Journal of Theoretical and Applied Information Technology
15th July 2024. Vol.102. No. 13

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

EVALUATING TEXTBLOB, LEXICON, SUPPORT VECTOR MACHINE, NAIVE BAYES, AND CHATGPT APPROACHES FOR SENTIMENT ANALYSIS OF NASDAQ LISTED COMPANIES

AZWARNI, NATHAR SHAH*
Faculty Of Computer And Informatics, Multimedia University Selangor, Malaysia

ABSTRACT

Sentiment analysis is a type of contextual text mining that finds and extracts subjective information from the source material in order to assist companies in understanding the social sentiment of their brand, product, or service while monitoring online conversations, especially Twitter has become a popular medium for individuals to express their opinions, share news, and discuss various topics, including stocks and companies. Stock market sentiment analysis is useful for understanding investor sentiments and forecasting market moves. Market players can use sentiment analysis tools to evaluate market sentiment and make educated investing decisions. The previous study examined data with fewer than ten thousand rows; however, this research will work with very huge data sets of more than one hundred thousand Nasdaq companies acquired from @Nasdaq and @AppleSupport Twitter accounts and @nasdaq and @apple from subreddit in Reddit social media. This study will compare the classification accuracy of Naive Bayes and SVM, as well as the time consumption of each strategy while classifying vast quantities of data. The TextBlob NLTK (Natural Language Toolkit) will be used in this study to label each phrase in the data using a lexicon-based method; also, this effort will employ ChatGPT, an OpenAI chatbot, to label each row of data received. As a consequence, it was discovered that SVM is the most superior approach in its classification, both in terms of Precision, Recall, and F1-Score metrics, as well as total accuracy, which reaches 93.5%, while Naive Bayes is at 61.5% and ChatGPT is at 42.2%.

Keywords – *Big Data, TextBlob, SVM (Support Vector Machine), Naïve Bayes, ChatGPT, Sentiment Analysis, Nasdaq*

1. INTRODUCTION

Sentiment analysis is a type of contextual text mining that finds and extracts subjective information from the source material in order to assist companies in understanding the social sentiment of their brand, product, or service while monitoring online conversations. The primary goal of sentiment analysis is to categorize text in a sentence or document and then decide whether the point of view expressed in the sentence or document is positive, negative, or neutral. According to Dergiades [1], sentiment research can yield profits for investors by assisting in decision making regarding the stock market.

The stock market is a dynamic and complicated system that is impacted by a variety of variables, such as economic indicators, news events, and investor sentiment. By analyzing sentiment in stock market-related documents such as news articles, social media posts, and financial reports, researchers and market participants can gain valuable insights into investor sentiment and make informed investment decisions. Social media platforms, especially Twitter [2], have become a popular medium for individuals to express their opinions, share news, and discuss various topics, including stocks and companies. The sheer volume and real-time nature of tweets make them a potentially valuable source of information for understanding market sentiment.

Stock market sentiment analysis plays a crucial role in understanding investor perceptions and predicting market movements. Sentiment analysis techniques, such as machine learning algorithms, offer a valuable means to extract sentimental information from textual data, enabling market participants to gauge market sentiment and make informed investment decisions. Previous articles dealt with data below ten thousand rows; however, this study will work with enormous data that is more than one hundred thousand. This research will compare the accuracy of Naive Bayes and SVM classification as well as the time consumption of each approach in classifying large amounts of data. This study will employ TextBlob NLTK (Natural Language Toolkit) [3] to label each sentence in the data using a lexicon-based technique; besides that, this work also employs ChatGPT, an OpenAI chatbot, to label each row of the acquired data [4].

The Naive Bayes algorithm [5] is based on Bayes' theorem and assumes the independence of features, making it computationally efficient and well-suited for sentiment analysis tasks. It leverages probabilistic principles to classify sentiment in stock market-related documents and a training dataset consisting of labeled documents to build a probabilistic model that associates features (words, phrases, or other textual elements) with sentiment classes (positive, negative, or neutral). This model then enables the classification of new, unlabeled documents into sentiment categories based on the likelihood of the occurrence of features within each class.

SVM [6] is a supervised learning algorithm that excels in binary classification tasks. It aims to find an optimal hyperplane that separates instances belonging to different classes with the maximum margin. In the context of sentiment analysis, SVM can be trained on a labeled dataset of stock market-related documents to learn a decision boundary that distinguishes between positive and negative sentiment. The algorithm's ability to handle high-dimensional feature spaces and its generalization properties make it suitable for sentiment classification in the stock market domain.

This study will examine which approach is the most accurate in predicting sentiment sentences and which technique is quicker when working with huge amounts of data in the context of stock market sentiment research.

Several methodologies are different between the prior work and the one we employ in this paper. Because certain lexicon techniques are particularly sensitive to lowercase and uppercase, the data cleaning procedure in [7] only conducts tokenization, eliminates stopwords, and also removes the Twitter symbol and does not include lowercase text. We transform all letters to lowercase first before deleting data noise, lemmatization, and tokenization. In [8], the data cleaning technique does not eliminate data noise, which affects the findings of the provided sentiment labels, while paper C does not go into depth on how data preprocessing is done. While the methodology for data cleaning in [9] is similar to that in this article, in [10] the removal of emojis and Twitter symbols is not performed, both of which are noisy data that cannot be read by the lexicon approach and will have an impact on the labeling conclusions. The previous articles used word tokenization, but in this study, we decided to utilize sentence tokenization since we wanted to know the sentiment of each sentence in Twitter comments about the Nasdaq companies list. In this study approach, we first convert all text to lowercase, then remove emojis, URLs, Twitter symbols such as @ (mention) and # (hashtag), any non-alphabetical and numeric characters, punctuation, and stopwords, and finally lemmatization and sentence tokenization.

