NLP on 10-K Annual Report of Small Cap Stocks

Team5: Jiaqi Su, Max Stevens, Timothy Lin, Zefang Fu

April 2024

1 Introduction

The annual report (Form 10-K) filed by public companies in the United States with the Securities and Exchange Commission (SEC) provides a comprehensive overview of the company's business and financial condition over the past years. Using techniques of Natural Language Processing (NLP), we can see insights and sentiments that directly contribute to the return of a firm. This paper explores applications of various NLP techniques to the 10-K reports of SP600 small cap stocks, specifically focusing on sentiment analysis.

Our report mostly examines the information in items 1 (Business), 1A (Risk Factor), and 7 (Management's Discussion and Analysis of Financial Condition and Results of Operations) of the 10K Annual Report. The three factors are crucial to answering how people are reacting to the stock in that specific state.

We decided to examine the small-cap stock in the SP 600 index. The reason behind doing so is that there are more inefficiencies within small-cap stocks that we can take advantage of.

We apply text clustering techniques to group companies based on the similarity of their business descriptions, and we employ sentiment analysis to examine the market's reaction to the risk factors and management's discussion within these fillings.

The results we get from this study can greatly improve our understanding of the correlation between market sentiment and market performance. This research can enhance the future development of predictive models for stock performance based on textual analysis of corporate filings.

2 Data

10K Annual is chosen for its credibility, comprehensiveness, and huge sentiment factor. The US Securities and Exchange Commission (SEC) and independent auditors annually file and review the 10K Annual Report. The rules

of the SEC ensure that the information provided meets certain standards of transparency and accuracy. Each annual report contains 16 items, including a business overview, a risk factor, financial statements, and user sentiment items that provide data for sentiment analysis.

Our 10K annual report data set is taken from the following Hugging Face link: "https://hugging face.co/datasets/JanosAudran/financial-reports-sec." The data consists of the annual reports of US public firms filing with the SEC EDGAR system from 1993–2020. Each annual report (10K filings) is broken into 20 sections. We split each section into individual sentences. Sentiment labels are provided on a per-file basis based on the market reaction around the filing date for three different time windows: [t-1, t+1], [t-1, t+5], and [t-1, t+30].

Our list of companies for the SP 600 stocks is taken from Wikipedia on March 18^{th}.

2.1 Data Cleaning

At the start of the process, our program went through extensive data cleaning. By concatenating all the row information into one single string that differentiates the row from the others, we ensured that all of our data was unique and duplicate information had been removed.

2.2 Prepossessing

Each SEC stock has a unique Central Index Key (CIK). We stored all the CIKs of the SP 600 stocks into a map to put them into a map in the format of a string.

Since we are only taking items 1, 1A, and 7 of a SP600 stock 10K filling, we will have to check for the correct section. The section data is converted to a string for comparison purposes.

2.3 Feature Extraction

A new data frame was created, only containing the data that is part of items 1, 1A, and 7 of the SP600 index. After the data is processed, all the unnecessary information is dropped, leaving the columns: sentence (text information in the specific section), section, fillingDate, docID, sentenceID, ticker, state of incorporation, and returns (1d, 5d, and 30d).

2.4 Data Chanllenges and Solutions

Due to the large data size and limited runtime in a free Google Colab environment, Our program had trouble loading data and exporting data without kernel interruptions. As a result, we had to divide the dataset into 1000 chunks

and read and output each chunk individually so that we would not start from scratch when our program was interrupted.

The data is then combined all together and exported as a signal CSV (Comma Seperated Values) file.

3 Methods

3.1 Documents Clustering on business description

Section 1 of the 10-K annual report describes the business of the company, including history, strategy, services, etc. The information contained in this part of the annual report can thus be used to indicate the industry of the company, which is also the sector of the corresponding stock. So with the business description section data, we did text clustering on 452 stocks from the S&P600 list with available 10-K data.

In natural language processing, text clustering is the task of grouping documents or pieces of text with similar contents or topics. The classical routine for text clustering is to turn text into vectors as input for a classical clustering task in machine learning. The three methods we use for vectorization in this project are "bag-of-words,", word embedding, and document embedding.

3.1.1 Bag-of-words

Bag-of-words is one of the most basic approaches to getting vector representations of sentences or documents in natural language processing. It only considers what words are included and the frequency of words, regardless of the order or grammar.[5]

For example, if we consider two sentences

> s_1 = "we offer software and services help clients to optimize their business"
> s_2 = "our products include software and services that help customers manage their business easily and efficiently"

There are 19 unique words so we would have a bag of words looks like

> {"we":1, "offer":2, "software":3, "and":4, "services":5, "help":6, "clients":7, "to":8, "optimize":9, "their":10, "business":11, "our":12, "products":13, "include":14, "that":15, "customers":16, "manage":17, "easily":18, "efficiently":19}

And using this dictionary-like model we can convert the two given sentences into vectors below:

$$\vec{v_1}^T = [1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0]$$
$$\vec{v_2}^T = [0,0,1,1,1,1,0,0,0,1,1,1,1,1,1,1,1,1,1,1]$$

In our project, we employ BoW to compute a vector representation of each small-cap stock's 10-K business description by having each vector count the frequency of words appearing in the document.

3.1.2 Word embedding

In natural language processing, word embedding refers to a representation of a word in a vector of real numbers. The greatest difference between word embedding and the previous bag-of-words technique is that words are not treated as atomic units but as continuous vectors instead[3]. Thus, the similarities and differences between words can be represented and expressed by the similarities and differences in their vector representations. Furthermore, syntactic and semantic regularities among words can be captured. For example, $\mathbf{v}_{King}$ - $\mathbf{v}_{Man} + \mathbf{v}_{Woman}$ is very close to $\mathbf{v}_{Queen}$ in vector space. [2] To compute such vector representations, there have been many methods using unsupervised models to learn relationships between words based on a large corpus.

The model we used in this project is GloVe developed by Stanford. [4] The main idea of GloVe is to compute vector representations based on the ratio of co-occurrence probabilities, where the co-occurrence probability is defined as the probability of word i's occurrence in the context of word j. [4] The co-occurrence probability indicates the frequency of two words appearing in the same context and thus reflects their similarity. Furthermore, the ratio of co-occurrence probabilities specifies their differences in the context of some other words. For example, the co-occurrence probability of "ice" and "water" is not negligible, and computing the ratio of the co-occurrence probability of "ice" and "solid" over the co-occurrence probability of "water" and "solid" would help locate the similarity and difference between "water" and "ice".

In our project, we use the version "glove.6B" trained on Wikipedia 2014 and Gigaword 5 with 6 billion tokens and 400,000 vocabularies to get 100-dimensional vector representations of each word. After taking the average value over all words in the business description of one stock, we have a specific 100-dimensional vector representation for each stock's business description and can thus proceed to clustering stocks.

3.1.3 Document embedding

Unlike word embedding, which gives a unique representation for each word, document embedding creates representations for each complete piece of text or

document. Thus, document embedding requires taking the contexts of words into account instead of simply combining vector representations of words in the target document. Similar to word embedding, the methods to compute also rely on unsupervised training of models across a large number of corpus.

For example, doc2vec is trained to predict masked words based on document vectors and some contexts. By updating the document vector to get higher accuracy in training, the model would finally output a vector with shared information across all words in the target document.[1]

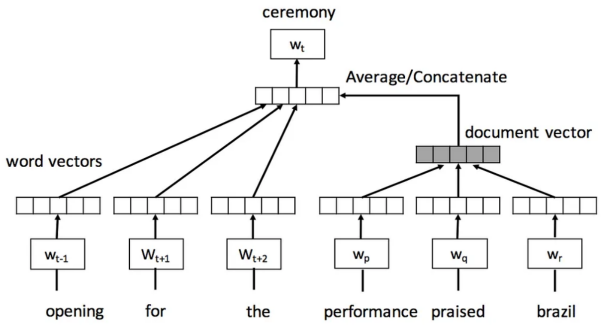

Figure 1: Doc2vec structure

There is another famous document embedding technique getting more popular these years called contextualized embedding based on encoder-decoder structure. Instead of having only a single static word embedding fixed for each word, each word's embedding is computed based on context, including information such as neighboring words and the relations between words at different positions. This can help solve the problem of the traditional word embedding method of having the same embedding for different meanings, like river "bank" and "bank" account. In this way, the embedding of a sentence or a paragraph can be formed by averaging or concatenating words' embedding with context information included.

