Open Price, Highest Price, Lowest Price, Close Price and Volume

All of the technical indicators that were discussed are based on the closed price, which is the price of a particular stock at the end of the trading day, of MAIN stock on the New York Stock Exchange. In figure 4.7, we can see the fluctuation in the close price between October 09, 2007 (the day the stock was initially offered for public trading) and May 20, 2022.

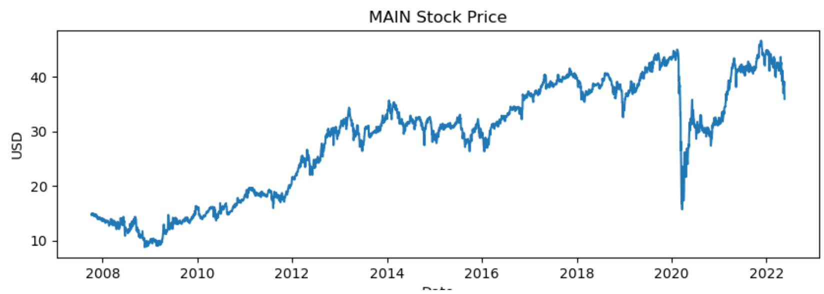


Figure 4.7: MAIN Stock Price
Ploted with Matplotlib

Techincal Indicators

Figure 4.8 depicts the correlation between the closing price of Main Street Capital Corporation and technical indicators during 400 days of open US stock market [1]. The time shown in fig. 4.8 extends from October 23, 2020 to May 25, 2022.

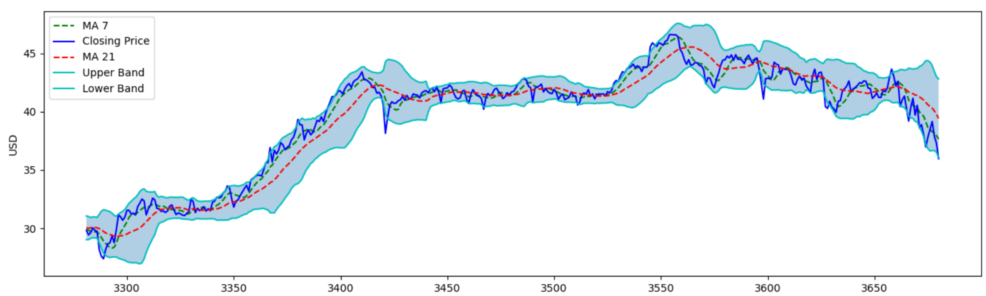


Figure 4.8: Tehnical indicators for MAIN - 400 days
Ploted with Matplotlib

Figure 4.8 demonstrates that both MA 7 and MA 21 exhibit a price direction trend and indicate if a certain stock's price movements are abnormal. The difference between MA 7 and MA 21 is the number of days; one measures stock movement based on the last 7 days, while the other uses the past 21 days. The most typical intervals used by analysts for their forecasts are 15 days, 20 days, 30 days, 50 days, 100 days, and 200 days. The shorter the period, the more sensitive the movement is to price changes.

[1] The US Stock market is not open every day of the year, is closed every weekend and on certain holidays such as Christmas, Labor Day, Independence Day and others

These moving averages are also known as the Simple Moving Average (SMA) and are calculated by the equation 4.7 where P_x is the closing price at the end of the market day x with $x \in [first\_day, first\_day + n]$ and the n is the number of days in interval.

$$SMA = \frac{P_1 + P_2 + P_3 + \cdots + P_n}{n} \tag{4.7}$$

LIT also use the Exponential Moving Average, a sort of moving average that provides a laverage of the most recent data, in this instance closing prices, over a specified period.

In EMA, the weight is calculated by adding 1 to the number of selected days and dividing the "smoothing factor" by the result; the "smoothing factor" is an arbitrary value, although the value 2 is often preferred. Equation 4.8 show the calculation of a weight for a moving average of 21 days and equation 4.9 show the same calculation for an interval of 15 days.

$$W_{21} = \frac{smoothingfactor}{interval\ + 1} \ = \ \frac{2}{22} \ = \ 0.0909 \ = \ 9.09\% \tag{4.8}$$

$$W_{15} = \frac{smoothingfactor}{interval\ + 1} \ = \ \frac{2}{16} \ = \ 0.125 \ = \ 12.5\% \tag{4.9}$$

So SMA has an equal weight over all values within its period, whereas EMA has a greater weight over the most recent values. The Exponential Moving Average places more emphasis on shorter time periods. Equation 4.10 show the EMA equation, l represent the last day from the interval, P represent the prices, so P_l represent the last price in given interval.

$$EMA = P_l \times W_n + EMA_{l-1} \times (1 - W_n) \tag{4.10}$$

Using the median, low, high, or open price instead of the closing price results in slight variances of the EMA that are also used by analysts.

