ST494: Statistical Learning

Owais Soomro: 180682890

Bilal Naeemuddin: 180425370

Justin Lee: 190954930

Taha Khurram: 193071840

GitHub: https://github.com/Owaissoomro/machine-learning-project

Abstract:

The aim for this research was to explore different ways of sentiment analysis on risk factors of 10-K reports using different classification methods. We used BOW, Naïve Bayes, Text Blob Classification and LDA to perform sentiment analysis using the dataset acquired through Sec API. We can conclude that BOW model performed well as the dictionary we used was specifically for financial sentiment analysis.

Table of Contents

Introduction to Natural Language Processing

Founded by Roger Schank in 1969, Natural Language Processing, commonly known as NLP, is a subsect of machine learning that aids computers with analyzing text quickly. NLP is a powerful tool to understand, analyze, manipulate, and potentially generate a human language. It examines the grammatical structure of sentences and the meaning of words in a specific context, then uses algorithms to extract meaning and deliver outputs (Olsson 2009). Some common examples of NLP are Google Translate, Grammarly & Alexa.

How does NLP work?

There are three main steps of NLP model. The lexical analysis is the first phase of the compiler also known as a scanner. It converts the high-level input program into a sequence of Tokens. Secondly, the syntactic analysis analyzes text using basic grammar rules to identify sentence structure, word organization, and how words relate to each other. Some of the main-sub tasks for syntactic are tokenization, PoS (Part of Speech) tagging, lemmatization, and stop-word removal. Lastly, the semantic analysis focuses on capturing the meaning of the text. It studies the meaning of each word; then, it looks at the combination of words and what they mean in the context. The two main subtasks are word sense disambiguation and relationship extraction. (Daelemans 2002)

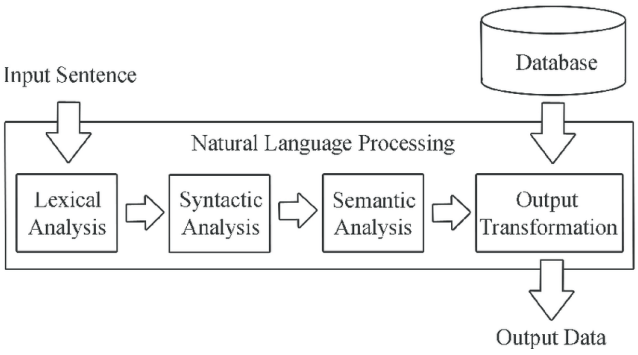

Figure 2: Natural Language Processing Framework

Source: ***https://www.oak-tree.tech/blog/data-science-nlp***

NLP in Finance:

The usage of NLP in finance is vast. Firstly, NLP is used to enhance, improve or generate signals for a particular strategy. An event-driven strategy is a perfect example of leveraging NLP. It can scrape, automate, and classify financial events, allowing you to have a set of events in your calendar to generate alpha for the portfolio. Furthermore, NLP can be used to complement research. American Century Investments uses NLP to complement its research with a sentiment model that aims to detect deception in management commentary/language during quarterly earning calls. Moreover, NLP is used to monitor companies. Deutsche Bank implemented an NLP algorithm to find whether the commitment made by firms to reduce carbon emissions correlated with achieved sustainability performance. NLP has extensive use in the capital markets industry as firms can use it to understand and interpret any textual information (Man. Luo, Lin 2019)

Research: Sentiment Analysis of 10-K Risk Section

Acquiring Data: SEC API

Now that we have learned what NLP and ML are, we can start with our research. Since we are performing sentiment analysis on 10-K's, the first step is to acquire data through various filings and reports as our raw data. To get 10-K annual fillings, we use the SEC API. This is a two-step process. We use an HTTP library to be able to make API Calls such as Requests, and we use a library to parse JSON formatted objects easily using Pandas (Mishdev 2020). We create a function to extract the risk section (1A) text to get our desired output (Figure 4).