The tool we used to get document embedding is the Cohere API, which also tries to include all context information and the topic of the paragraph in the document embedding.

3.1.4 K-Means clustering

With a representation of the business description for each stock, we can apply clustering algorithms to identify groups of stock with similar content by grouping vectors close to each other in our vector space. K-means is a classical clustering technique that can divide all data into k groups, where k is a predefined hyperparameter. The idea of this algorithm is to divide N data

points in M dimension space into K groups such that the in-group Euclidean distances, i.e. sum of squares, are minimized. The standard K-Means algorithm is:

1. Initialize k center points c_i for $i \in \{1, 2, ..., k\}$.
2. For $i \in \{1, 2, ..., k\}$ set each cluster C_i to include all data points such that the distance between the data point to any center c_j of cluster C_j for any $j \in \{1, 2, ..., k\}, j \neq i$ is greater than or equal to the distance between the data point center c_i of cluster C_i.
3. For $i \in \{1, 2, ..., k\}$ compute the new center c_i of cluster C_i to be the center of mass of all data points in the cluster $c_i = \frac{1}{|C_i|} \sum_{x_i \in C_i} x_i$.
4. Repeat step2 and step3 until no data point change its cluster assignment.

3.1.5 Label validation

To validate and compare the methods, we used manual labeling to extract the true sector label with the highest frequency in each cluster as the predicted label for all stocks in that cluster. Therefore, we can compare the predicted and true labels of stocks to see whether our clustering preserves the similarity between stocks by grouping similar business description text in 10-K annual reports.

3.2 Sentiment analysis on Management's Discussion and Risk Factors

In this study, we employed VADER SentimentIntensityAnalyzer and TextBlob to compute sentiment scores from business management description and risk factors of SP 600 companies. We processed multiple CSV files to consolidate relevant data, assigning sentiment scores to each text entry. Then, we averaged these scores by company and date. Concurrently, we extracted and cleaned 1-day, 5-day, and 30-day stock return data for these companies. Using these datasets, we calculated the correlation coefficients to assess the linear relationship between sentiment scores and stock returns. For this study, we focused on two areas from the 10K data: Management's Discussion and Risk Factors. As the names suggest, Management's Discussion section gives the company's own views on the results of business over that period. Similarly, with Risk Factors, this included information on the most significant risks to the company. From these sections, our goal was to find a correlation between their sentiment scores and the companies' stock returns over the periods of 1, 5, and 30 days from the release of the 10K data. Once our data was processed, we ran each sentence through the sentiment analyzers and built the average sentiment scores for each stock filing date with each sentiment tool and for both sections. With these average sentiments and the stock returns, we were able to run an analysis on how they were connected. This is a key piece to the way our sentiment analysis

was run, looking at the individual scores by sentence and taking the average rather than running the sentiment on the section as a whole.

3.2.1 Data Collection and Processing

We first processed the datasets in order to make it easier for sentiment score analysis. CSV files containing the management descriptions and stock return information were collected from a specified directory. The Python OS library was utilized to list and read multiple CSV files within a folder path. Each CSV file was filtered to remove irrelevant rows based on a predefined section number. The filtered data from all CSV files was then concatenated into a single DataFrame for analysis. We intially looked separately at the Management's Discussion and Risk Factor sections then merged those datafiles together based on the ticker symbols and filing dates. For the merged file we only looked at rows where there were both scores for Vader and TextBlob

3.2.2 Sentiment Score Analysis

The methods we used for sentiment score analysis are VADER (Valence Aware Dictionary and Sentiment Reasoner) and TextBlob. VADER has a built-in dictionary (lexicon) of words and phrases, each of which has been manually rated for sentiment valence by human annotators. Valence scores range from -4 (extremely negative) to +4 (extremely positive). When VADER analyzes text, it looks at the words and phrases in the lexicon and calculates the intensity of sentiment present in the text. This is done by summing the valence scores of each word and normalizing to a range between -1 and +1. This score is called the compound score, which VADER uses to represent the overall sentiment of a piece of text, and the compound sentiment scores were calculated for each management description in the DataFrame. The data frame was grouped by company tickers and filing dates, and the mean sentiment score for each group was computed.

TextBlob is a popular Python library for processing textual data. It provides a simple API for diving into common natural language processing (NLP) tasks such as part-of-speech tagging, noun phrase extraction, sentiment analysis, classification, translation, and more. TextBlob's sentiment analysis relies on a pre-trained model that uses a bag-of-words technique. It doesn't take the order of words into account but instead looks at the presence and frequency of words that have been pre-labeled as carrying positive, negative, or neutral sentiment in a training set. The model aggregates the scores of individual words, which have been assigned polarity and subjectivity scores based on human-labeled training data, and computes the average to determine the text's overall sentiment. It has a float within the range [-1.0, 1.0], where -1.0 signifies a negative sentiment, 0 signifies a neutral sentiment, and 1.0 signifies a positive sentiment. This score represents the emotional reading of the text.

3.2.3 Correlation Analysis

We analyzed the correlation by using the correlation coefficient and plots. The correlation coefficient was interpreted to evaluate the strength and direction of the linear relationship between sentiment scores and stock returns. We used three methods for correlation calculation: Pearson, Spearman Rank, and Kendall Tau. Pearson coefficients range from -1 to 1, with positive values representing a positive relationship and negative values representing a negative relationship. It assumes that the variables follow a normal distribution and evaluates the strength and direction of their linear association. Kendall Tau is similar to Pearson, but it's based on the ranks of the data rather than the actual values. It measures the similarity of the orderings of the data when ranked by each of the variables. It is robust to outliers and does not assume a linear relationship. Lastly, the Spearman rank, which is similar to the Kendall-Tau correlation, is based on the difference between the ranks of the data. It measures the monotonicity of the relationship between the variables. Like Kendall Tau, it's robust to outliers and does not assume a linear relationship. With three correlation methods and two sentiment methods, we had a strong foundation to analyze our data.

3.2.4 Wordcloud Analysis

After completing our analysis of sentiment scores and returns, we turned to finding other ways to represent and learn from this data. One of these ways was through word clouds, a visual descriptive tool that ingests the sentence data we had and produces an image of the words most included in the data. These words have different sizes based on how often they occur and provide an interesting visual of what is inside these sections of a 10-K. These word clouds were constructed by taking all of our sentences in each section (Management's Discussion and Risk Factors) and concatenating them into the full string of words. From there, it was run through the word cloud package on Python, in which we included the removal of the natural language tool kits for stop words like "at," "the," etc.

4 Results

4.1 Documents Clustering on business description

We use a confusion matrix to compare accuracy and error in our predicted labels. As we can see from the figures, clustering using bag-of-words has few one-to-one labels between the clusters and true sectors, while word embedding using GloVe makes clustering more similar to true sectors, and document embedding has the most one-to-one mapping between clusters and sectors.

If we look into the details for each sector, we can see that in all three methods, stocks in "health care" and "financials" sectors are most easily grouped

together, and stocks in "customer staples" and "customer discretionary" are most often confused. This makes sense since health care and financial companies contain products and services that are different from other sectors, while customer staples and discretionary companies would describe their businesses in similar ways. In bag-of-words and word embedding results, "industrials" are confused with "information technology," while "industrials" are confused with "materials" in document embedding. This is because the descriptions of industrial and information technology companies both use some technology-related words, but document embedding analyzes the similarity of documents not depending on word occurrence but contextualized meaning.