The upper and lower bands are an addition to the moving average approach, representing the addition and subtraction of a standard deviation represented by equation 4.11.

$$SD \ = \ \sqrt{\frac{\sum \mid x \ - \mu \mid^2}{n}} \tag{4.11}$$

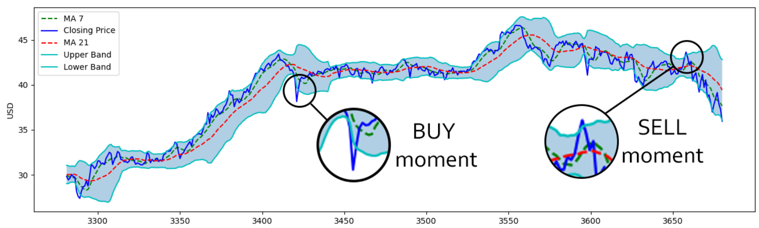


Figure 4.9: Upper and lower bands opportunities

LIT adopts an upper and lower band with a standard deviation of 2 and a 21-day moving average. This fluctuates, and sometimes the stock price exceeds the acceptable range. This often signifies an excessive purchase or sale of a stock. When an overbuy occurs, the investor should sell, and when an oversale occurs, they should buy, since the trend should return to normal and the investor should benefit. In figure 4.9 we can see this action taking place.

Analysts use the Moving Average Convergence/Divergence indicator to determine the stability, direction, volatility, and longevity of a stock's trend especially on times of high market volatility. MACD is often used in conjunction with other technical indicators, such as the histogram, as seen in Figure 4.10. The histogram is composed of lines that run vertically and represent the difference in value that exists between the MACD lines.

Moving Average Convergence/Divergence is calculated by substracting the Exponential Moving Average of 26 days from EMA of 12 days, the EMA equation is explanied in equation 4.10

$$MACD = EMA_{12} - EMA_{26} \tag{4.12}$$

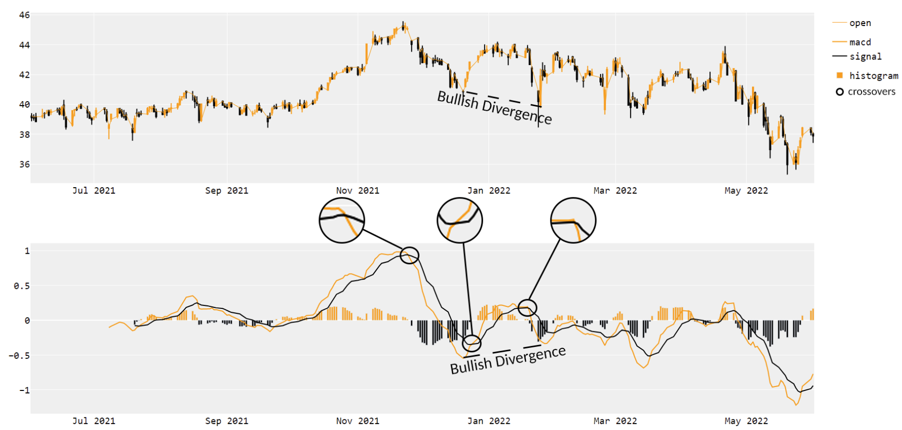


Figure 4.10: MACD for stock MAIN
Ploted with Plotly python package

The signal line is an indicator for the MACD that assists in the viewing of MACD's own turn movements. Whenever these two lines intersect, as shown by the "crossovers" in Figure 4.10, there is a possible buy or sell entry opportunity in the stock market.

A bullish divergence occurs when the MACD creates two rising dips while the price forms two falling lows. When the long-term trend remains higher, this is a reliable bullish indicator.

In figure 4.10, there is no bearish divergence, but this divergence may be seen as two dropping highs that correspond with two rising highs, in contradiction to the bullish divergence.

The easiest method to remember them is to integrate the price fluctuation line with the macd panel line. This method result in two symbols $bearish\ divergence$ marked as $>$ and $bullish\ divergence$ marked as $<$.

When the MACD line rises over the signal line, it indicates a purchasing opportunity since the stock price is lower than it should be.

When the MACD line falls below the signal line, it indicates a potential selling opportunity.

Indexes

A mechanism for monitoring the performance of a group of stocks, bonds, or other assets. Certain organizations provide bundles of assets that mimic these indexes, so these indexes may theoretically be purchased, but they are also excellent indications for analysts.

Oftentimes, an index serves as a benchmark for other stocks, exchange-traded funds, and mutual funds, so companies can justify a downward trend by stating that a particular index is down, so it is normal for their company to be down as well; flip side, if the trend is upward, they can claim that they beat a particular index in value, indicating that they are a great value company.

Indexes used in LIT:

- S&P 500

  S&P 500 is one of the most recognizable US indexes and represents the 500 largest US corporations by market capitalization and stability 4.5.

  S&P 500 is regarded as a respectable index since it consists of companies with solid financials that are often owned by individual investors. This index represents about 80% of the total value of the US stock market.

Company	Price per stock USD	Capital USD
APPLE INC.	148.72	2.41 T
MICROSOFT CORPORATION	272.42	2.04 T
ALPHABET INC.	2277.84	1.50 T
• • •		
UNDER ARMOUR,INC.	10.33	4.674 B

Table 4.5: S&P 500 composition June 1, 2022
[MarketScreener.com]

- FTSE 100

  The FTSE 100 is a market capitalization-weighted index of blue chip businesses listed in the United Kingdom. The index follows the trading activity of the 100 biggest businesses listed on the London Stock Exchange that are able to pass a screening based on their liquidity and size 4.6.

