Stock Dashboard

# Acknowledgements

# Abstract

The Stock Dashboard is a website where the user can search for specific stocks and view live data, such as their article sentiment, analyst target prices, future stock predictions, article topics and more. The website's goal is to help the user make future investing decisions. The data analysis is performed in the backend Python pipeline and stored on the MongoDB cloud database.

The article data is web scraped from various financial news publishers using Requests and Yahoo Finance API. The data is then cleaned using Beautiful Soup and Pandas. After performing more vigorous data pre-processing like lemmatization, topics are extracted from the text using Gensim and sentiment is analyzed using a lexicon-based approach with the VADER sentiment analyzer. Stock price predictions are made using historical prices and features extracted from historical prices like Moving Averages. The data is passed through a Long short-term memory (LSTM) neural network, after which predictions are made for the next ten days. The Python code is tested using unit testing. All the data is stored on MongoDB Atlas and is updated daily.

The website consists of a search page, dashboard and stock screener. On the search page, the user can search for the desired stock; the dashboard contains all the data visualizations made using ChartJS. A NodeJS server is used in conjunction with ExpressJS to access the data. In the stock screener, the user can filter by various metrics, like price or market cap, to search for stocks, which can help make future investing decisions

Contents

[Acknowledgements 1](#_Toc99976651)

[Abstract 1](#_Toc99976652)

[Literature Review 2](#_Toc99976653)

[Motivation 3](#_Toc99976654)

[Passion for investing and data analysis 3](#_Toc99976655)

[Alternative to expensive software 4](#_Toc99976656)

[The dashboard form as a great fit for the average investor 4](#_Toc99976657)

[Developing the right skills 4](#_Toc99976658)

[Inspiration from existing websites 5](#_Toc99976659)

[Personal use 5](#_Toc99976660)

[Aims and Objectives 5](#_Toc99976661)

[Summer Preparation and challenge week 5](#_Toc99976662)

[Minimum Viable Product 6](#_Toc99976663)

[Final Product 6](#_Toc99976664)

[Challenges 7](#_Toc99976665)

[Technical Documentation 7](#_Toc99976666)

[User Interaction and Design 7](#_Toc99976667)

[Dashboard 7](#_Toc99976668)

[Stock Screener 8](#_Toc99976669)

[Source Code Summary 8](#_Toc99976670)

[Web Scraping 8](#_Toc99976671)

[Data Cleaning 8](#_Toc99976672)

[Sentiment Analysis 8](#_Toc99976673)

[Topic Extraction 9](#_Toc99976674)

[Predictive Modelling 9](#_Toc99976675)

[Database 9](#_Toc99976676)

[Quality Assurance 9](#_Toc99976677)

[Test strategy 9](#_Toc99976678)

[User Testing 9](#_Toc99976679)

[Unit Testing 9](#_Toc99976680)

[Project Planning 9](#_Toc99976681)

[Jira 10](#_Toc99976682)

[Gitlab 10](#_Toc99976683)

[Gantt Roadmap 10](#_Toc99976684)

[Conclusion 10](#_Toc99976685)

[Summary 10](#_Toc99976686)

[Future Work 10](#_Toc99976687)

[References 10](#_Toc99976688)

# Literature Review

**Bonta, V. a. J. N., 2019. A comprehensive study on lexicon based approaches for sentiment analysis.. Asian Journal of Computer Science and Technology, S2(8), pp. 1-6.**

(Bonta, 2019) compares the sentiment analysis accuracy of the 3 most popular sentiment analysis tools in Python – Natural language processing toolkit (NLTK), Text Blob and VADER. The text analysed are movie reviews from rottentomatos.com[[1]](#footnote-1). Bonta, V. and Janardhan, N.K.N. came to the conclusion that VADER is the “gold standard” list of lexical features which are specially attuned to find semantics in micro blog text. The accuracy of VADER was 77.0%, compared 74% and 62% for Textblob and NLTK respectively. Accuracy is the number of correctly predicted data points out of all the data points. Also, VADER follows grammatical and syntactical conventions for expressing and emphasizing sentiment intensity, for example “that movie was good” compared to “that movie was GOOD” or “that movie was GOOD!!!”.

**Robert P. Schumaker, Y. Z. C.-N. H. H. C., 2012. Evaluating sentiment in financial news articles. *Decision Support Systems,* 53(3), pp. 458-464.**

(Robert P. Schumaker, 2012) investigated whether sentiment and tone of finical news articles correlate to measurable stock price movements. (Robert P. Schumaker, 2012) found out that subjective news articles were easier to predict in price direction (59.0% versus 50.0% of chance alone) and using a simple trading engine, subjective articles garnered a 3.30% return. Investigate the correlation further they found that our system was able to predict price decreases in articles of a positive sentiment 53.5% of the time, and price increases in articles of a negative sentiment 52.4% of the time. This can be an indicator that traders behave in a contrarian manner, e.g., see good news, sell; see bad news, buy.