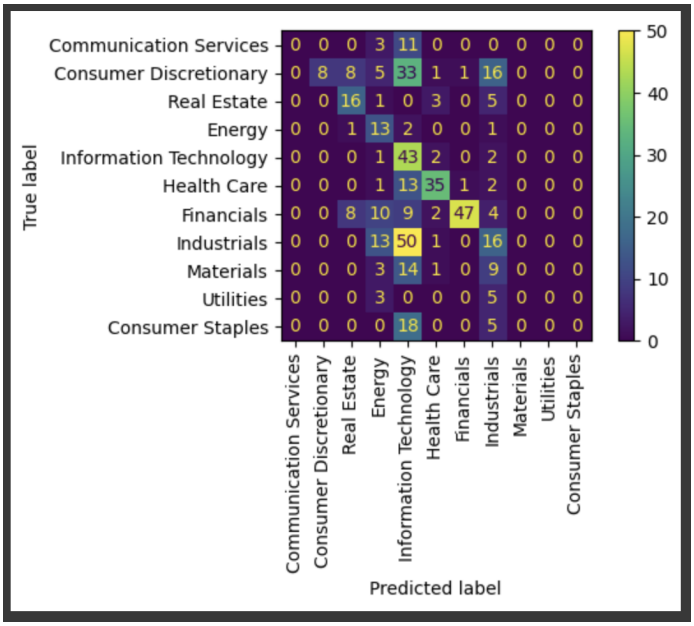

Figure 2: Bag-of-words clustering result

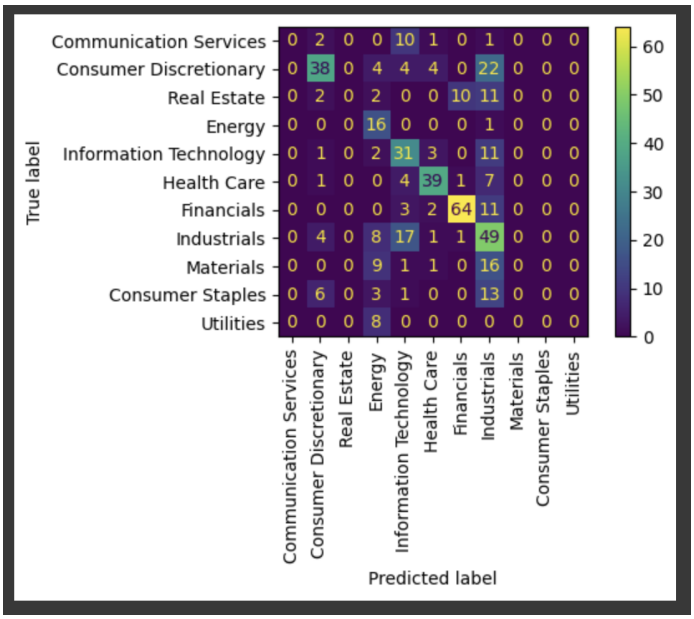

Figure 3: Word Embedding clustering result

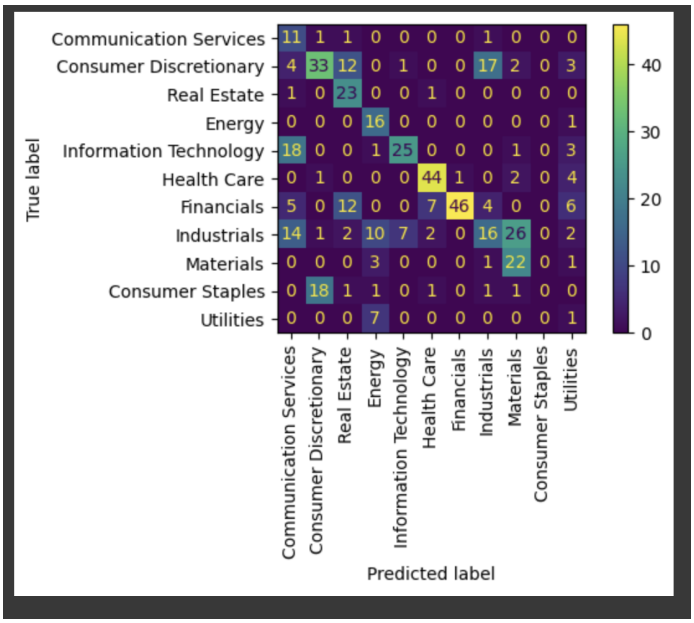

Figure 4: Document Embedding clustering result

Sector Label Accuracy			
Methods	Bag-of-words	Word Embedding	Document Embedding
Accuracy	40.00%	53.14%	53.14%

4.2 Sentiment Analysis

From our data, we were able to build a number of graphs and correlation tables comparing the strength of sentiment analysis with both the Vader and TextBlob sentiment analyzers and the data from the Management's Discussion and Risk Factors sections of the 10-Ks.

4.2.1 Individual correlations between returns and sentiments

The first structure of results we looked at was the main focus we came into the project with, is there a correlation between the sentiments for these sections and the corresponding stock returns from the release date. Once our data was processed, the average sentiments were calculated, and we constructed plots and correlation tables on the data to visualize it.

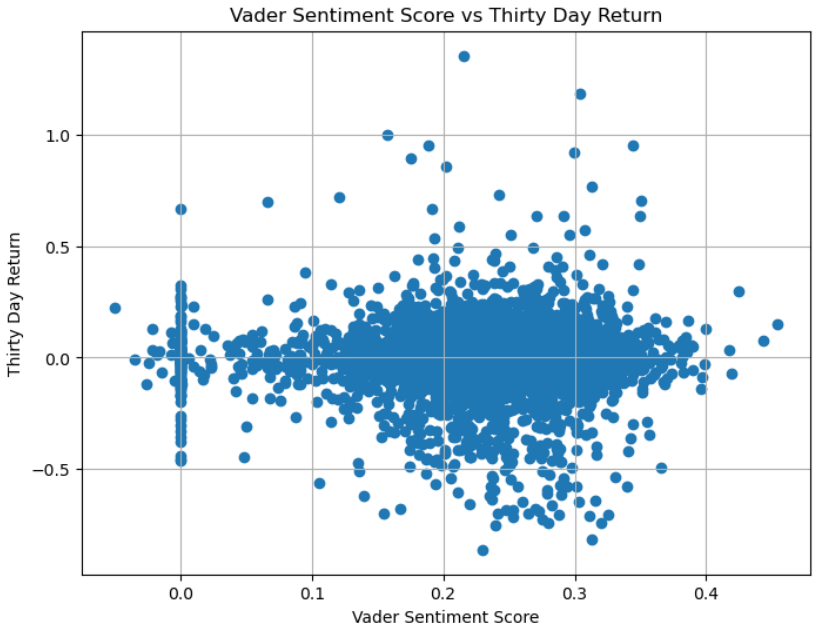

Figure 5: 30-day Vader correlation for Management's Discussion scatterplot

We initially went with a scatter plot to show the data but found that it was not a very effective visualization tool and opted for the box and whisker plot to better understand the data. Comparing the two plots you get a stronger