Blue chip stocks are those of extremely big, well-recognized corporations with a lengthy track record of solid financial success. Blue chip stocks are often expensive since they have a solid reputation and are frequently industry leaders. These stocks are recognized for their ability to withstand challenging market situations and provide substantial returns in favorable market conditions.

Company	Price per stock GBX	Capital GBX
SHELL PLC.	2361	176 B
AstraZeneca PLC.	10442	161B
HSBC HoldingsPLC.	529.60	107 B
• • •		
ITV PLC.	70.60	2 B

Table 4.6: FTSE100 composition June 1, 2022
[London Stock Exchange]

- NIKKEI 225

NIKKEI 225 and FTSE 100 are both made of blue chip companies.NIKKEI 225 Is one of the most esteemed Japanese stock indexes. This index's constituent stocks are all traded on the Tokyo Stock Exchange 4.7.

Company	Price per stock JPY	Capital JPY
TOYOTA MOTOR CORP.	2	35 T
SONY GROUP CORP.	12	15 T
NIPPON TELEGRAPH AND TELEPHONE CORP.	3.9	14 T
• • •		
TOHO ZINC LTD.	2.38	32 B

Table 4.7: NIKKEI 225 composition June 1, 2022
[MarketScreener.com]

- BSE SENSEX

BSE SENSEX is a fairly modest index in terms of the number of companies it contains, with just 30 stocks 4.8. Its constituent stocks are the largest and most frequently traded in India on Bombay Stock Exchange. When the index was formed, there were just 14 stocks included.

Company	Price per stock INR	Capital INR
Reliance Industries	2.683	18.17 T
TATA Consultancy	3.402	12.45 T
HDFC Bank	1.384	7.68 T
• • •		
Dr. Reddy's Labs	4.336	719 B

Table 4.8: BSE SENSEX composition June 1, 2022
[Investing.com]

- Russell 2000

RUSSELL 2000 is, like S&P 500, composed of US-traded stocks. S&P 500 is a good index for large companies, while RUSSELL is a superior index for small companies, as it is comprised of one of the smallest companies in the United States of America 4.9.

Company	Price per stock USD	Capital USD
Ovintiv INC.	55.99	14.71 B
Antero Resources	42.88	13.34 B
Chesapeake Energy INC.	97.38	12.39 B
• • •		
Kaleido Biosciences	0.0833	3.55 M

Table 4.9: Russell 2000 composition June 1, 2022
[fknol market screener]

In table 4.9, on the last position, is Kaleido Biosciences, which was one of the worst performing companies in the United States in 2021, losing almost 97 percent of its value. This type of company would never appear in an index like S&P

500, but this is not necessarily a bad thing, because it shows the true face of small companies that struggle to compete in a very harsh market, and when combined with S&P 500, provides a better perspective of the economy as a whole, not just a cherry picked economy.

- HSI

HSI is an index that follows 45 Hong Kong Exchange-listed firms. These companies are among the biggest on the stock market and account for 67 percent of the exchange's total value. First in terms of market capitalization, Tencent's 3.45 trillion Hong Kong dollars comprise 15.24 percent of the HSI's 22.48 trillion HKD value.

Company	Price per stock HKD	Capital HKD
Tencent	356	3.43 T
Alibaba Group	91.70	1.91 T
China Construction Bank	5.73	1.44 T
• • •		
Aac Techinologies Holdings INC	17	20 B

Table 4.10: HSI composition June 1, 2022
[aastocks market screener]

- SSE Composite Index

Unlike other indexes, the SSE index does not categorize stocks based on a particular concept. RUSSEL groups companies by tiny market capitalizations, S&P takes the most stable and large corporations, as do BSE SENSEX and NIKKEI 225, while SSE Composite Index includes all stocks listed on the Shanghai Stock Exchange.

LIT use the more broad SSE Compsite Index as opposed to the more specific SSE 180, SSE 50, and CSI 300, CSI 200, and CSI 100 indices that categorize stocks from Mainland China stock markets according on market size. SSE Indexes monitor just the Shangai Stock Exchange, while CSI monitors both the Shangai and Shenzhen Stock Exchanges.

Company	Price per stock CNY	Capital CNY
Industrial And Comercial Bank of China	4.63	1.70 T
China Construction Bank	5.95	1.25 T
Agriculture Bank China	3.02	1.05 T
• • •		
Aac Techinologies Holdings INC	0.31	185.34 B

Table 4.11: SSE composition June 1, 2022
[Investing.com]

- VIX

  Instead of monitoring its own list of stocks, the VIX monitors the options of the S&P 500. Options are contracts that provide investors the right, but not the obligation, to purchase or sell underlying assets. They are exchanged on a distinct market known as the option market, not the stock market. They are interesting to investors because they provide the purchase or sale of certain equities at a predetermined price over a specified time period. They also allows betting on stock declines, so an investor can profit if a stock's price is falling.This kind of contract does not allow for long-term holding and accepts extremely large laveages. Typically, when an investor uses them, he or she is certain of a certain stock market trend.

  VIX is the sole index used by LIT that does not track stocks but rather stock options.

3 similar companies on the stock market

In the dataset for training and forecasting, there are another three companies comparable to the one provided. The companies are decided by their industry and price similarities. The price similarity is derived using a 15 percent variance up or down from the original price. If no companies meet the price similarity criteria, the most valuable stocks in their sector are selected.

