Stock Market Portfolio Management with Time Series and Sentiment Analysis

Sachin Parsa

Master of Science in Computer Science
University of Illinois at Chicago

Committee Members: Dr. Natalie Parde, Dr. Cornelia Caragea
Supervised By: Dr. Natalie Parde
UIN: 662046242
Fall 2022

Contents

1 Introduction **3**

2 Literature Review **4**
2.1 FinBERT: Financial Sentiment Analysis with Pre-trained Language Models: 4

3 Methodology **4**
3.1 Data Collection 5
3.2 Clustering 5
3.3 Time Series Model 6
3.4 Sentiment Analysis 6
3.5 Portfolio Optimization 7

4 Evaluation and Results **7**
4.1 Clustering 7
4.2 Time Series 9
4.3 Sentiment Analysis 10
4.4 Portfolio Optimization 11

5 Conclusion **12**

6 Future Work **13**

Stock Market Portfolio Management with Time Series and Sentiment Analysis

November 28, 2022

Abstract

Stock market prediction is one of the oldest and most challenging tasks, leading to the creation of the $100 billion Fintech industry. There are so many microeconomics and macroeconomic concepts that determine the movement of stocks. I hypothesize that taking the historic performance of stock, public sentiment, and risk calculations are some of the most influential reasons for stock movements. In this project, the objective is to make an optimal stock portfolio using machine learning concepts like time series analysis, sentiment analysis, and clustering to filter the best stocks. With this list of stocks, I use a portfolio optimization technique, which minimizes the Sharpe ratio in order to reduce the risk and maximize return on investment. This strategy is tested by creating a portfolio for October of 2022 and comparing the performance of the portfolio with the performance of the S & P 500 index.

1 Introduction

Stocks are known to have characteristics based on their historic performance. If a stock has gone up 100% in one day, it has the potential to do the same in the future, given similar market conditions. This was the primary motivation behind using time series to analyze historic performance of a stock and provide predictions for the future. The annual returns of the stock also show a strong indication of the historic performance. Analyzing annual returns and comparing it to other stocks in the market is another indicator for future performance.

Stocks are also highly influenced by emotions, which has become more prevalent in recent years after the pandemic in 2020. Fundamental analysis does not work when evaluating stocks. The market is driven by sentiment and potential rather than assets and earnings; therefore, the other aspect of the project involves understanding the sentiment of the stock by analyzing news headlines related to the stocks.

Risk is another important metric used for making investment decisions. Fintech firms spend billions of dollars to minimize risks in their portfolios. Using various machine learning methods, this project takes factors such as historic performance, risk, and stock sentiment into account to tackle different areas. Analyzing historic performance is done using time series analysis. Clustering is used to analyze annual returns and compare performance of different stocks. Sentiment analysis is used to analyze the emotion of the stock and portfolio optimization is done to minimize risk. The goal of the project is to provide a portfolio of the

This project is made public on : Project Github Page

best performing stocks and the allocation of funds within the portfolio to maximize capital gains. The portfolio's cumulative gains are than compared to the gains of S & P 500 to analyze the portfolio performance.

2 Literature Review

2.1 FinBERT: Financial Sentiment Analysis with Pre-trained Language Models:

Sentiment analysis is a complicated task which extracts sentiment from given text. The Bidirectional Encoder Representations from Transformers (BERT) model was a huge breakthrough producing great results for sentiment analysis. The BERT model, however, was a general sentiment model, which does not consider complex financial language. Therefore, the sentiment analysis model used in this project was the Financial BERT (FinBERT) model. This model is similar to the BERT model, but trained specifically for financial applications in order to account for domain specific language. The paper shows experiments comparing the BERT model and the FinBERT model performance. The author hypothesizes that the FinBERT model will perform better than any other models given financial data. [1] The FinBERT model reduces the number of layers used compared to the BERT model. It can retain 97% of the performance with 40% less parameters. The accuracy of the model is also higher than 80% which is impressive for sentiment analysis tasks. [2]

There are a few drawbacks to the FinBERT model. One of the biggest issues is that the model does not perform well when it encounters numbers. In the absence of directional text, it usually characterizes given text as neutral. The FinBERT architecture is very similar to the BERT model with a few modifications. It uses slanting triangular learning rates, freezing the classification layer. The layers are gradually unfrozen such that the lowest level in the architecture is the least fine-tuned. Through the experimentation of the paper, it is proven that the FinBERT model is faster, cheaper, and more accurate than the BERT model.