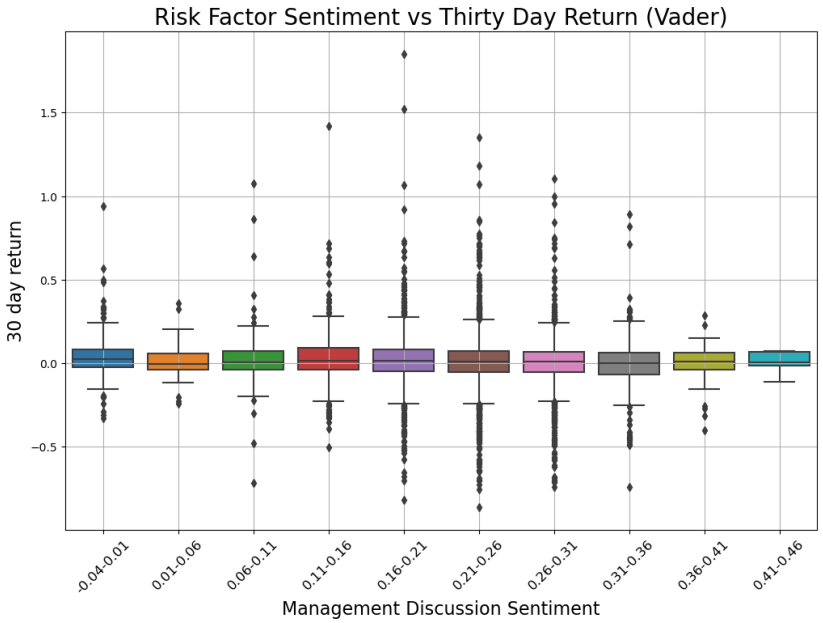

Figure 6: 30-day Vader correlation for Management's Discussion boxplot

sense from the box plot that throughout the changes in the sentiment scores, the returns tend to stay in a similar breakdown. As you can see from our correlation tables (located on the following page), our largest magnitude of correlation for the Risk Factors section was 0.0156 from the Pearson correlation on Vader sentiment with 5 day returns. On the Management's Discussion side, our largest magnitude was -0.0495 from the Spearman rank correlation on Vader sentiment with 30 day returns. Overall, these correlations were not quite as strong as we would've hoped, but that was slightly expected based on the lag in the release of the 10-Ks. These stock returns were for the dates after the release of the 10-K, whereas the data that goes into these 10-Ks was known long before its release. Therefore, with the lag, we are not surprised that our correlations were not as strong as was hoped. One interesting piece to note was that correlations for Vader were stronger than for TextBlob in each section of returns besides the 5 day returns. For Risk Factors, TextBlob had a higher correlation for 2 out of the 3 correlations, and for Management's Discussion, TextBlob was higher in all instances. In the case of Risk Factors, no trends are presented from these results correlations are varied throughout. However, on the Management's Discussion side, we do see larger magnitude results from the 30 day returns. Also interesting to note that these are all negative. This may be a result from in the larger horizons putting more weight on the company from a fundamental perspective which may be found from their Management's Discussion. These correlation are much larger from the rest, but in conclusion of

our analysis, there was not a strong enough correlation between sentiments and returns to suggest this would be a profitable investing strategy to implement.

Table 1: Correlations for Management's Discussion

		Correlation	
		Vader	TextBlob
One Day Return	Pearson	0.0036	−0.0021
	Kendall Tau	0.0071	0.0019
	Spearman Rank	0.0112	0.0028
Five Day Return	Pearson	0.0022	0.0071
	Kendall Tau	0.0046	0.0061
	Spearman Rank	0.0072	0.0093
Thirty Day Return	Pearson	−0.0469	−0.0199
	Kendall Tau	−0.0332	−0.0103
	Spearman Rank	−0.0495	−0.0152

Table 2: Correlations for Risk Factors

		Correlation	
		Vader	TextBlob
One Day Return	Pearson	0.0048	−0.0135
	Kendall Tau	0.0019	−0.0121
	Spearman Rank	0.0027	−0.0182
Five Day Return	Pearson	0.0156	−0.0067
	Kendall Tau	0.0051	−0.0056
	Spearman Rank	0.0076	−0.0084
Thirty Day Return	Pearson	0.0023	0.0002
	Kendall Tau	−0.0030	−0.0015
	Spearman Rank	−0.0045	−0.0026

4.2.2 Example of Sentiment Score

Here is an example of a Risk Factor section that was given a strong negative sentiment score. It is for ticker MATW, from the date of 2006-12-13, and was given a sentiment score on Vader -.34125, one of the most negative of the scores.

> ITEM 1A. RISK FACTORS. Risk factors specific to the Company relate primarily to the Casket segment and include Civil Investigative Demands from the Attorneys General in Maryland, Florida and Connecticut and the potential loss of the segment's largest independent distributor of caskets. Each of these factors is described more fully in Item 3, "Legal Proceedings" of this Form 10-K. Other

general risk factors that could affect the Company's future results principally include changes in domestic or international economic conditions, changes in foreign currency exchange rates, changes in commodity pricing that affect the cost of materials used in the manufacture of the company's products, changes in death rates, changes in product demand or pricing as a result of consolidation in the industries in which the company operates, changes in product demand or pricing as a result of domestic or international competitive pressures, unknown risks in connection with the Company's acquisitions, and technological factors beyond the Company's control. Although the Company does not have any customers that would be considered individually significant to consolidated sales, changes in the distribution of the Company's products or the potential loss of one or more of the Company's larger customers could be considered a risk factor. These factors are also included in this Form 10-K under the caption "Cautionary Statement Regarding Forward-Looking Information."

4.2.3 Comparing Vader and TextBlob Sentiment Analyzers

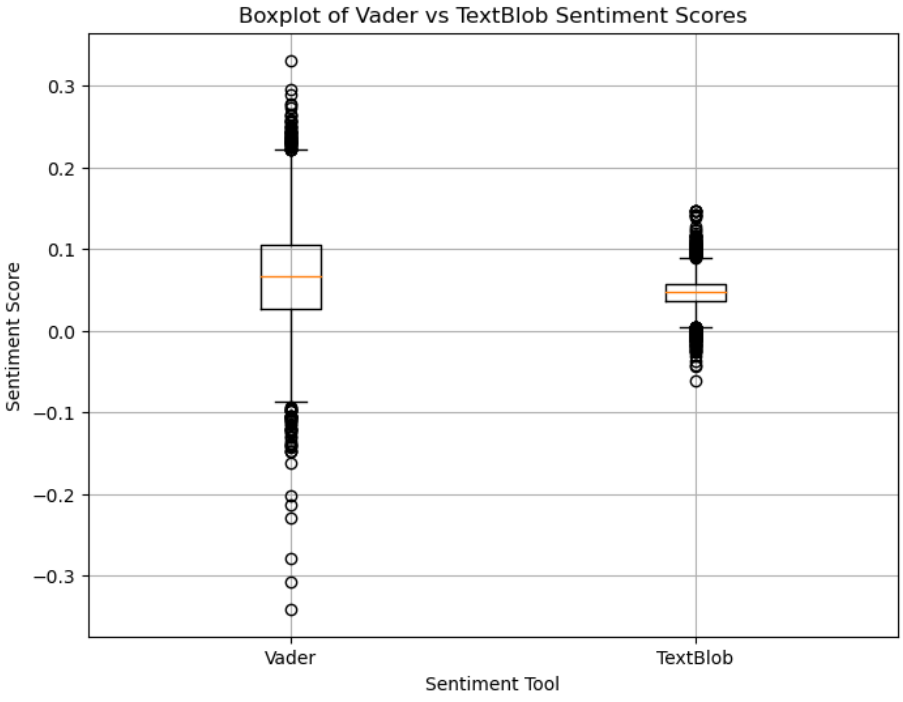

Figure 7: boxplot for vader and textblob sentiment scores

Another area we decided to look into were the tools we used in our study. The two main tools for this section are our sentiment analyzers, Vader and TextBlob.

Table 3: Correlations for Vader vs TextBlob

Correlation Type	Risk Factors	Management's Discussion
Pearson	0.2891	0.4873
Kendall Tau	0.1801	0.2902
Spearman Rank	0.2659	0.4158

When looking at the graph, we see the values for sentiment scores for both Vader and TextBlob from the perspective of a box plot. The spread of these scores stands out the most, as Textblob has a much smaller spread, with all of its data barring outliers falling between 0.0 and 0.1. On the other hand, Vader had about 50% of its data falling in that range, with its full data barring outliers being from -.095 to 0.22.

TextBlob's methodology, which assigns pre-defined sentiment scores to individual words or phrases and aggregates them to compute an overall sentiment score, may lead to more nuanced and contextually-appropriate sentiment assessments, contributing to a narrower spread of sentiment scores.

Additionally, TextBlob's handling of ambiguous or neutral words might involve assigning more nuanced sentiment scores, thereby reducing the variability in sentiment assessments and resulting in a smaller spread of sentiment scores compared to Vader.