Consider MAIN Street Capital, which works in the Financial Services sector and has a market valuation of $2.76 billion. LIT's association algorithm selects the following three firms: Ameris Bancorp, Artisan Partners Asset Management Inc, and Associated Banc-Corp by their sector and price diffrence as seen in table 4.12.

	Company Symbol	Market Cap	Deviation from MAIN	Sector
ORIGINAL	MAIN	2.765B	0%	Financial Services
SELECTED	ABCB	3.125B	+ 11.52 %	
	APAM	2.573B	- 6.94 %	
	ASB	3.076B	+ 10.11 %	

Table 4.12: Similar stocks to MAIN stock as recomanded by LIT

Now consider the edge situation, in which three stocks are not discovered inside the required price interval. Apple was used as an example since this is essentially an edge case for companies with very large market caps as seen in table 4.13.

	Company Symbol	Market Cap	Deviation from AAPL	Sector
ORIGINAL	AAPL	2.41 T	0%	Financial Services
SELECTED	MSFT	2.04 T	- 15.35 %	
	GOOG	1.50 T	- 37.76 %	
	GOOGL	1.50 T	- 37.76 %	

Table 4.13: Similar stocks to AAPL stock as recomanded by LIT

Alphabet is represented by both GOOG and GOOGL, although they trade separately since they are different sorts of shares. In this particular instance, the price per share of the stocks is rather comparable; nevertheless, it is important to keep in mind that this is not always the case for other firms that offer different classes of shares. GOOG shares are classified as class A, whilst GOOG shares are classified as class C. The primary distinction between the two types of shares is that GOOGL shares come with voting rights, allowing shareholders to have a say in how the firm is run.

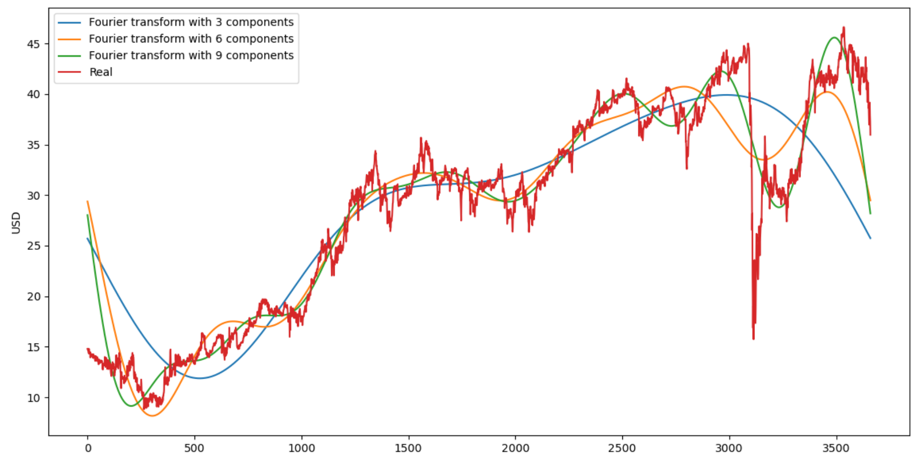


Figure 4.11: MAIN (close) stock prices and Fourier transforms
Ploted with Matplotlib

The Fourier series are used to see the price trend over time.

The interval across which a Fourier series is applied has a direct bearing on how sensitive the series is to changes in price. The smaller the interval, the more sensitive the series is to price shifts.

LIT includes three Fourier series, Three-day, Six-Day, and Nine-Day Fourier. As can be seen in figure 4.11, the fourier of three days adjusted to the actual price in a more abrupt manner compared to the fourier of nine days.

4.3.5 GAN

Goodfellow first created GAN with the intention of using it to generate images [5, Generative adversarial nets. Advances in neural information processing systems 27]; but, as can be seen in LIT, it is also capable of extremely successfully generating time series. A GAN typically consists of two components: a generator and a disseminator.

The Discriminator is the component that differentiates false data from actual data, while the Generator creates fake data as closely as possible to the real data. When the Discriminator no longer recognizes genuine data compared to those supplied by the Generator, it is understood that the Generator produced data sufficiently similar to reality.

During the first cycle of the process, the generator takes the vector representing the latent space and generates synthetic data, which is then sent to the discriminator. The discriminator assigns classifications to the data, and then it sends any errors it finds on to the generator so it may be used to adjust the weights. Therefore, the generator becomes better at creating the data with each subsequent iteration of the process.

LIT GAN model can be seen in figure 4.12, the approach is modeled as a MinMax game. MinMax is a phrase often used in artificial intelligence and many other domains, such as statistics, decision theory, and game theory, for reducing the potential loss in the worst-case scenario.