3 Methodology

Different data sets were created for time series and sentiment analysis. Using Application Programming Interfaces (API) and various python libraries, the historic performance of the stocks and the headlines associated with the stocks were collected. The time series model was then trained for four years' worth of stock market data from September of 2018 to September of 2022 using the 29 stocks in the Dow Jones Industrial Average index (DJIA). The trained time series model was then tested for October 2022 to check for the accuracy of the predictions. Clustering techniques were used to remove low-performing stocks from consideration in the portfolio.
The headlines for the stocks were collected from Yahoo Finance and other news outlets using API calls. The sentiment analysis model used was FinBERT which is a pre-trained model and was tested on the headlines data set. The final step was to use the best performing stocks filtered by the previous steps and optimize the portfolio. A function is then optimized to find the appropriate allocation of funds in the portfolio. The portfolio performance is then compared to the performance of the S & P 500 index.

3.1 Data Collection

The first data set was collected using pandas data reader and Yahoo Finance. The data consists of all the stock data for the 29 stocks in the DJIA from Sept 30th of 2018 to Sept 30th, 2022 (excluding all non-trading days). Figure 1 shows a snapshot of the data set.

	Date	Open	High	Low	Close	Adj Close	Volume	ticker
0	2018-10-01	212.399994	213.399994	211.309998	212.190002	182.765472	1828800	MMM
1	2018-10-02	212.380005	215.839996	212.100006	215.710007	185.797348	1749400	MMM
2	2018-10-03	216.000000	217.339996	214.940002	215.759995	185.840408	2139200	MMM
3	2018-10-04	214.850006	215.660004	212.039993	213.839996	184.186661	1682500	MMM
4	2018-10-05	214.350006	215.029999	211.059998	213.190002	183.626801	2140300	MMM
...	...	...	...	...	...	...	...	...
29198	2022-09-23	100.620003	101.180000	98.019997	99.500000	99.500000	11976900	DIS
29199	2022-09-26	98.949997	100.660004	98.059998	98.120003	98.120003	9760500	DIS
29200	2022-09-27	99.529999	99.639999	95.430000	95.849998	95.849998	13360200	DIS
29201	2022-09-28	95.790001	99.870003	95.449997	99.400002	99.400002	12895500	DIS
29202	2022-09-29	98.529999	98.599998	96.230003	97.449997	97.449997	9435100	DIS

29203 rows × 8 columns

Figure 1: Data set containing historic data

The data set consists of 29203 rows since it has 4 years worth of trading information for 29 stocks and 8 columns. The columns are date, open (the price at which the stock opens on the given date), high (the highest price of the stock in a given trading day), low (the lowest price of the stock in a given trading day), close (the price at which the stock closes on the given day), volume (the number of stocks traded on the given trading day), ticker (the name of the stock being traded), and adjusted close (amended stock price to account for any corporate actions like stock splits, dividends etc.). The information used in this project was the ticker name, date, open price, and the adjusted close price. The adjusted close was used in the function optimization part of the project. The data set in figure 1 was used for time series analysis, to train the model as well as the clustering task to analyze annual returns. The same method of using pandas data reader and Yahoo Finance was used to get the test data for October of 2022 and the data for S & P 500.

The headlines used for this project came from web-scraping Yahoo Finance. Python's library, "Beautiful Soup", was used to grab response text from the Yahoo Finance's web page for each provided ticker symbol. The Beautiful Soup library's in-built functions, find and find all, found all the headlines and dates for each ticker. This allowed for filtering so that only October headlines could be extracted. Figure 2 shows the resultant data set.