	company	url	text
0	TSLA	https://www.sec.gov/Archives/edgar/data/131860...	ITEM 1A. RISK FACTORS\n\nYou should carefully...
1	AMZN	https://www.sec.gov/Archives/edgar/data/101872...	Item 1A. \n\nRisk Factors \n\nPlease carefull...
2	GOOGL	https://www.sec.gov/ix?doc=/Archives/edgar/dat...	ITEM 1A. RISK FACTORS \n\nOur operations and f...
3	VLO	https://www.sec.gov/ix?doc=/Archives/edgar/dat...	ITEM 1A. RISK FACTORS \n\nYou should carefully...
4	MCK	https://www.sec.gov/ix?doc=/Archives/edgar/dat...	Item 1A. Risk Factors. \n\nOther than factual ...
5	BA	https://www.sec.gov/ix?doc=/Archives/edgar/dat...	Item 1A. Risk Factors \n\nAn investment in our...

Figure 4: Data through SEC API

Pre-Processing: Tokenizing, Lemmatization and Stemming

Now that we have the data, the next step is to break down the text into small units called tokens. The process of simply breaking a text into words is known as tokenization. We need tokenization to break unstructured data and natural language text into chunks of information that can be considered discrete elements. The token occurrences can further be used directly as vectors representing that document. Moreover, stemming is a process of producing morphological variations of the base word. An example would be the word boat might also return terms such as "boats" and "boating." In contrast, lemmatization looks beyond word reduction and considers a language's entire vocabulary to apply a morphological analysis to words. For example, the lemma of 'was' is 'be,' and the lemma of 'mice' is 'mouse.' To pre-process our data, we use the Natural Language Tool Kit, commonly known as NLTK (Dixit 2018). NLTK is an open-source library that processes natural language through tokenizing, stemming, and lemmatizing (Figure 5).

	company	url	text	tokenized_text
0	TSLA	https://www.sec.gov/Archives/edgar/data/131860...	ITEM 1A. RISK FACTORS\n\nYou should carefully...	[ITEM, 1A, ., RISK, FACTORS, You, carefully, c...
1	AMZN	https://www.sec.gov/Archives/edgar/data/101872...	Item 1A. \n\nRisk Factors \n\nPlease carefull...	[Item, 1A, ., Risk, Factors, Please, carefully...
2	GOOGL	https://www.sec.gov/ix?doc=/Archives/edgar/dat...	ITEM 1A. RISK FACTORS \n\nOur operations and f...	[ITEM, 1A, ., RISK, FACTORS, Our, operations, ...
3	VLO	https://www.sec.gov/ix?doc=/Archives/edgar/dat...	ITEM 1A. RISK FACTORS \n\nYou should carefully...	[ITEM, 1A, ., RISK, FACTORS, You, carefully, c...
4	BA	https://www.sec.gov/ix?doc=/Archives/edgar/dat...	Item 1A. Risk Factors \n\nAn investment in our...	[Item, 1A, ., Risk, Factors, An, investment, c...

Figure 5: Tokenization of Risk Section

Source: Jupyter Notebook

Classification: General Overview

Now that we have our pre-processed data, our goal is to classify our data set. Classification is a process of categorizing the data into different classes allowing it to be labeled. There are various classification methods, but we will focus on binary classification for our research. Binary classification is between two class labels (Figure 6). It involves one class, the normal state, and another class, the abnormal state. An example is "cancer not detected," which is the normal state of a task that involves a medical test, and "cancer detected" is the abnormal state. In our research, we will classify our data into two classes; positive and negative.