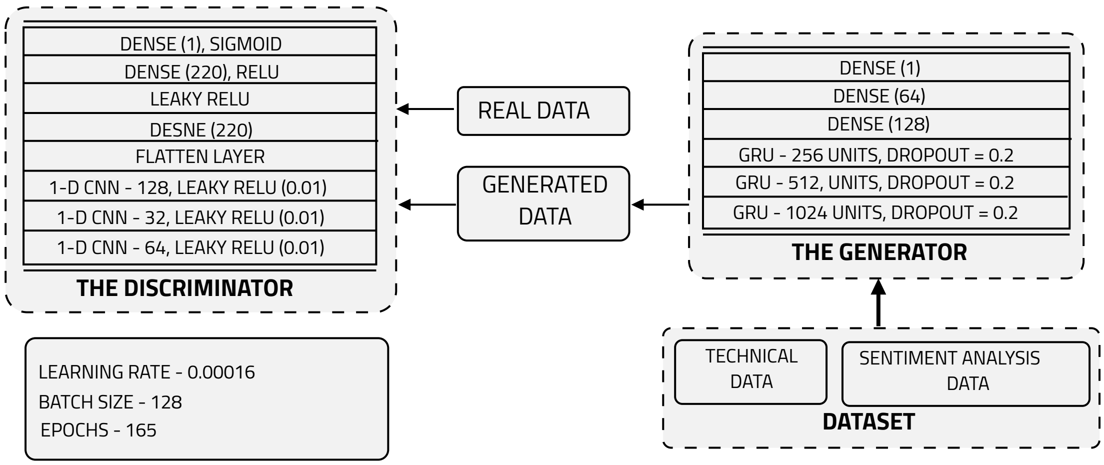


Figure 4.12: GAN Model

The Chossen discriminator is a Convolutional Neuronal Network with a single dimension. The first section consists of three layers of 32, 64, and 128 units, each having a LeakyRelu activation function and an Alpha value of 0.01.

The kernel size for the first CNN is three, followed by five for the second CNN and third one.

At the end of both Discriminator and Generator dense layers are used in order to change the dimension of the vectors to 1.

Tensorflow allow to a easy way to transcribe the model 4.12 in a clean code easy to understand and use as seen in figure 4.13 and figure 4.14.

```
model = Sequential()
model.add(GRU(units=1024,recurrent_dropout=0.2))
model.add(GRU(units=512, recurrent_dropout=0.2))
model.add(GRU(units=256, recurrent_dropout=0.2))
model.add(Dense(128))
model.add(Dense(64))
model.add(Dense(1))
```

Figure 4.13: Simplified python Generator code from LIT

```
cnn_net = tf.keras.Sequential()
cnn_net.add(Conv1D(32, kernel_size=3
            , activation=LeakyReLU(alpha=0.01)))
cnn_net.add(Conv1D(64, kernel_size=5
            , activation=LeakyReLU(alpha=0.01)))
cnn_net.add(Conv1D(128, kernel_size=5
            , activation=LeakyReLU(alpha=0.01)))
cnn_net.add(Flatten())
cnn_net.add(Dense(220))
cnn_net.add(LeakyReLU())
cnn_net.add(Dense(220, activation='relu'))
cnn_net.add(Dense(1, activation='sigmoid'))
```

Figure 4.14: Simplified python Discriminator code from LIT

The equation shown in 4.13 represents the GRU generator of the proposed model. Alternatively LSTM can be chose as generator, the difference between them at sections 4.3.2 and 4.3.3.

$$g_\theta : Z \to X \tag{4.13}$$

The symbols θ indicate the weights of the neural network, while the symbols Z represent the combined data of conventional technical analysis and news sentiment, with values ranging from -1 to 1.

In equation 4.14, we mathematically demonstrate how the generator attempts to produce data that is so near to the truth that the discriminator cannot distinguish between the genuine data and the data made by the generator.

$$\min_{\theta} \max_{w} [\log(D_w(X)) + \log(1 - D_w(g_\theta(Z)))] \tag{4.14}$$

These equations, 4.15 and 4.16, reflect the training process. This equations use cross entropy in binary form. In this equation, the generator and discriminator are in opposition, each moving in the opposite direction of the other.

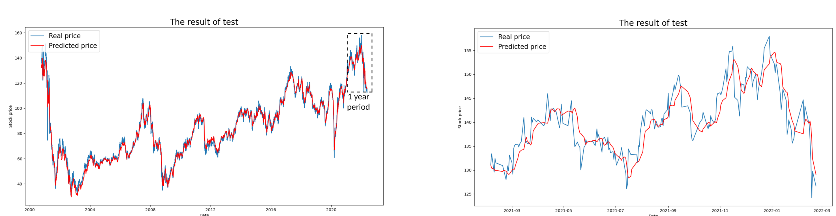
$$w \to w + \alpha \sum_{1}^{n} \nabla_w [\log(D_w(X_i)) + \log + \log(1 - D_w(g_\theta(Z_i)))] \tag{4.15}$$

$$\theta \to \theta - \alpha \sum_{1}^{n} \nabla_\theta [\log(1 - D_w(g_\theta(Z_i)))] \tag{4.16}$$

Using the Root Mean Square Error(RMSE) measure, actual data is compared to false data in order to evaluate the findings. $Actual$ refers to the original price from the dataset, while $Predicted$ represents the stock price as determined by the GRU generator. In 4.17, α represents the pace of learning (learning rate) and N the number of given days.

$$RMSE = \sqrt{\frac{\sum_{1}^{n}(Predicted_1 - Actual_i)^2}{N}} \tag{4.17}$$

In 4.15a are actual prices and forecast results for SIE.DE stock over a period of 25 years with an resulted RMSE of 3.2595. For the purpose of providing a clearer illustration, a further test was performed on a shorter time interval, namely one year, and its results are depicted in figure 4.15b, where the RMSE was 3.3469.