	Ticker	Date	Headline	Sentiment
0	CAT	10/09/2022	Is There An Opportunity With Caterpillar Inc.'...	1
1	CAT	10/10/2022	Will Caterpillar (CAT) Beat Estimates Again in...	1
2	CAT	10/11/2022	Caterpillar (CAT) Gains As Market Dips: What Y...	1
3	CAT	10/12/2022	2 Cathie Wood Investments That Could Deliver S...	1
4	CAT	10/12/2022	Caterpillar Inc. Maintains Dividend	1
...	...	...	...	...
156	AMGN	10/24/2022	Can Amgen Finally Make an Upside Breakout?	1
157	AMGN	10/25/2022	Insiders at Amgen Inc. (NASDAQ:AMGN) sold US$4...	-1
158	AMGN	10/25/2022	Investors Heavily Search Amgen Inc. (AMGN): He...	1
159	AMGN	10/29/2022	Amgen (NASDAQ:AMGN) jumps 8.7% this week, thou...	1
160	AMGN	10/31/2022	Take the Zacks Approach to Beat the Market: Ca...	1

161 rows × 4 columns

Figure 2: Sentiment Analysis Dataset

3.2 Clustering

The data set was first pre-processed by finding the average annual returns for each stock over the four-year period. Preprocessing steps like removing null values and converting the data set from a long to a wide format were done. A new data frame, which only consisted of the stock ticker and the annual returns, was made. The data was then normalized, and the scaled data was then used to determine the best clustering model.

The models considered included: K-means, hierarchical single-linkage, hierarchical complete-linkage, and DBSCAN. The silhouette coefficient was used as an evaluation metric for each model to determine the best model. The parameters used for each model are as follows:

- K-Means: 3 clusters, 10 iterations and random clustering
- Single linkage: 3 clusters, Euclidean distance
- Complete linkage: 3 clusters, Euclidean distance
- DBSCAN: eps = 0.5, Euclidean distance

The clusters were plotted based on the annual returns, which provided the low, medium, and high performing stocks. Apple's annual returns were 46%, which was much higher than the rest of the other DJIA stocks. Therefore, Apple stock was considered as an outlier and removed for the consideration of the clustering model. Since, K-means clustering and complete linkage methods use Euclidean distance for analysis, it causes the models to be sensitive to outliers. After determining the best model by the silhouette coefficient, the low performing stocks are removed from the portfolio consideration.

3.3 Time Series Model

The model was trained on 4 years of data from Sept 30^{th}, 2018, to Sept 30^{th}, 2022. The SARIMA time series model was used for analysis where we set the p,d,q values to 1 in order to provide the best results. The p value is the number of auto-regressive terms. The d value is the difference order in the trend, and the q value is the moving average for the trend. Since the SARIMA model was used, there are three additional seasonality parameters to consider, P, D, Q. These are the seasonality auto-regressive order, seasonal difference order, and the seasonal moving average respectively. There is also an s value, which is the number of time steps for a single seasonal period. The value for s is set to 12, so the yearly seasonality, like the holiday effect, is considered as well. These values provided the best results for the time series model.

The correlation, the mean squared error, the percent change, and percent increase were all calculated for the different stocks and were compared to the actual stock performance. The percent error is calculated for the predicted price increase and the actual price increase to check for the accuracy of the time series model. The percent change was used to normalize the results, since some stocks could be worth much more than the others. The predicted and the actual values of all the considered stocks were plotted for comparison. The data was then sorted by the percent increase in the stocks. Finally, the top five stocks were filtered such that only those were considered for the portfolio.

3.4 Sentiment Analysis

The FinBERT model was used for sentiment analysis. The model is like the BERT model except it was trained on a financial corpus called TRC2-financial (subset of Reuters TRC2) and financial phrase bank. TRC2-financial was filtered using financial keywords, and it includes 46,143 documents with more than 29M words and nearly 400K sentences.[2] Financial phase bank consists of 4845 English sentences selected from financial news. [2] The data set consisted of the headlines of the top 5 stocks, which were filtered by the time series and clustering models. These headlines were manually annotated to check for accuracy. The movement of stocks is unaffected by neutral headlines, so these were disregarded in the model. The model was imported from hugging face and the pipeline was made with the help of the transformers library.