As for the higher correlations between Vader and TextBlob for Management's Discussion we see in the table, there are a number of reasons that support this. Management's Discussion sections often focus on summarizing past performance, discussing future strategies, and providing insights into the company's operations. The relatively consistent and positive tone of such discussions may lead to more agreement between TextBlob and Vader in sentiment assessment, resulting in higher correlations compared to the varied and potentially negative tone of Risk Factors discussions.

Management's Discussion sections typically contain information that is more directly related to the company's overall performance, financial outlook, and strategic initiatives. As a result, both TextBlob and Vader may interpret the sentiment of such content in a more consistent manner, leading to higher correlations between their sentiment scores compared to the broader and potentially less directly related content found in Risk Factors sections.

4.2.4 Comparing Management's Discussion and Risk Factors

When looking at our data, we were focused on two separate sections. Here we decided to look at how they compare to each other when put together.

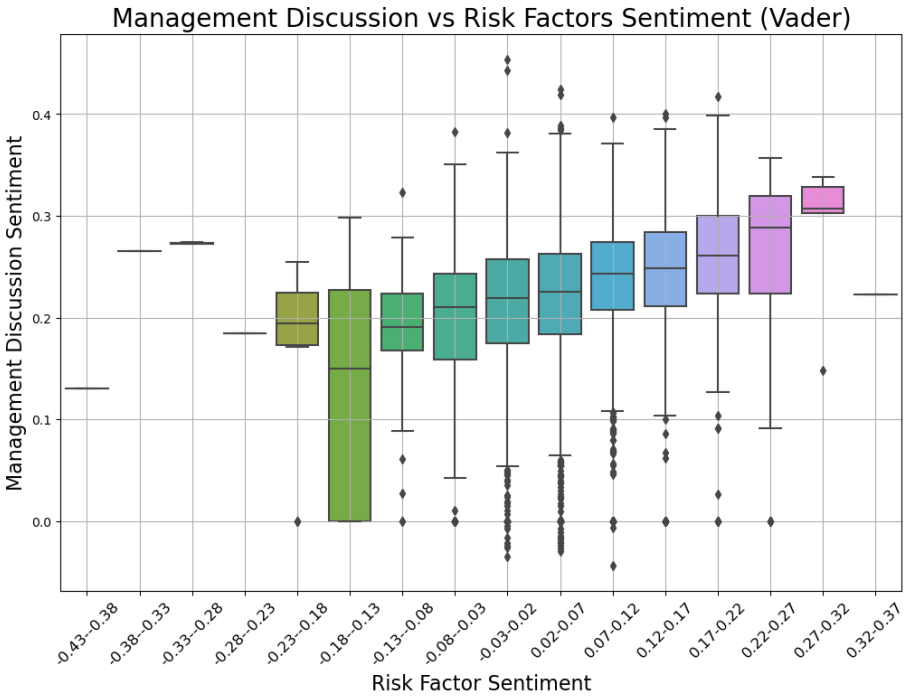

Figure 8: box plot for management discussion and risk factors sentiment scores

This graph here shows the sentiment scores for Risk Factor on the x and Management's Discussion on the y. Over intervals of.05 for Risk Factor scores, Management's Discussion scores are placed into boxes. From this, we can see the positive correlation between Risk Factors and Management's Discussion. This we expected before we began our research: intuitively, if there are fewer risks faced by a business, you would expect a more positive outlook from the management. We were not surprised to see as well that Management's Discussion featured very few sentiment scores below 0.0. This section is where those at the company are given a chance to speak on how the business went over the previous year, and it is most evident that those people will want to spin whatever tale as positively as they can to increase others outlook on the business.

4.2.5 Analysis of WordClouds

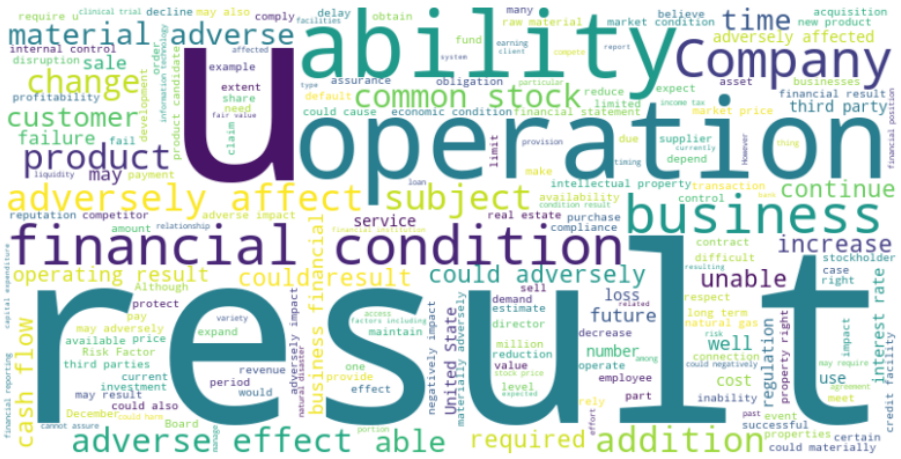

Figure 9: Wordcloud for Risk Factors

The first word cloud, representing Risk Factors, illuminates the spectrum of concerns a company faces and discloses in its 10-K filings. Terms such as "ability," "operation," and "company" suggest a focus on the company's operational capabilities. When a company emphasizes its "ability," it often refers to the competencies or capacities that are crucial for successful business performance. This might encompass everything from maintaining competitive advantages to adapting to market changes.

The term "financial condition" is pivotal because it encapsulates the company's financial health, including liquidity, solvency, and overall stability. A strong emphasis on this phrase indicates that financial robustness is a significant point of consideration, possibly reflecting on the company's resilience against market fluctuations and economic downturns.

"Adverse effect" implies that the risks identified could potentially have negative impacts on the company's profitability, market share, or growth trajectories. Such risks could range from regulatory changes to unexpected market dynamics or competitive pressures. Seeing "adverse effects" prominently can be a cue for stakeholders to seek out specific risks that the company is particularly concerned about, such as environmental liabilities, legal challenges, or technology disruptions.

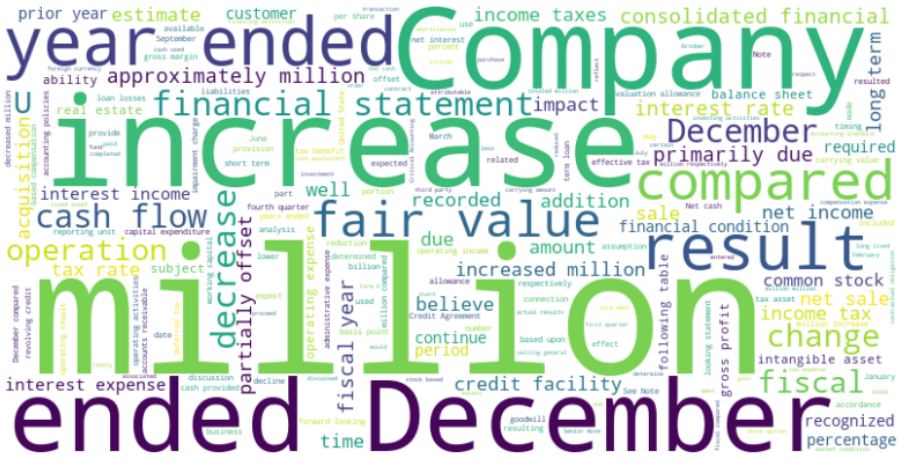

Figure 10: Wordcloud for Management Discussion

Moving on to the second word cloud, focused on the Management Discussion section, the concentration on terms like "increase," "million," "financial statements," "year ended," and "December" paints a picture of the company's financial narrative over the designated fiscal year. The prominence of "increase" might denote positive growth trends, improvements in revenue or profits, or perhaps expansion in market share or operational capacities.

"Million" and "financial statements" likely refer to the quantitative aspects of the company's financial disclosures, possibly touching on revenue, profits, capital expenditures, or investments. These figures are fundamental to understanding the company's financial position and are key indicators of its performance.