(a) The prediction over the whole dataset

(b) The prediction over 1 year period

Figure 4.15: SIE.DE real prices vs predicted prices

Chapter 5

Workflow

The flow of data between different services and microservices will be shown in this chapter. In order to have a deeper comprehension of the movement, the processes will be broken down and described for each individual component of the project, namely social, portfolio, and watcher. For a more comprehensive summary, the chapter on architecture sums this up extremely well.

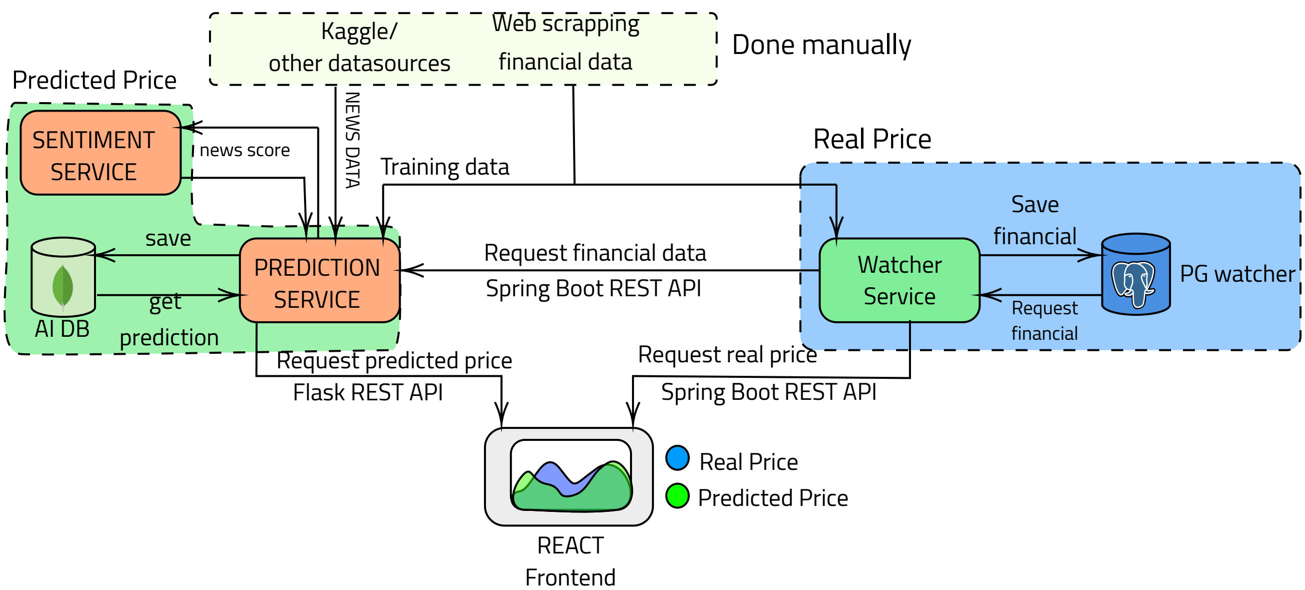


Figure 5.1: Watcher workflow

The processes required to determine the actual and expected price of a given stock are shown in Figure 5.1.

The procedure is launched from the top down, with the administrator setting the required date period for web scraped data that is obtained from Yahoo Finance, inserted into the service Watcher database, and sent as training data to the prediction service. In the case of LIT, this data is manually retrieved from Kaggale, although Kaggale it is not required and other data sources can be used.

The majority of data sources contain little to no information about the news' sentiment score, so LIT's approach is to use news as just text and annotate the data itself. This is accomplished with the assistance of Sentiment Services, which receive a given text and output the text along with a sentiment score between -1 and 1. Sentiment Service classifies -1 as very bad news regarding a certain stock and 1 as extremely positive news; everything around zero is considered neutral.

In figure 5.2, it is shown how the user may interact with this data through a chart, in this case the given stock is MAIN. The user is able to see, zoom in or out, and scroll through the prices of all dates processed by the Prediction AI Service and kept in the Watcher Service. Additionally, it may download the chart as a .PNG or .SVG file.