The model creation involved using a learning rate of 3e-5 and ran until there is no improvement in validation loss for 10 epochs. The pre-trained model was imported and it was run on the collected headlines to find the sentiment. The scores provided for each sentiment were then classified into positive if the score was more than 0.5 and negative if the score was less than or equal to 0.5. The predicted labels were then recorded next to the actual values for comparison. The precision, recall, F1 score, and accuracy were calculated to evaluate the model. If there were more positive than negative headlines then stock was removed from portfolio consideration. The positive and negative sentiment was also seen over the course of the month to compare the performance and sentiment of the portfolio. After this step, the final list of stocks was created for portfolio optimization.

3.5 Portfolio Optimization

The final portfolio list is provided after the clustering, time series and sentiment analysis models filter from the initial list of 29 stocks. The main criteria for optimization of the portfolio is Sharpe ratio. It is a metric, which adjusts returns while also considering risk. Minimizing risk in a portfolio and maximizing capital gains are the primary objectives of the project. Hence, The Sharpe ratio is used as the metric for portfolio optimization. It is calculated by subtracting the return of the portfolio value with the risk-free rate, which is divided by the standard deviation of the portfolio's excess return. In this experiment the return in portfolio is the average of the daily returns of the portfolio. The standard deviation of the portfolio's excess return is considered to be the standard deviation of the daily returns of the portfolio.

The Sharpe ratio is an annualized measurement, and it varies significantly if different intervals are used. Therefore, to adjust for daily returns Sharpe ratio needs to be normalized by multiplying the square root of 252. There are 252 trading days in a year so multiplying this value to the Sharpe ratio provides the normalized results. The starting value is taken to be 1 million dollars, but this can be customized according to personal finances.

Portfolio optimization used the minimize function from the scipy library. This function minimizes a scalar function. Since we want to maximize the Sharpe ratio, the minimize function is applied to minimize the negative of the Sharpe ratio. This finally provides what percentage of your money should be assigned to which stock, such that capital gains are maximized. If this portfolio works better than the performance of SPY, which is a common measure of success, it shows that the strategy of using these machine learning techniques for prediction is viable.

4 Evaluation and Results

4.1 Clustering

The annual returns were calculated for all the stocks in Dow Jones industrial average and three different clustering techniques were used to determine the best model for clustering. Figure 3 shows the results for K-means clustering with the outlier Apple. Apple is considered an outlier since it performed 16% better then the next best performing stock. The silhouette coefficient was 0.66 and the high performing cluster only consisted of Apple and Microsoft. Since, K-Means clustering is outlier sensitive, Apple was removed for the remaining clustering tasks.

The silhouette coefficient for K-Means clustering when Apple was removed was 0.68. The silhouette coefficient was the metric used to compare the different models. Figure 4 shows the resultant clusters with K-Means without Apple.

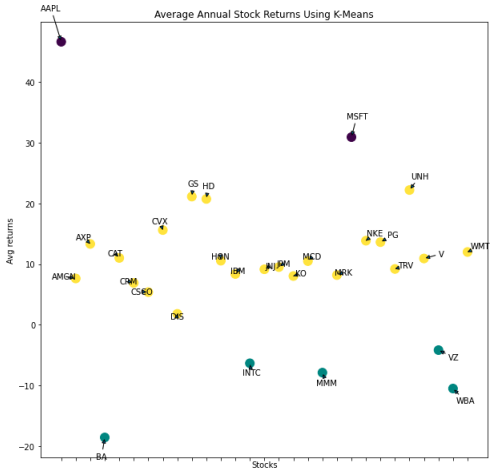

Figure 3: K-Means with Apple

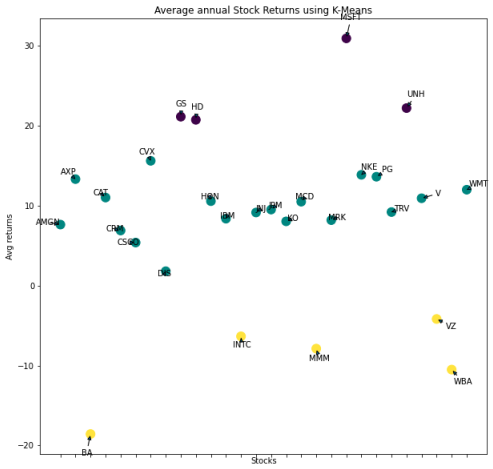

Figure 4: K-Means without Apple

The single linkage hierarchical clustering resulted in a much lower silhouette coefficient of 0.38, which is significantly lower compared to the K-Means clustering method. The resultant clusters of the single linkage hierarchical method are shown in figure 5.