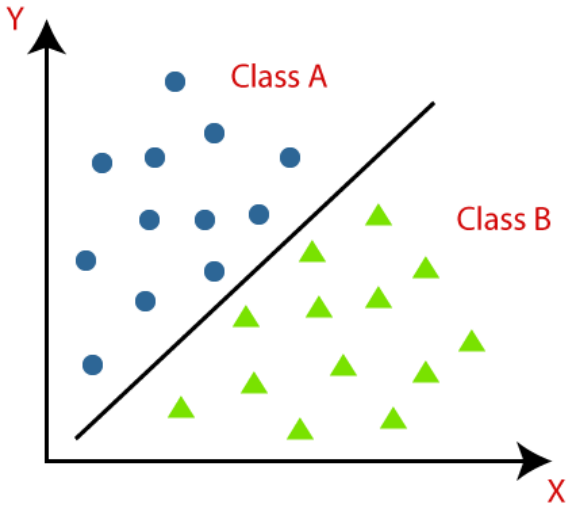

Figure 6: Binary Classification

Source: ***https://www.javatpoint.com/classification-algorithm-in-machine-learning***

Classification Model 1: Bag of Word Model

A bag of words model is a way of extracting features from the text used in modeling or machine learning algorithms. It is called a "bag" of words because any information about the document's structure or order is discarded (Zhou 2019). The model is only concerned with whether known words occur in the document, not where (Figure 7).

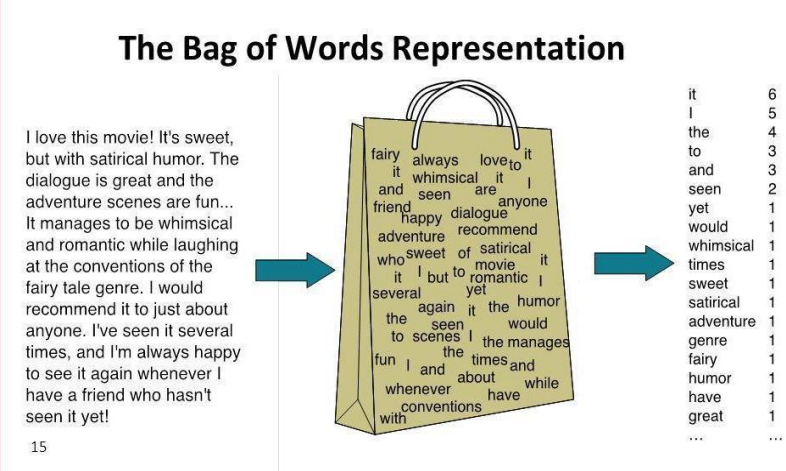

Figure 7: Bag of Words Model

Source: ***https://www.excelr.com/blog/data-science/natural-language-processing/implementation-of-bag-of-words-using-python***

We use Loughran & McDonald's dictionary as our known words in our sentiment analysis. The dictionary classifies the words into two classes: positive and negative. For example, words such as "abandon" and "bankruptcy" can be classified as negative, whereas words such as "opportunity" and "improved" are classified as positive. Now once we have our tokenized data set, we use the model to answer these questions:

1. How many negative words are there in the dataset & what are those words?
2. How many positive words are there in the dataset & what are those words?

Once we have the bag of words model output for each 10-K risk file, we try to create a cumulative sentiment using the bag of word sentiment score. A sentiment score is between -1 to +1, where -1 represents negative, and +1 represents positive. This sentiment score explains the intensity of the overall sentiment, explaining whether the 10-K Risk section is leaning towards positive or negative (Figure 8).

	company	url	text	word_count	pos_count	neg_count	pos_words	neg_words	sentiment_bow
0	TSLA	https://www.sec.gov/Archives/edgar/data/131860...	ITEM 1A. RISK FACTORS\n\nYou should carefully...	12962	89	460	[stable, able, successfully, achieve, successf...	[adversely, closed, stringent, suspended, susp...	-0.675774
1	AMZN	https://www.sec.gov/Archives/edgar/data/101872...	Item 1A. \n\nRisk Factors \n\nPlease carefull...	6624	60	293	[greater, greater, better, alliances, enhanced...	[adversely, strain, strain, able (with negatio...	-0.660057
2	GOOGL	https://www.sec.gov/ix?doc=/Archives/edgar/dat...	ITEM 1A. RISK FACTORS \n\nOur operations and f...	11728	79	621	[effective, able, superior, successfully, inno...	[harm, loss, harm, terminate, harm, difficult,...	-0.774286
3	VLO	https://www.sec.gov/ix?doc=/Archives/edgar/dat...	ITEM 1A. RISK FACTORS \n\nYou should carefully...	9466	37	347	[profitability, advances, desirable, improveme...	[adversely, adversely, volatile, volatile, vol...	-0.807292
4	BA	https://www.sec.gov/ix?doc=/Archives/edgar/dat...	Item 1A. Risk Factors \n\nAn investment in our...	8767	52	351	[successfully, stability, profitability, advan...	[downturn, severe, instability, challenges, di...	-0.741935