The specific mention of "year ended" and "December" anchors the discussion to the fiscal year's end, which is a critical time for financial reporting and reflection. This period marks when companies sum up their annual performance, providing a comprehensive overview that investors and analysts scrutinize for indications of financial health and strategic direction.

The repetition of temporal markers and financial terms suggests that the MDA section is strongly oriented towards discussing temporal changes, financial metrics, and possibly the expectations for future financial periods. It's a section where the company not only presents its financial outcomes but also contextualizes them, explaining the reasons behind fluctuations, comparing them to past performance, and forecasting future expectations.

In both word clouds, the repeated terms offer a gateway into the company's self-assessment and strategic priorities.

5 Summary

Applying modern natural language processing techniques, including text clustering and sentiment analysis, to the 10-K annual reports dataset, we have found that there is much valuable information that can be used to deepen our understanding of each stock's business and even be related to changes in returns.

With the result of text clustering on business descriptions, we can see that the business descriptions of some sectors are similar while those of other sectors are quite different. Overall, the clusters given by the methods we used do not match the sector labels pretty well. However, it is clear that document embedding and word embedding provide much better results than bag-of-words, indicating the importance of extracting meaning over simply counting words in common. Potential future explorations or applications of this task could be checking correlations of returns in a cluster. It would be interesting to know whether the cluster given by the text clustering method has stocks with similar performance since they are considered to have similar business descriptions.

For the correlation analysis, three statistical methods were utilized: Pearson correlation coefficient, which measures linear relationships; Kendall Tau, a non-parametric test that assesses associations based on the ranking of data; and Spearman Rank, another non-parametric method focused on monotonic relationships. These methods were chosen for their different sensitivities to data distribution and outlier robustness.

The results presented in the tables show a range of correlation coefficients, most of which are close to zero, indicating a very weak linear relationship between sentiment scores and stock returns for both sections of the 10-K filings. The analysis was further refined by investigating individual sentence scores rather than whole sections to calculate average sentiment scores, potentially leading to a more nuanced understanding of the text.

The comparison between VADER and TextBlob sentiment scores revealed notable differences, with TextBlob generally showing a narrower spread of sentiment scores, as visualized in the boxplot. TextBlob, which assigns pre-defined sentiment scores to words or phrases and aggregates these to compute an overall sentiment score, may offer more contextually appropriate sentiment assessments, leading to a more nuanced spread of sentiment scores. Conversely, VADER, which uses a lexicon and rule-based model, showed a wider spread of sentiment scores, suggesting a different sensitivity to the textual data.

In summary, while both sentiment analysis tools provided insight into the sentiment of the textual data from 10-K filings, their correlation with short-term stock performance appeared to be minimal. The study's comparison between VADER and TextBlob highlighted differences in their analytical approaches and the resulting sentiment score distributions. These findings contribute to the un-

derstanding of how sentiment analysis can be applied to financial documents and its potential impact on stock market behavior.

References

[1] Andrew M. Dai, Christopher Olah, and Quoc V. Le. "Document Embedding with Paragraph Vectors". In: *CoRR* abs/1507.07998 (2015). arXiv: 1507.07998. URL: http://arxiv.org/abs/1507.07998.

[2] Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. "Linguistic Regularities in Continuous Space Word Representations". In: (June 2013). Ed. by Lucy Vanderwende, Hal Daumé III, and Katrin Kirchhoff, pp. 746–751. URL: https://aclanthology.org/N13-1090.

[3] Tomas Mikolov et al. "Efficient Estimation of Word Representations in Vector Space". In: (2013). arXiv: 1301.3781 [cs.CL].

[4] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. "GloVe: Global Vectors for Word Representation". In: *Empirical Methods in Natural Language Processing (EMNLP)*. 2014, pp. 1532–1543. DOI: 10.3115/v1/D14-1162. URL: http://www.aclweb.org/anthology/D14-1162.

[5] Wisam Qader, Musa M. Ameen, and Bilal Ahmed. "An Overview of Bag of Words;Importance, Implementation, Applications, and Challenges". In: (June 2019), pp. 200–204. DOI: 10.1109/IEC47844.2019.8950616.