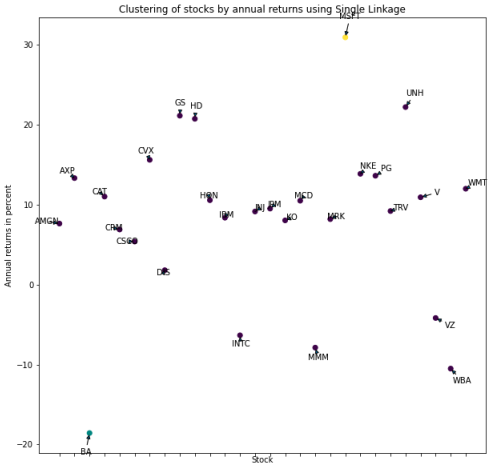

Figure 5: Clustering with single linkage

Everything was labeled into one cluster except for Boeing and Microsoft which clearly shows a poor performing model. The complete linkage method however produced the same results as the K-Means clustering method with a silhouette coefficient of 0.68 and the clusters had the same results as K-Means as shown in figure 4. DBSCAN was the last clustering techniques used and the silhouette coefficient was 0.49 which is worse than K-Means and complete linkage method. It only had 2 clusters and the clustering was not meaningful for analysis. The results are shown in figure 6.

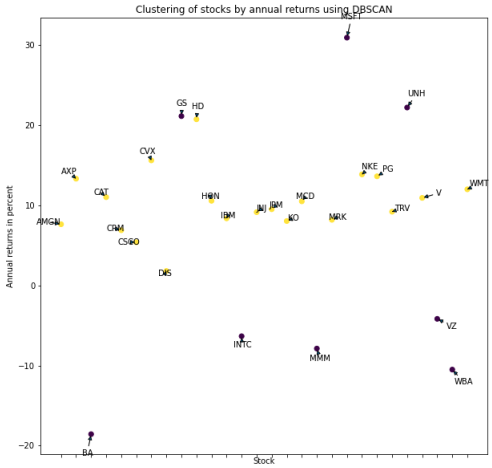

Figure 6: Clustering with DBSCAN

After analyzing the different clustering techniques, the K-Means clustering,

and complete linkage model were the best performing models. The cluster thresholds after removing apple are shown in table 1.

Stock Performance for Returns	
Clusters	**Annual Returns**
Low	$< 1\%$
Medium	1% - 19%
High	$> 20\%$

Table 1: Resultant annual returns clusters

This provided the initial filtering of the stocks by removing the low performing stocks, which were INTC, MMM, VZ, WBA and BA. Table 2 summarizes the results of the clustering experiment.

Clustering Model Comparison	
Model	**Silhouette Coefficient**
K-Means	0.68
Single Linkage	0.38
Complete Linkage	0.68
DBSCAN	0.49

Table 2: Silhouette coefficients for different clustering methods

4.2 Time Series

The time series model was run to predict the month of October 2022 for the 29 different stocks and the results were stored in a dataframe. The dataframe contained the the predicted values, actual values, correlation, mean square error, and percent error.The predicted and the actual values were stored to compare the results and the percent error, correlation and mean square error were calculated for each stock. These stocks were later sorted based on their predicted percent increase to normalize the comparison between the stocks, and the top 5 performing stocks were filtered out. The top 5 stocks were AMGN, JPM, TRV, CVX and CAT. The results of the predicted and the actual values are summarized in table 3. While the percent error and correlation of the stocks is given in table 4.