Figure 8: Bag of Word Model Results

Classification Model 2: Naïve Bayes Classification

Naïve Bayes is a supervised classification algorithm based on Bayes theorem. The algorithm is called "Naïve" because it assumes that each feature is independent of the other features, which is not the case in real life. To understand the Naïve Bayes algorithm, we need to understand Bayes Theorem. Bayes' theorem describes the probability of an event based on knowledge of conditions that might be related to the event. For example, two cards are drawn without replacement from the deck. What is the probability that the second card is an ace, given that the first card drawn was also an ace? This is where the Bayes comes in as it uses conditional probability (Figure 9).

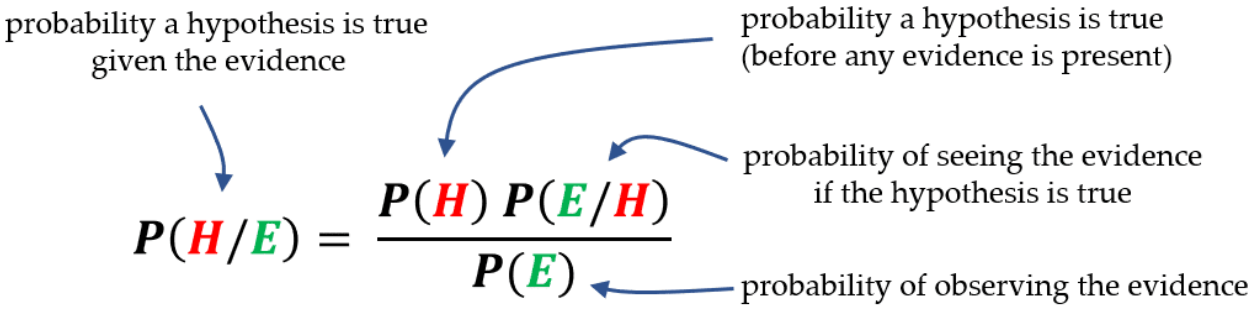

Figure 9: Bayes Theorem

Source: ***https://www.gaussianwaves.com/2021/04/bayes-theorem/***

Now that we have understood how Bayes Theorem works, we can apply it to our sentiment analysis. We use a Multinomial Naïve Bayes, where the features are assumed to be generated from a simple multinomial distribution. A limitation of the Naïve Bayes algorithm is the "Zero Conditional Probability Problem," meaning for features having zero frequency; the total probability also becomes zero. To fix this problem, we use a smoothing technique called Laplace Smoothing. Laplace Smoothing ensures that our posterior probabilities are never zero by adding 1 to the numerator and d (Number of dimensions in the data set) to the denominator and incorporating a smoothing parameter represented by alpha (α) (Ratz 2022) (Figure 10).

$$p_{i,\ \text{empirical}} = \frac{x_i}{N}$$

but the posterior probability when additively smoothed is

$$p_{i,\ \alpha\text{-smoothed}} = \frac{x_i + \alpha}{N + \alpha d};$$

Figure 10: Laplace Smoothing

Source: ***https://www.quora.com/What-is-Laplacian-smoothing-and-why-do-we-need-it-in-a-Naive-Bayes-classifier***

Now that we have our algorithm ready, we need to train it so it can predict between positive and negative. Due to the lack of 10-K data sets, we used the Cornell IMDB review data set and split the data set into two. We use 80% of our data set to train and 20% to test. The main question that arises is how you evaluate the performance of our machine learning model. We use a confusion matrix, accuracy, precision, recall & F-1 score. Firstly, the confusion matrix consists of four elements: True Positive, True Negative, False Positive, and False Negative. True positive &

negative happens when the model has predicted accurately, whereas false positive & negative occurs when the model has not predicted the data accurately (Figure 11).