**A. Porshnev, I. R. a. A. S., 2013. Machine Learning in Prediction of Stock Market Indicators Based on Historical Data and Data from Twitter Sentiment Analysis. 2013 IEEE 13th International Conference on Data Mining Workshops, Volume 13, pp. 440-444.**

(A. Porshnev, 2013) discussed and tested the possibility of using Twitter users’ moods and psychological states of people to improve stock returns. For analysis of psychological states they used a lexicon-based approach, and analysed 755 million tweets. (A. Porshnev, 2013) concluded that the addition of information did not allow for a significant increase in accuracy. The best average accuracy rate 64.10% was achieved by using a Support Vector Machine algorithm to predict DJIA indicator, which was only slightly better than a Neural Network.

From (Robert P. Schumaker, 2012) we can conclude that negative and positive sentiment can be used as a tool for increasing returns in the stock market. However, there is a gap in the research In terms of the use of the VADER sentiment analyser, which outperforms other lexicon based approaches (Bonta, 2019), in analysing finical news articles. (A. Porshnev, 2013) stated that Twitter sentiment does not increase stock returns, but it is unknown whether the use of VADER in am analysing finical news articles improves stock returns. As well as whether A Neural Network in conjunction with a lexicon based approach of financial news article sentiment analysis can provide positive returns in the stock market.

# Motivation

## Passion for investing and data analysis

I decided to make a stock dashboard because I am passionate about investing and love to analyse and visualize data, especially financial data. I have been following the stock market movements for over 4 years now and in the last few years, I have gained enough income where I can start investing myself. I think fundamental stock analysis using historical information is a great way to find inconsistencies in the market. ADD REFERENCE

During my first year studying here, I worked on a team project where we had to predict the house prices in Boston. This project was the start of my journey into the world of data analysis and predictive modelling. Since then I have worked on various projects that require data analysis and data visualization and have always found them enjoyable.

So for my Capstone project, I decided to combine both, my passion for investing and data analysis into one.

## Alternative to expensive software

A lot of stock analysis software is overly expensive or require a costly monthly subscription, for example, the Bloomberg Terminal can cost 2000$ per month or the FactSet terminal, which costs 12 000$ per year. For the average investor like me and many others, this is way too expansive. But isn’t it unfair that the rich investors can use tools that we can’t? Creating my own stock analysis tool is a great way to save some money and improve my financial knowledge along the way. Of course, my program would never be as advanced as the Bloomberg terminal, but after investing regularly for over 2 years I have understood that having too much data can also be a burden. A dashboard is a great tool for this since it doesn’t contain too much information, but it can still provide plenty to make an investing decision.

## The dashboard form as a great fit for the average investor

The definition for a dashboard is a graphical summary of various pieces of important information, typically used to give an overview of a business.

I chose to have the page in a dashboard form, that is, having a lot of information on a single page that does not require any scrolling, because, in my opinion, it is a useful way of conveying information, to both new and experienced investors. The average investor cannot spend hours analyzing stocks, so by using a dashboard the investor can quickly see the most valuable information without the need to browse through multiple pages and websites.

## Developing the right skills

My goal after graduations is to work as a Data analyst or pyhton Developer. A project that requires bot hdata anlayis and python programming is a great fit for me. Data visuliztion is a big part of beeing a data analyst, however if I just used Python for the data visuilztion, this owuld require the user to have python installed, and in most cases the vizulaitions could not be interacted with. Learning another vizuilation tool like ChartJS can be useful in creating beatiful vislzitions that anyone with na internet connaction can see, as well as allowed me to further devlop my frontend development skills.

Natural Language Processing is also a latge part ofg my project, in terms of Sentiment Analysis and Topic Extraction

Instead of suing a RDBMS database like MySQL. I though it would eb a good idea to learn NoSQL, since i nthe fincial industry a lot of comanies use NoSQL instead of SQL and it is great for storing fincaial data like stock price, which in MySQL woudl requirre a huge collectio nthat conatisn tens of thousands of rows.

## Inspiration from existing websites

Throughout the last two years I have used the online website Simply Wall Street to help me make investing decisions. Simply wall street is a stock and sectero analysis tool with the goal of making stock analysis simpler. It condenses large amoubnts of data from varius stocks to give the user an oveeview

## Personal use

# Aims and Objectives

The main aim of the website was to help the user make future investing decisions, by providing accurate and useful information related to the stock, as well as allowing the user to compare different stocks to each other. From this I conclude that the website must be easy to use, useful for both new investors and experienced investors, as well as contain information that is important in deciding whether a stock is a good buy or not.