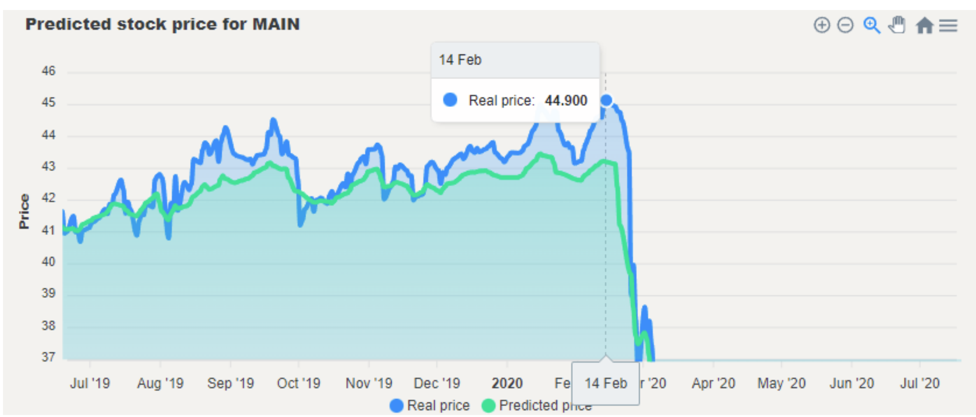
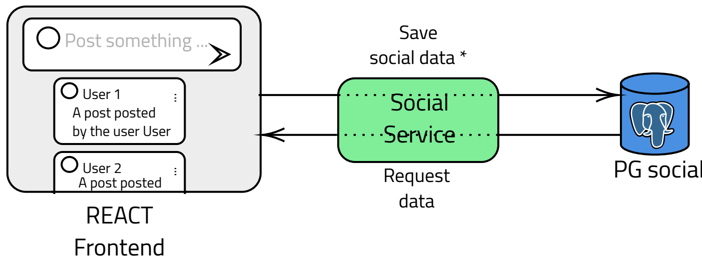


Figure 5.2: Real price compared with Predicted price
[Screenshot from LIT]

As described in Section Architecture, these services function together, but they are not tightly coupled in terms of operability, thus one may operate independently from the others. Therefore, in this chapter, they are described somewhat separately in terms of work flow, but this does not imply that they are unconnected; rather, they are only loosely tied, which is desirable. Without further ado, let's examine the Social and Portfolio sections.



Social data* means everthing related to the user data, and user interecation on the app, such as like, post comments.

Figure 5.3: Social workflow

Social is more straightforward than Watcher in terms of workflow. The React frontend communicates with Spring Boot to access the platform's capabilities and database. This feature allows users to create postings on the site, comparable to those on Facebook or Reddit. The user may respond to the post with a variety of reactions, comment, and reply. Remark is directly related to the post, whereas reply is a comment related to a specific user's comment.

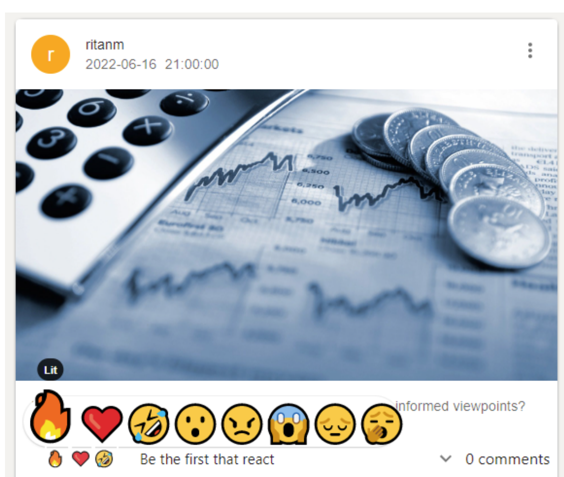


Figure 5.4: Reactions on a post
[Screenshot from LIT]

If a user chooses to utilize this feature, a reaction represents a user's emotional response to a specific post. In recent years, an increasing number of platforms have adopted this advanced approach, including LIT. This project's reactions are LIT, LOVE, AMUSING, AMAZING, ANGER, FEAR, SAD, and BORING as seen in figure 5.4. LIT indicates something like to "super like," or "the finest" is an app signature; the other reactions are self-explanatory.

In the Portfolio 5.5 section, the user may enter transactions like as purchases, sales, and even dividends. After the user has supplied all the data by manual input or CSV file, the data may be shown in charts. The charts are identical to those in the Watcher (see Figure 5.2), plus or minus any available projections for the user's stocks.

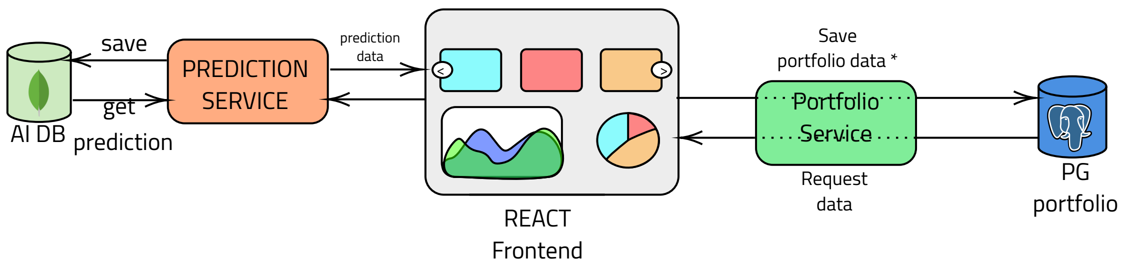


Figure 5.5: Portfolio workflow

Chapter 6

Future Improvements

6.1 Self sufficiency

In the not too distant future, LIT ought to automate some of the human operations that it now employs, such as the downloading of news data from Kaggle by hand, as discussed in Chapter 5. In this regard ought LIT to become more self sufficient. When it acquires a certain number of individuals on the platform, LIT could potentially be able to collect news from its own social network feed.

This approach of self-feeding the Prediction Service with user posts from the Social Service is feasible because, as discussed in Chapters 3 and 5 , the Sentiment Service is loosely connected and is capable of processing any data, including the user posts. This is not possible at this stage of development due to the absence of users on the platform; however, once a sufficient number of users who publish often and diversely have joined, LIT will no longer need the services of third parties.