Appendix

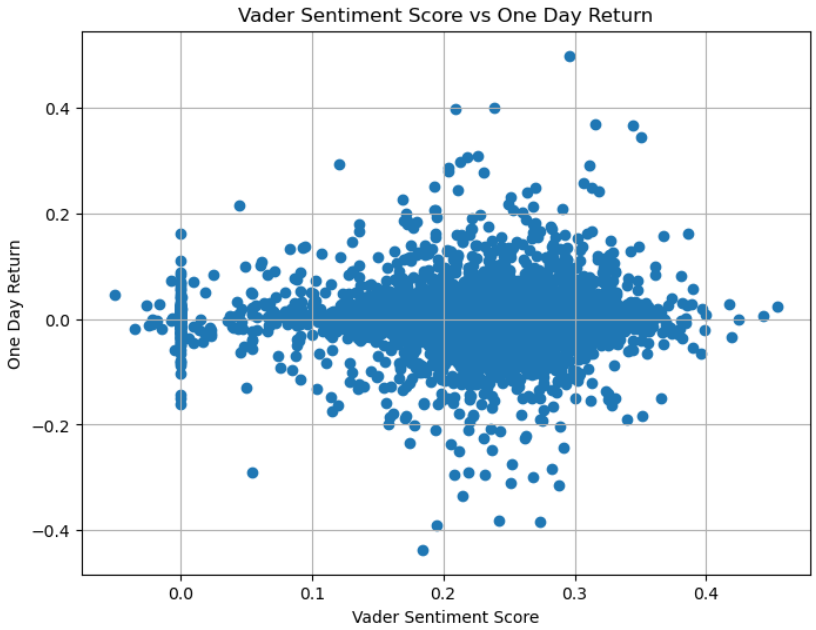

Figure 11: 1-day vader correlation

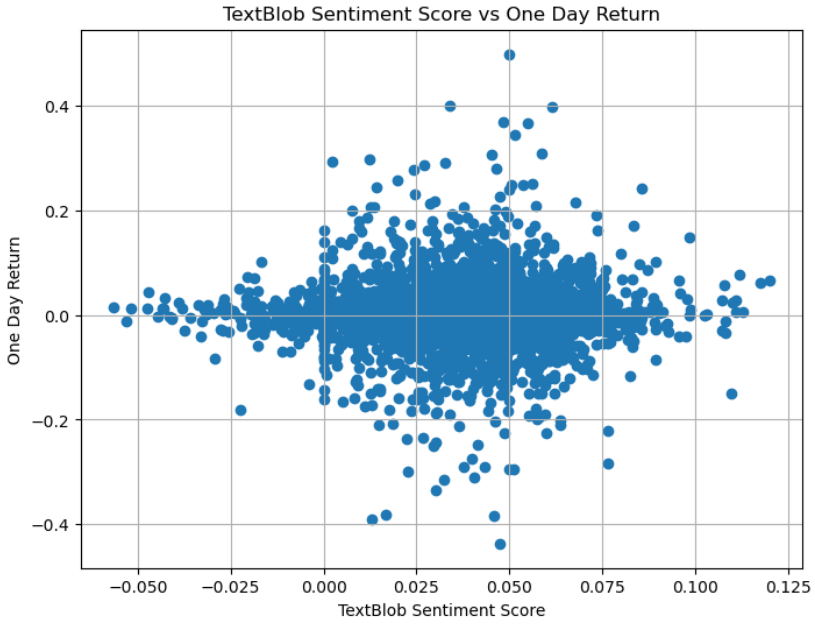

Figure 12: 1-day tb correlation

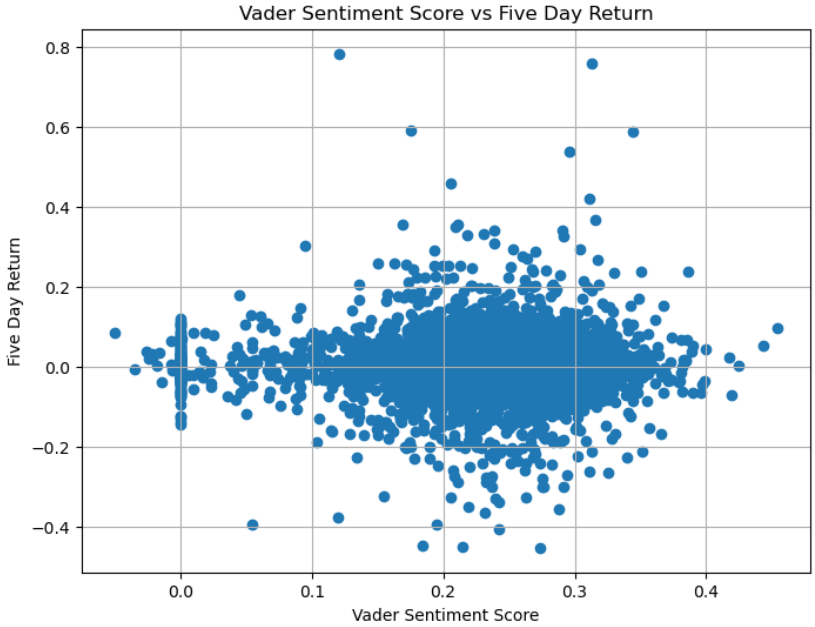

Figure 13: 5-day vader correlation

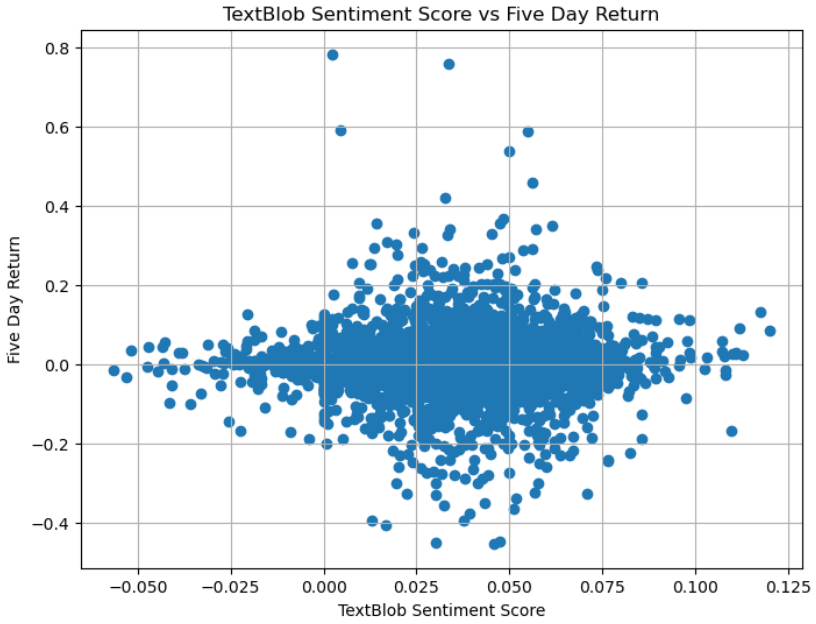

Figure 14: 5-day tb correlation

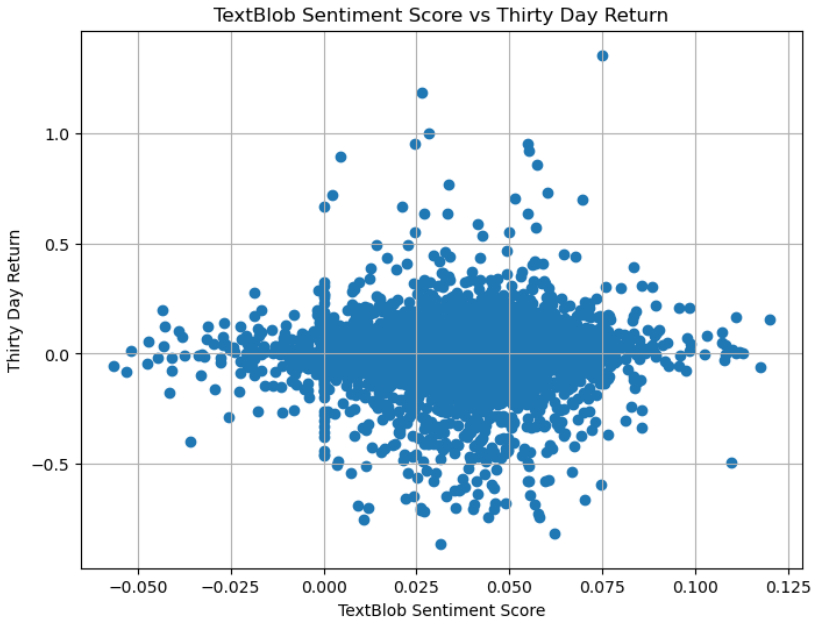

Figure 15: 30-day tb correlation

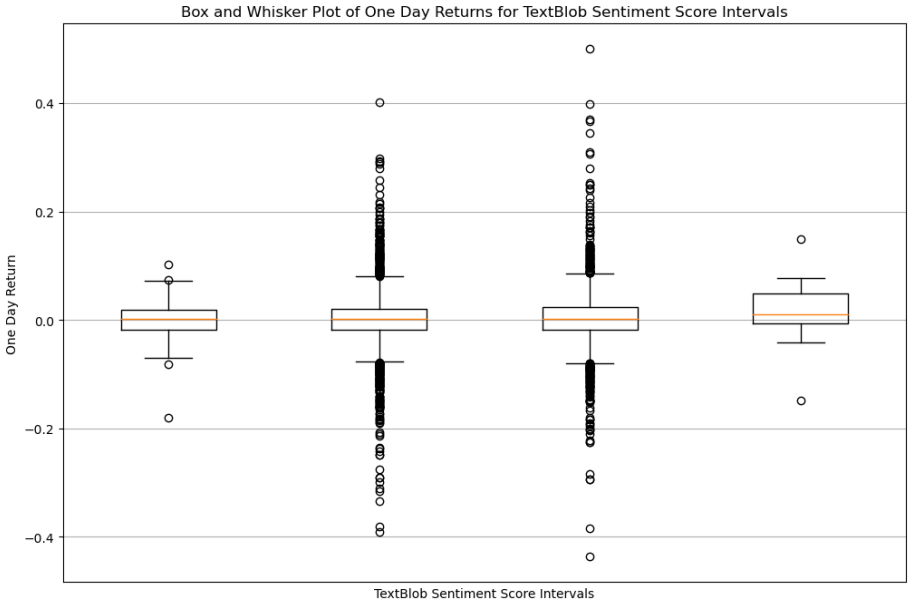

Figure 16: boxplot for one-day return textblob