Time series comparison		
Stock	**Predicted Increase**	**Actual Increase**
AMGN	16.45%	17.93%
JPM	16.63%	19.01%
TRV	17.08%	17.95%
CVX	25.27%	19.53%
CAT	27.39%	28.61%

Table 3: Time series predictions and actual data

Top 5 Performing Stocks		
Stock	**Correlation**	**Percent Error**
AMGN	84.60%	8.27%
JPM	91.90%	12.53%
TRV	94.90%	4.83%
CVX	94.22%	29.43%
CAT	93.66%	4.25%

Table 4: Time series model performance

The predictions and the actual values were plotted for comparison and the results are shown for the top 5 performing stocks in figures 7 - 11. The red line shows the predicted price, and the blue is the actual price. The x axis shows the days it is being predicted after the end of the training period which is Sept 30^{th}. The y axis shows the price of the stock.

```
Price Increase 38.03863507088769
Percent Increase 16.44811487094364
MSE: 53.276
Correlation: 0.846
AMGN
```

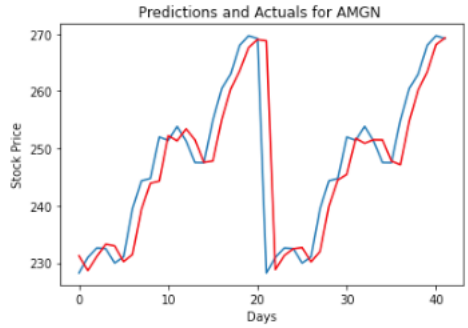

Figure 7: AMGN

```
Price Increase 45.53066913191711
Percent Increase 27.398443890049613
MSE: 26.179
Correlation: 0.937
CAT
```

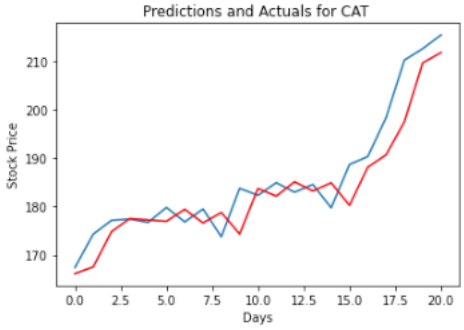

Figure 8: CAT

```
Price Increase 36.550185029527626
Percent Increase 25.273821021217348
MSE: 12.724
Correlation: 0.942
CVX
```

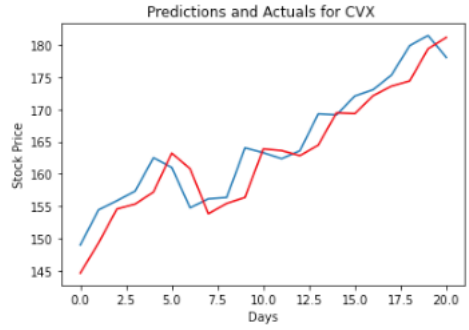

Figure 9: CVX

```
Price Increase 17.814413541817913
Percent Increase 16.62893670430148
MSE: 10.297
Correlation: 0.919
JPM
```

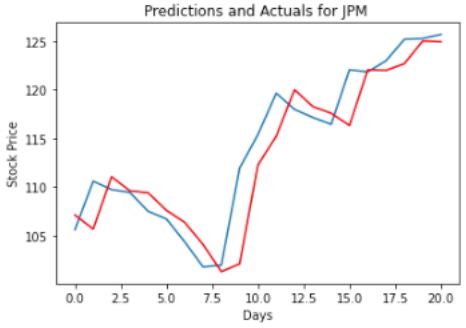

Figure 10: JPM

```
Price Increase 26.086614980436053
Percent Increase 17.080035467245093
MSE: 9.695
Correlation: 0.949
TRV
```

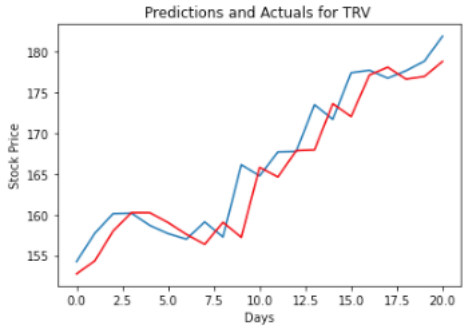

Figure 11: TRV

4.3 Sentiment Analysis

The sentiment analysis model was evaluated using precision, recall, accuracy and F1 score. Figure 12 shows the results of the sentiment analysis model.

```
Precision:  0.8291139240506329
Recall:  0.9849624060150376
Accuracy:  0.8198757763975155
F1 Score:  0.9003436426116839
```

Figure 12: Results of sentiment analysis model

These results are impressive for a sentiment analysis task. The F1 score of 90% and accuracy of 82% is very high compared to baseline accuracy of LSTM models, which is around 70% accuracy and