Financial data, can be download automatically, with ease when this project will be ported in cloud, AWS being one of the best soultions because the platform has a component called Lambda Function, that a acts as a small program that can be triggered at certain moments in time, so the scrapper can be called at the end of each finacial day to collect data. This is the best approach if the application remains non-comercial, if the project become comercial, of course, a third party provider is need. Web scrapping in comercial projects is considered a gray are in terms of legality.

6.2 Deployment

LIT is now operating locally, but as stated in the Architecture 3 section of this document, the project is dockerized and has YML files that explain the settings, so from a deployment viewpoint, the project is ready, but AWS-specific cloud settings will be required in the future. AWS Aurora Serverless is the greatest choice for databases since it consumes resources on demand. AWS RDS is the second alternative, however it is more expensive and less resourceful than Aurora.

Also in deployment we have access on AWS Lambda, which is a tiny program that runs independently from the app and may contact LIT services at certain periods, is a crucial feature for the latter phases of the project. This approach will be incredibly efficient, since the services will not be required to run timers; Amazon Lambda will take care of this.

Bibliography

[1] Jacob Devlin Ming-Wei Chang Kenton Lee Kristina Toutanova: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding (2019)
https://arxiv.org/pdf/1810.04805.pdf
Accessed on: 10-April-2022

[2] Madir J., FinTech: Law and Regulation (Elgar Financial Law and Practice series) (2021, Edward Elgar Publishing)

[3] Neos Thanh, Flask Web Development: Developing Web Applications with Python (Google Commerce Ltd.)
Book preview: https://play.google.com/books/reader?id=oNU-EAAAQBAJ&pg=GBS.PA1&hl=ro
Accessed on: 01-May-2022

[4] Saifedean A., THE FIAT STANDARD The Debt Slavery Alternative to Human Civilization (2021, The Saif House)

[5] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, Aaron Courville, David Warde-Farley, Sherjil Ozair, and Yoshua Bengio. 2014. Generative adversarial nets. Advances in neural information processing systems 27 (2014)

[6] Vikas Bajpai, Anukriti Bansal: Stock price prediction using BERT and GAN (2021)
https://arxiv.org/pdf/2107.09055.pdf
Accessed on: 13-November-2021

[7] Most used web frameworks among developers worldwide, as of 2021
https://www.statista.com/statistics/1124699/worldwide-developer-survey-most-used-frameworks-web/
Accessed on: 25-April-2022

[8] Stackoverflow survey on database usage
https://insights.stackoverflow.com/survey/2021#most-popular-technologies-database-prof
Accessed on: 25-April-2022

[9] Apache Tomcat versions
https://tomcat.apache.org/
Accessed on: 20-April-2022

[10] Font scripts technologies market share
https://www.wappalyzer.com/technologies/font-scripts
Accessed on: 20-April-2022

[11] Font awesome 6 features
https://fontawesome.com/versions
Accessed on: 25-April-2022

[12] PostgreSQL vs MySQL: Is There a Clear Winner?
https://soshace.com/postgresql-vs-mysql-is-there-a-clear-winner/
Accessed on: 01-May-2022

[13] Pandas Source Code - Github Page
https://github.com/pandas-dev/pandas
Accessed on: 15-April-2022

[14] Poligon.Io pricing plans
https://polygon.io/pricing
Accessed on: 25-May-2022

[15] Finhub.Io pricing plans
https://finnhub.io/pricing
Accessed on: 25-May-2022

[16] Financial Modeling Prep Pricing plans
https://site.financialmodelingprep.com/developer/docs/pricing
Accessed on: 25-May-2022

[17] IEX cloud Pricing plans
https://iexcloud.io/pricing/#price-table-section
Accessed on: 25-May-2022

[18] Intrinio Pricing plans
https://intrinio.com/pricing
Accessed on: 25-May-2022

[19] Newsapi.ai pricing plans
https://www.newsapi.ai/plans
Accessed on: 25-May-2022

[20] Media stack pricing plans
https://mediastack.com/product
Accessed on 25-May-2022

[21] Yahoo Finance Atos SE(ATO.PA) company details page
https://finance.yahoo.com/quote/ATO.PA?p=ATO.PA&.tsrc=fin-srch
Accessed on 27-May-2022

[22] Elastic Search official presentation
https://www.elastic.co/what-is/elasticsearch
Accessed on 28-May-2022

[23] Docker overview
https://borosan.gitbook.io/docker-handbook/docker-containers?cv=1
Accessed on 29-May-2022

LIT

ORIGINALITY REPORT

6%	4%	3%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Liverpool John Moores University Student Paper	<1%
2	arxiv.org Internet Source	<1%
3	Submitted to University of Sydney Student Paper	<1%
4	borosan.gitbook.io Internet Source	<1%
5	khuhub.khu.ac.kr Internet Source	<1%
6	examples.javacodegeeks.com Internet Source	<1%
7	Lanxin Zhao, Wanrong Gao, Jianbin Fang. "High-Performance English–Chinese Machine Translation Based on GPU-Enabled Deep Neural Networks with Domain Corpus", Applied Sciences, 2021 Publication	<1%