The backend Python code on the other hand must be detailed and documented, reliable and efficient as well as scalable. Detailed and documented because this can help give context to future readers and allowing for the code to be reused in future projects. (Meza, 2018). Well documented code will allow me not lose momentum when coding or taking a break to work on projects for other modules.

The code should be reliable and efficient as I was planning to add hundreds of stocks for the user to view and automate such that the codes runs every morning to update the most relevant information. Since most of the code would run for very stock any minor speed and effecincly improvemtns would be amplified by the amount of stocks to be updated.

## Summer Preparation and challenge week

During the Summer preparation and challenge week my main objectives were to do research on what tools I could use for each section of the website and the code, as well as create the layout of the

During the summer I researched research papers, i nthe field of datan anlsyis, sentiment analysis and price prediction. Duringf this time I also did research into multimple pYhton libaries thta I coudl sue to perfomer the tasks, like NLTK and TenserFlow. Luickily I already had some experince with NumPy and Pandas, whci I used throughj the project for data cleaning and data analysis.

For webscraping I did reasraechi not multimple financial news websites and I the end decided on to use FINVIZ, as it contains imfdoation about hoasuands of stocks, contains articles relevant to the stock (even though later the articles did require further filtering) and contains many rleavnt metrics for specif stocks, like Price to Earnings[[2]](#footnote-2) ratios and Dividends[[3]](#footnote-3).

In terms of the frontend I was quite unsure what tools I would use to excpet that I woudllie kto use Chart JS, becasue the graphs looekd very good and could be interacted with. Since I winted to keep the forntend quite simple because the backend was my priority. I decided to use basic HTML, CSS and JavaScript, which would allow me to spend more time on the bakend - improveong the code and making better predictions. Howver with these tools I oculd not accesss the databse from the wesbite. For thgis I needed a JavaSciuprt abkcend server, whoch I made using NodeJS. THwe ncie thing about this is that this servber allows me to create an API, where other users can use my data, to permod their own vizuilations or predictions.

## Minimum Viable Product

The aims for the Minimal Viable product were as follows:

- [x] Create the layout of the Dashboard with sample data - i knew that I wouldntr be able to get all of the backend functionality done by the MVP, but having atleast a layout of the daashbord let me know what data I should be focuising on i n the backend.

- [x] Setup the MongoDb Database - I decided to sue a NoSQL database, since storing

- [x] Create the pipeline for newly added stocks

- [x] Web scrape infoamtion about the stocka such as articles and funetmantal infomation

- [x] Clean the articles - remove punction, lowercase

- [x] Analyze the sentiment of the articles using a alexicon approuch

- [x] Create the landing page / search page

- [x] Basic Neural Netwokr to predict future stock price

- [ ] Stock Screener

## Final Product

In addition to the features in the MVP, the objective for the final Product was to:

- [x] Only analyze the setimnint of imporatnat artilces, that is artrilces taht mention the stock multimple times

- [x] Extract topics from the articles

- [x] Create the pipeline to update stock already in the database

- [x] Stock Screener

- [x]

<p align="right">(<a href="#top">back to top</a>)</p>

## Challenges

During the early staged of the project, the biggest challenge was learning all the required new tools for making the website, such as ChartJS, which sadly does not have a very comprehensive documentation or NodeJS and ExpressJS, which I had not used before the project. To overcome this problem, I took online courses on LinkedIn learning and watched YouTube video tutorials. However, this was time consuming and resulted in me spending less time on making the project in itself.

As for the final product I decided to use a lexicon based approach using VADER, it is very import that the “bag of words” that the analyser uses is as fitted for article analysis as it can be. For this I tried using a custom lexicon of negative, neutral and positive words, which resulted in worse results than using the pre-defined lexicon for VADER, which is fitted to be sued for social media text analysis.

**Few, S., 2006. Information Dashboard Design: The Effective Visual Communication Of. 2nd ed. Sebastopol, CA: O’Reilly.**

**Information Dashboard Design: The Effective Visual Communication is a book**

# Technical Documentation

## User Interaction and Design

The user interacts with the website through the 3 pages: search page. The dashboard and the stock screener.

The design of the website is simple and uses a dark colorway with more varied colors used in the charts. It was important for the information in the website

### Dashboard

Visual display of the most information needed to achieve one or more objectives which fits entirely on a single computer screen so it can be monitored at a glance (Few, 2006)

**The information in a dahbaord is presented visually usually as a combination of of next and graphics.** Dahsbaords are hihly graphical not because it is cute but because graphical represention can often communicate with greater efficiancly and rickher meaning that text alone. (Few, 2006). With this in mind, I gave myself the task of having as little of text as possible and trying to convey information almost exclulsvely through hraphical fomrs instead of text. Even though Python in itself does support dashboard grapgival design and visuliztion, ChartJS in my opnion is an even better tool as it allows for detailed cosimztion of the graphs, quick krendietimg pseeds for anyone with a modern browser and would allow me to create beautiful and eye catching visualizations, which most investing websites like FINVIZ or MarketWatch do not have.