F1 scores is around 65%. The data set is not evenly balanced since, the news API collects all the data related to the stock. Figure 13 shows the number of headlines for each stock:

```
JPM     51
AMGN    36
CAT     31
CVX     31
TRV     12
Name: Ticker, dtype: int64
```

Figure 13: Number of headlines per stock

The final data set with the predicted values recorded is shown in figure 14:

	Ticker	Date	Headline	Sentiment	Predicted_sentiment
0	CAT	10/09/2022	Is There An Opportunity With Caterpillar Inc.'...	1	1
1	CAT	10/10/2022	Will Caterpillar (CAT) Beat Estimates Again in...	1	1
2	CAT	10/11/2022	Caterpillar (CAT) Gains As Market Dips: What Y...	1	1
3	CAT	10/12/2022	2 Cathie Wood Investments That Could Deliver S...	1	1
4	CAT	10/12/2022	Caterpillar Inc. Maintains Dividend	1	1
...	...	...	...	...	...
156	AMGN	10/24/2022	Can Amgen Finally Make an Upside Breakout?	1	1
157	AMGN	10/25/2022	Insiders at Amgen Inc. (NASDAQ:AMGN) sold US$4...	-1	1
158	AMGN	10/25/2022	Investors Heavily Search Amgen Inc. (AMGN): He...	1	1
159	AMGN	10/29/2022	Amgen (NASDAQ:AMGN) jumps 8.7% this week, thou...	1	1
160	AMGN	10/31/2022	Take the Zacks Approach to Beat the Market: Ca...	1	1

161 rows × 5 columns

Figure 14: Sentiment analysis data set with predicted values

The data set consists of 161 total headlines and the predicted and the actual sentiment are stored next to each other for comparison. The model predicts a positive or negative label with 1 being positive and -1 being negative. The FinBERT model predicts a score based on the headline's text and if it is above 0.5 it is considered positive otherwise considered negative. The total number of positive and negative headlines for each stock were then plotted next to each other to analyze if there were any stocks, which had more negative headlines than positive headlines. The graph in figure 15 shows the results.

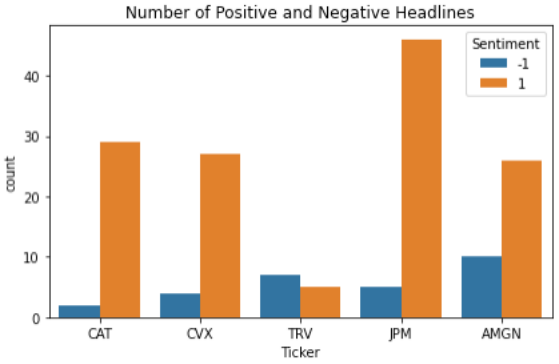

Figure 15: Number of positive and negative headlines per stock

As we can see from the graph, all the stocks except for TRV have more positive headlines than negative headlines. This indicates that TRV should be removed from consideration for the optimal portfolio.The number of positive and negative headlines throughout the month were recorded. This was to to analyze the overall portfolio sentiment throughout the timeline of the month. The results are shown in figure 16.

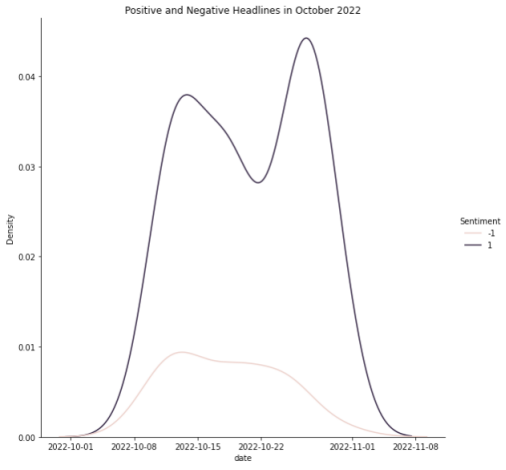

Figure 16: Positive and negative headline over the course of October 2022

4.4 Portfolio Optimization

The final portfolio considered for optimization were AMGN, JPM, CVX, and CAT. The experiment was run with and without TRV to analyze if it is considered

a high-risk stock. The results were then compared to the performance of SPY. The portfolio performance and the SPY performance is comapred for the month of October 2022 in figure 17.