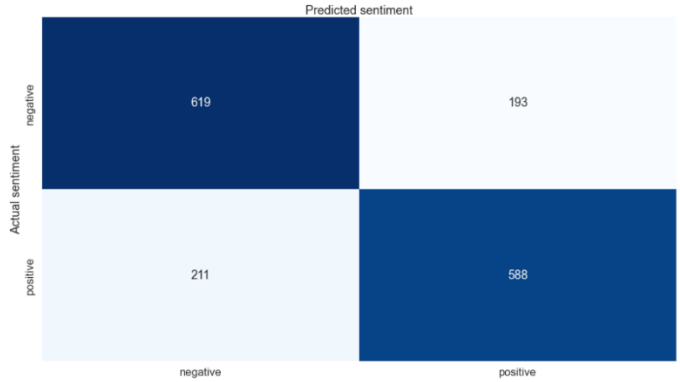

Figure 11: Confusion Matrix for IMDB Reviews

The data from the confusion matrix can be used to compute our accuracy score, which in essence explains how accurate our model predictions were. The accuracy score is a great measure, but it is not a good metric when the data set is unbalanced. An example is if 90 people are healthy and 10 people are unhealthy. The model is able to predict all 90 healthy people right, but it also classifies the 10 unhealthy people as "healthy". The accuracy score would be 90% which is not true since the model cannot classify accurately. Due to this problem, we use precision & recall. Precision ideally explains the precision of our model, whereas recall explains the sensitivity and true positive rate. Both the metrics: Precision and Recall, are combined to compute the F-1 score. F-1 score is the harmonic mean of precision and recall (Figure 12).

	precision	recall	f1-score	support
negative	0.75	0.76	0.75	812
positive	0.75	0.74	0.74	799
accuracy			0.75	1611
macro avg	0.75	0.75	0.75	1611
weighted avg	0.75	0.75	0.75	1611

Figure 12: Classification Report

Now that we have trained and evaluated our model, we test it on our 10-K data set. We classify the tokenized words into positive and negative, which is used to compute an average sentiment for the overall 10-k (Figure 13).

	company	url	sentiment_MNB	avg_sentiment_MNB
0	TSLA	https://www.sec.gov/Archives/edgar/data/131860...	[negative, negative, negative, negative, negat...	negative
1	AMZN	https://www.sec.gov/Archives/edgar/data/101872...	[negative, negative, negative, negative, negat...	negative
2	GOOGL	https://www.sec.gov/ix?doc=/Archives/edgar/dat...	[negative, negative, negative, negative, negat...	negative
3	VLO	https://www.sec.gov/ix?doc=/Archives/edgar/dat...	[negative, negative, negative, negative, negat...	negative
4	BA	https://www.sec.gov/ix?doc=/Archives/edgar/dat...	[negative, negative, negative, negative, negat...	negative

Figure 13: Navies Classifier Results

Classification Model 3: Latent Dirichlet Allocation

Latent Dirichlet Allocation (LDA) is a generative statistical model used for topic modeling, which is a method for unsupervised classification of documents, like clustering on numeric data, which identifies some natural groups of items (topics) that occur in a collection of documents. LDA is used by assuming that each document in a collection is generated from a mixture of topics, and each topic consists of a distribution over words. The model then infers the topics and their distributions by analyzing the words in the documents. LDA can be used to generate topics to understand a document's overall theme, and is often used in recommendation system, document summarizations, and document classification. We achieved an accuracy of 0.49 through LDA analysis.(Figure 14)