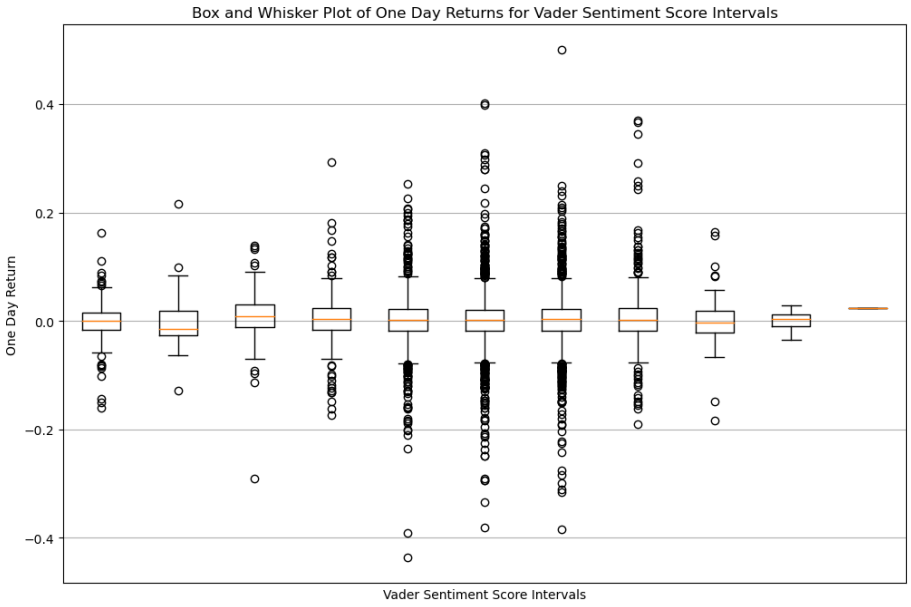

Figure 17: boxplot for one-day return vader

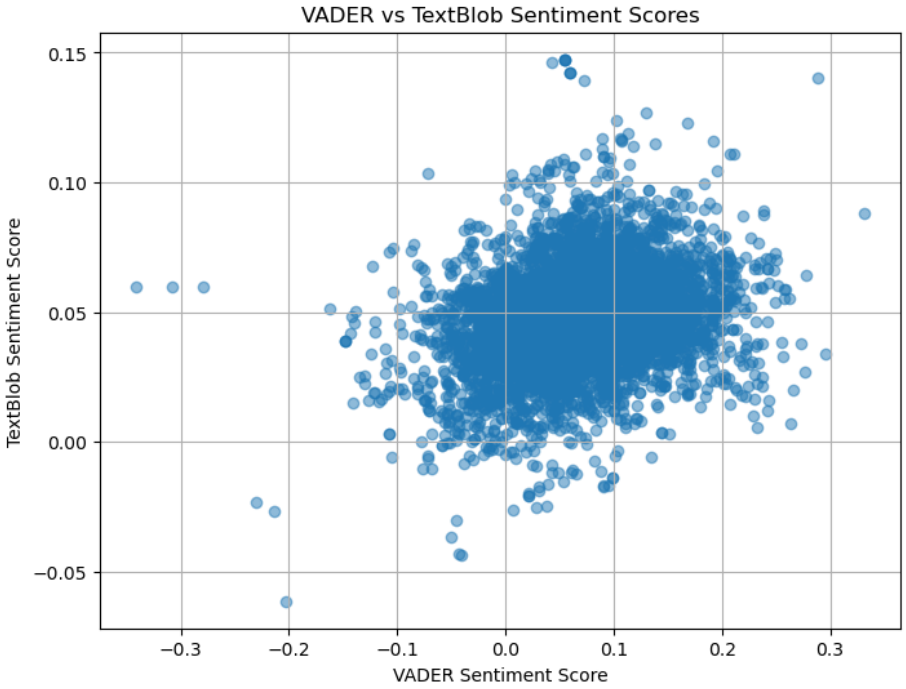

Figure 18: scatter plot for vader vs textblob sentiments

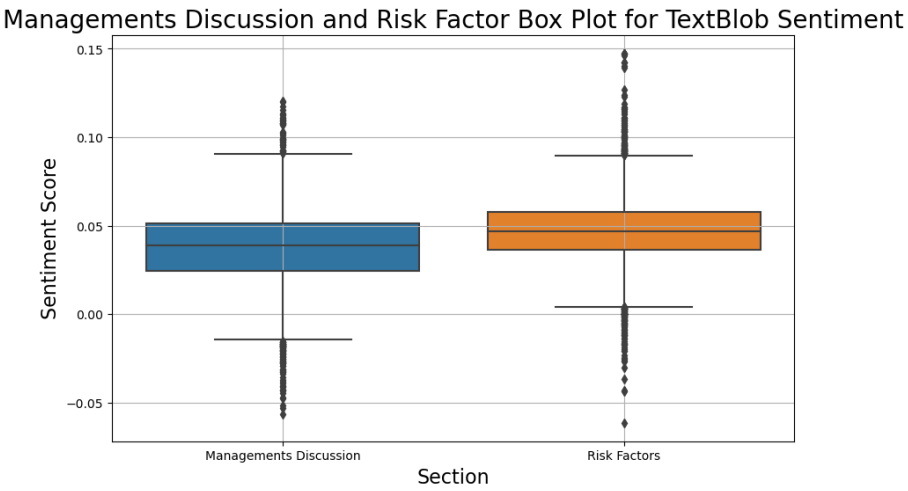

Figure 19: box plot for MD and RF sentiments (vader)

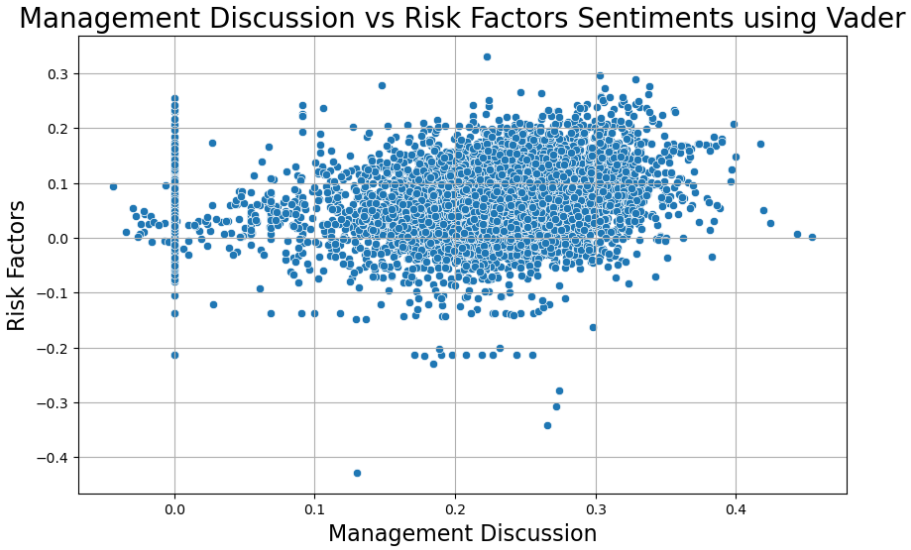

Figure 20: scatter plot for MD vs RF sentiments (vader)

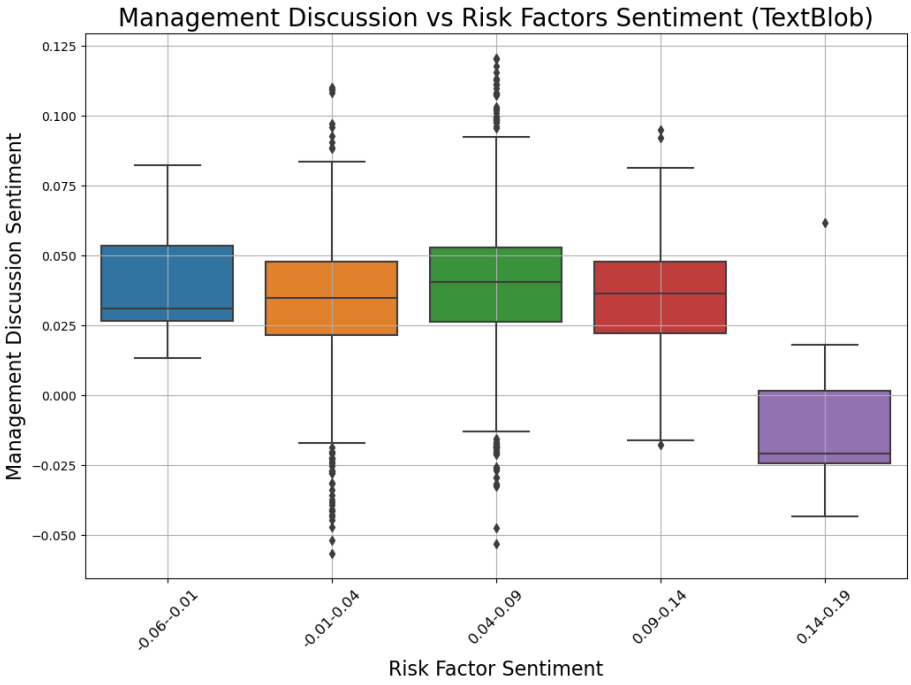

Figure 21: box plot for MD vs RF sentiments (textblob)