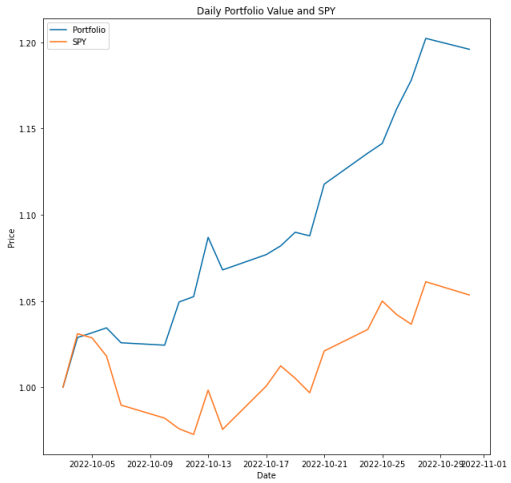

Figure 17: Comparison of SPY and portfolio performance with TRV

As shown in figure, the portfolio is performing far better than the S & P 500 index. The resulting allocation of the portfolio are summarized in table 5. 0% of the funds were allocated to TRV (the actual percent increase was the lowest among the top 5 stocks chosen for the portfolio) which confirms that the sentiment analysis was accurate for removing this stock from the portfolio. The total cumulative returns were 19.59% while the SPY only increased by 5.35%. The Sharpe ratio with TRV was 1.28 while it was 1.81 when it was removed. Since there was no allocation of funds to TRV the cummalative returns stayed the same for both portfolios.

As we can see TRV is set to 0 in this case, so this stock is high risk and hence avoided. The portfolio returns and the allocation are the same with and without TRV. This shows that the sentiment analysis model analyzing more negative headlines and eliminating it from the portfolio was accurate. The portfolio without TRV provides the same results just using less computation power.

Final Allocation of Stocks	
Stock	**Allocation**
AMGN	42.17%
JPM	1.09%
CVX	41.89%
CAT	14.84%

Table 5: Final allocations in portfolio

5 Conclusion

The experimentation done in this project does support the hypothesis that historic performance, sentiment, and risk are some of the most important factors to consider when making investments in the stock market. The final portfolio provides almost 14% more increase in capital gains than the S & P 500 index. There needs to be more data points to analyze, and the model has not seen extreme conditions like recessions to be able to properly predict the future month results.
There were many challenges during this project, one of the biggest challenges was getting the news headlines and manually annotating them. Yahoo Finance does not store more than 1 month worth of headlines. Since the APIs for collecting this data is written, the model will collect this data and this limitation will no longer be a concern. The dataset collected for different stock headlines was manually annotated which might have some errors. This can be improved if it was annotated by a professional in finance. The project can provide the optimal portfolio and the associated headlines as well as the allocation of funds. The initially objectives and analysis were achieved by the project. The final cumulative returns of almost 20% for a month is an impressive result.

6 Future Work

There are a lot of features which can be added to this project including customizing the risk a user is willing to take, the starting amount of money for the user as well. The initial stocks considered are just 29 from the Dow Jones industrial average, with more computation power we can do this for the S & P 500 which consists of 500 stocks. There are other factors which can be considered for a particular stock like the volume, float, support, resistance, market conditions and risk/reward ratio. There can be more experimentation to determine if adding these features would benefit the model. There can be more data like twitter and reddit data which can be considered for the sentiment analysis as well. These features may provide better results than the 20% increase in capital gains.

References

[1] K. T. Jacob Devlin. Bert: Pre-training of deep bidirectional transformers for language understanding.

[2] D.H.ZhuangLiu.Finbert:Apre-trainedfinanciallanguagerepresentation model for financial text mining.

[3] Chen, Peng & Niu, Aichen & Liu, Duanyang & Jiang, Wei & Ma, Bin. (2018). Time Series Forecasting of Temperatures using SARIMA: An Example from Nanjing. IOP Conference Series: Materials Science and Engineering.

[4] Dongkuan Xu Yingjie Tian. (2015). A Comprehensive Survey of Clustering Algorithms