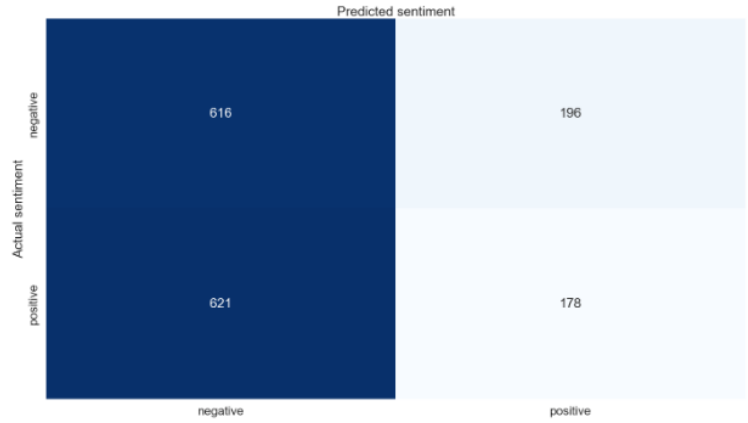

Figure 14: Latent Dirichlet Allocation Confusion Matrix

Classification Model 4: Text Blob Classification

TextBlob is an open-source python library built for natural language processing that classifies the sentiment by using its pre-trained inbuilt classifier. TextBlob explains the sentiments in two ways; polarity and subjectivity. Polarity lies between [-1,1], -1 defines a negative sentiment, and 1 defines a positive sentiment. Negation words reverse the polarity, whereas subjectivity quantifies the amount of personal opinion and factual information contained in the text. The higher subjectivity means that the text includes personal opinions rather than information. TextBlob has one more parameter — intensity. TextBlob calculates subjectivity by looking at the 'intensity.'

Limitation: Human Classification

Due to the lack of 10-K data sets, all the reports were ready manually and classified into positive and negative. Human classification is used as our base to see the performance of our machine learning classifiers.

Results

Now that we have classified our data through 4 different models, we can use the human classification as our benchmark to compare the results and performance our machine learning models (Figure 15).

	ticker	Sentiment BOW	Sentiment MNB	polarity	subjectivity	human_sentiment	Sentiment LDA
0	TSLA	-0.675774	negative	0.048224	0.412786	-0.9	positive
1	AMZN	-0.660057	negative	0.028300	0.397047	-0.8	negative
2	GOOGL	-0.774286	negative	0.063706	0.420971	0.2	negative
3	VLO	-0.807292	negative	0.034221	0.384977	-0.8	negative
4	BA	-0.741935	negative	0.045132	0.385268	-0.5	positive

Figure 15: Combined Results

Ticker 1: TSLA

Starting with Tesla's risk section, we computed its human sentiment to be -0.9 due to two main reasons. Firstly, Tesla is heavily reliant on an efficient supply chain, and due to significant disruption caused by the global pandemic has affected its overall financial status. Secondly, the EV sector has recently emerged, and the competitive landscape has saturated, allowing consumers to have a variety of options to choose from, which hinders Tesla's growth. As we have computed our human sentiment, we use our machine learning models to predict the sentiment of its risk section. According to BOW & MNB, the sentiment was -0.67, which signifies the negative sentiment of the risk section. Furthermore, the Textblob Classification classifies Tesla's Risk Section as a neutral sentiment, but due to a higher subjectivity, it can be explained that the 10-K risk text is an opinion-based analysis. Whereas LDA's sentiment is positive which in this case is not accurate as Tesla faces a lot of challenges.

Ticker 2: AMZN

Looking at Amazon's Risk section, we computed that its human sentiment is -0.8 due to the massive surge in volume because of lockdowns. The value Amazon provides to the consumer is to receive goods and services while at home. This was an opportunity for Amazon to increase its growth. To sustain that growth Amazon has introduced new products, services, and geographic expansions but with limited to no experience in those segments. This puts Amazon at vast risk of political and economic conditions. Tuning back to our machine learning model, using the BOW model, MNB and LDA we computed its sentiment to be at -0.66, implicating that the sentiment is negative, which aligns with the sentiment constructed using human analysis.