**Dashboards display the information needed to achive a specifin objectives.** To create a useful dashboard it was important to first figure out the objective, which in my case was to inform the user about a stocks health and investing potential. However now it wascrucial to understand what information would allow the user to improve their understand of the stock and decide their next step. The fifncail charectercis of a successful company are stable earning or as stated by (Payne, 2011) the messurment of Value Line Earnings Predictabilit , which is the reliability of earnings forecasts. Return on Equity (ROE). As well as comarping the sepcif company to the broeader market or industry averages. As !!! dicsovred setiment can be used to make market returns, thus it would be useful to include historical sentiment of articles.

### Stock Screener

## Source Code Summary

### Web Scraping

The amin source for web scraping information is FINVIZ.

### Data Cleaning

### Sentiment Analysis

Sentiment analysis is the process of computationally identifying and categorizing the opinions expressed in a piece of text, especially in order to determine the writer’s attitude towards a particular topic, product, etc. is positive, negative or neutral. (Bonta, 2019)

Unlike machine learning algorithms VADER performs better across various kinds of domains. As compared to machine learning techniques, VADER has several advantages. Firstly, it is both quick and computationally economic, which in the case of analysing thousand od articles is highly important. VADER runs directly from standard modern laptop or computer and does not require powerful hardware; a corpus takes a fraction of a second to analyse with VADER, but it approximately takes hours when using more complex models like Support Vector Machine. Second advantage is that the lexicon and the rules used by the VADER are directly accessible and not hidden. Therefore, VADER is easily understood, extended and modified. (Bonta, 2019)

Thus I decided that VADER would be the best fit for my projects, as it is fast, could run on my personal computer and does not require a test and training set, which in the early stages of my project I could not provide.

After performing web scraping and data cleaning, the text is split into sentences, because VADER works better on shorter texts [REFERENCE] and for most publishers the last 3 sentences are removed. The last 3 sentences are removed because often at the end of the article there would be either information about the author or disclosure information, like “This article represents the opinion of the writer, who may disagree with the “official” recommendation position of a Motley Fool premium advisory service.” or information about other articles or the publisher.

After this the title and each sentence is given a positive, neutral and negative score, which together make the compound score from -1 for very negative to 1 for very positive. Each sentence is given a wight based on its length and the tile is given the highest weight of 35% of the final rating, together these create the article compound score which is used is the dashboard visualizations and latest article section.

### Topic Extraction

### Predictive Modelling

Machine learning algorithms are used in the stock market forecasting for a long time [1], [2]. The most common methods are Neural Networks and Support Vector Machine [1], [3]. Usually machine learning algorithms trained on technical data about stock movements, for example moving averages. (A. Porshnev, 2013)

### Database

## Quality Assurance

### Test strategy

### User Testing

### Unit Testing

# Project Planning

## Jira

## Gitlab

## Gantt Roadmap

# Conclusion

## Summary

## Future Work

# References

A. Porshnev, I. R. a. A. S., 2013. Machine Learning in Prediction of Stock Market Indicators Based on Historical Data and Data from Twitter Sentiment Analysis. *2013 IEEE 13th International Conference on Data Mining Workshops,* Volume 13, pp. 440-444.

Bonta, V. a. J. N., 2019. A comprehensive study on lexicon based approaches for sentiment analysis.. *Asian Journal of Computer Science and Technology,* S2(8), pp. 1-6.

Few, S., 2006. *Information Dashboard Design: The Effective Visual Communication Of.* 2nd ed. Sebastopol, CA: O'reilly.

Meza, F., 2018. *Olioapps.* [Online]   
Available at: https://www.olioapps.com/blog/the-value-of-code-documentation/  
[Accessed 04 04 2022].

Payne, B., 2011. On The Financial Characteristics Of Firms That Have Achieved The Highest Levels Of Accuracy In Earnings Predictability. *Journal of Business & Economics Research,* 5(9).

Robert P. Schumaker, Y. Z. C.-N. H. H. C., 2012. Evaluating sentiment in financial news articles. *Decision Support Systems,* 53(3), pp. 458-464.

1. <https://www.rottentomatoes.com/> is an American review-aggregation website for film and television. Each film or show is given a score of 0% to 100% by critics and users. [↑](#footnote-ref-1)
2. [↑](#footnote-ref-2)
3. [↑](#footnote-ref-3)