Ticker 3: GOOGL

As we focus on Google, we concluded that human sentiment is 0.2, which means it is more positive. As we read through Google's risk section, we can analyze that the risk Google faces are solely dependent on its operational capabilities. For example, a significant risk Google faces is building more innovative technology, but due to Google being a pioneer in innovation for over two decades, it has developed an efficient way of capturing more consumers. This allows Google to launch their products with an existing consumer base, such as adding a Google Hangout to capitalize on the increased online meetings. In this case, the machine learning models failed to identify which risks are more significant than others and can have a ripple effect on the company as it gave a sentiment of -0.77. A way to better fine-tune our model would be to train the algorithm on a much larger data set, which will allow it to predict more accurately.

Ticker 4: VLO

Looking at Valero Energy Corp, the human sentiment was concluded to be -0.8 due to its high dependency on factors such price of crude oil, corn, and other feedstocks, which they don't control. This exposes their business to many political, economic, and geographical risks, which can cause a decrease in the growth and financial health. Furthermore, due to increased focus on alternative energy, the demand for fossil fuels and GHG emissions has decreased, a significant risk for VLO. Our machine learning model predicts the sentiment accurately, computing a value of -0.81, concurring its negative sentiment.

Ticker 5: BA

Boeing is one of the companies that took a severe hit cause of COVID-19. We computed its sentiment to be -0.5 because of two factors. Firstly, their overall debt has increased due to the pandemic, putting them at liquidity risk. Secondly, due to the pandemic, the commercial airline

sector took a significant hit, affecting Boeing's growth as a company. Our machine learning model was able to classify its sentiment as negative and a value of -0.74, aligning it with our human sentiment. The Textblob Classification could not predict accurately as it gave a neutral score to the risk section, which can be explained as it had been pre-trained using different datasets.

Conclusion:

As we can see from our sentiment analysis, we can conclude that the Sentiment Bag of Word Model was the most accurate way to measure the sentiment due to the comprehensive dictionary made explicitly for financial statement analysis. Secondly, the Multinomial Naïve Bayes also predicted the sentiment accurately except while analyzing Google's sentiment leading to an accuracy score of 80%. Moreover, the LDA method had an accuracy of 60% which is not the best comparing to other models. Lastly, the TextBlob Classification could not predict the sentiment accordingly simply because it was pre-trained with a variety of data sets.

References:

X. *Man, T. Luo and J. Lin, "Financial Sentiment Analysis(FSA): A Survey," 2019 IEEE International Conference on Industrial Cyber Physical Systems (ICPS), 2019, pp. 617-622, doi: 10.1109/ICPHYS.2019.8780312.*

K. Mishev, A. Gjorgjevikj, I. Vodenska, L. T. Chitkushev and D. Trajanov, "Evaluation of Sentiment Analysis in Finance: From Lexicons to Transformers," in IEEE Access, vol. 8, pp. 131662-131682, 2020, doi: 10.1109/ACCESS.2020.3009626.

Aliteraturesurveyofactivemachinelearninginthecontext of ... *(n.d.). Retrieved April 22, 2022, Araci, D. (2019, August 27).*

Finbert: Financial sentiment analysis with pre-trained language models. *arXiv.org. Retrieved April 21, 2022*

Dixit, I. (2019, August 22). Python's Natural Language Tool Kit (NLTK) tutorial part - 1. *Medium. Retrieved April 21, 2022*

Ratz, A. V. (2022, April 8). Multinomial NAÏVE Bayes' for documents classification and Natural Language Processing (NLP). *Medium. Retrieved April 21, 2022*

Zhou, V. (2019, December 11). A simple explanation of the bag-of-words model. *Medium. Retrieved April 21, 2022*