The remainder of this work is structured as follows: Section 2 discusses related work on sentiment analysis in the stock market using Naive Bayes and SVM. Section 3 will briefly explain the datasets utilized for this article as well as the data preprocessing methods employed, followed by a detailed discussion of the sentiment analysis approach we developed for the purpose of this work as well as the ChatGPT data labeling. Section 4 will go into the classification and analysis of the results. Section 5 will summarize our results and provide options for further research.

2. RELATED WORK

This section will discuss a number of research articles that have similarities in the machine learning techniques employed, as well as some parallels in the methodology we apply in data preparation to categorize data as positive, negative, or neutral sentiments. In these studies, Naive Bayes and SVM are used in the classification process.

John Kordonis et al. [7] analyze the association between tweet sentiment and stock prices in their study, Stock Price Forecasting through Sentiment Analysis on Twitter. This article aims to forecast how the market will behave in the future by utilizing sentiment analysis on a series of tweets from the past few days, as well as test if the contrarian investing hypothesis holds true. Finally, John Kordonis identified a link between tweet emotion and stock prices.

John Kordonis et al. gather stock prices. API for Yahoo Finance This dataset includes the Open,

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Close, High, and Low values for each day, as well as sentiment from Twitter users obtained using the Twitter API. The authors employ the AFINN lexicon to identify data as positive, negative, or neutral and then proceed with the classification process using Naive Bayes and SVM, as well as N-grams in the featured extraction part. As a consequence, Naive Bayes yields significantly greater accuracy than SVM, precisely 0.80609 vs. 0.79308; then use SVM to examine the link between tweet sentiment and stock market prices, and then compare the results to the expected stock movements, achieving an average accuracy of 87% in proper stock movement prediction.

Meanwhile, Eugene F. Fama contends in Efficient Capital Markets: A Review of Theory and Empirical Work that stock prices do not change in line with sentiment research due to a lack of movement [11]. The Efficient Market Hypothesis (EMH) states that the stock price currently represents all relevant information in an efficient market. This implies that information is excellent from previous information, current information, and information from the firm itself, which is sometimes referred to as insider knowledge.

According to Tommy Wijaya Sagala et al.'s article [8], Technical analysis, together with news emotions, impacts stock prices. Technical analysis is a way of forecasting future stock movements by analyzing trade data such as average price and volume fluctuations. Technical analysts anticipate future stock price movements using price movement charts and a number of analytical methods.

As a result of evaluating two separate characteristics for stock price movement categorization, consisting of technical analysis features and online media sentiment features, Tommy Wijaya Sagala et al. predicted stock price movements utilizing a mix of technical analysis and sentiment analysis. According to experimental results, employing SVM to incorporate technical analysis features and online media sentiment labels on the ASII dataset resulted in the maximum accuracy. The accuracy achieved was 57.50%.

Nadika Sigit Sinatrya et al. [10] In their paper, Classification of Stock Price Movement With Sentiment Analysis and Commodity Price: Case paper of Metals and Mining Sector, employed Support Vector Machine (SVM), Nave Bayes, and K-Nearest Neighbour (KNN) algorithms. The price was then classified as "up," "down," or "constant" by the classifier. Data for this study was gathered from Google News and Yahoo! Finance. They used the gnews python package to extract 4200 news items from Google News, which were then manually labeled as positive, negative, or neutral. After cleaning and labeling the data, there are 3062 records left with 122 negative feelings, 238 good feelings, and 2702 neutral sensations. The next step is data preprocessing and TF-IDF was utilized in feature extraction before the classification phase. The results reveal that the Nave Bayes Algorithm achieves the best model with an accuracy of 60% in three days by merging copper price and sentiment analysis elements.

In the meantime, KaiSiang Chong and Nathar Shah researched [9] classifier model hyperparameters that are important for sentiment analysis and the models' optimization potential. The research was carried out using the Google Colab environment. A portion of the preprocessed data including 20.000 comments was used. There are 6219 positive comments, 6196 negative comments, and 7585 total comments. The Grid Search approach is utilized for hyperparameter tuning and can find both models' hyperparameters that are relevant for sentiment analysis. The findings suggest that alpha and fit_prior are significant hyperparameters for Naive Bayes, whereas C, kernel, and gamma are critical hyperparameters for SVM. After performing hyperparameter tuning, SVM outperformed Naive Bayes. The study demonstrates that hyperparameter tuning may improve model accuracy, and SVM has a higher potential for optimization than Naive Bayes.

All completed research studies only work with data sets of less than ten thousand lines, and the accuracy of Naive Bayes is higher than that of SVM on average.

3. METHODOLOGY

As part of this methodology, two supervised machine learning techniques, SVM and Naive Bayes, are compared in order to determine the performance of each technique in terms of accuracy, precision, recall, and F1 score, which indicates the effectiveness of each model in making accurate decisions, which can be used by investors to assess the performance of Nasdaq companies.

This methodology will cover data collection, cleaning, tokenization, and labeling before moving on to data processing with SVM and Naive Bayes.

Journal of Theoretical and Applied Information Technology
15th July 2024. Vol.102. No. 13

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The methodology workflow is visualized in Fig. 1, and the procedure will be detailed in full.

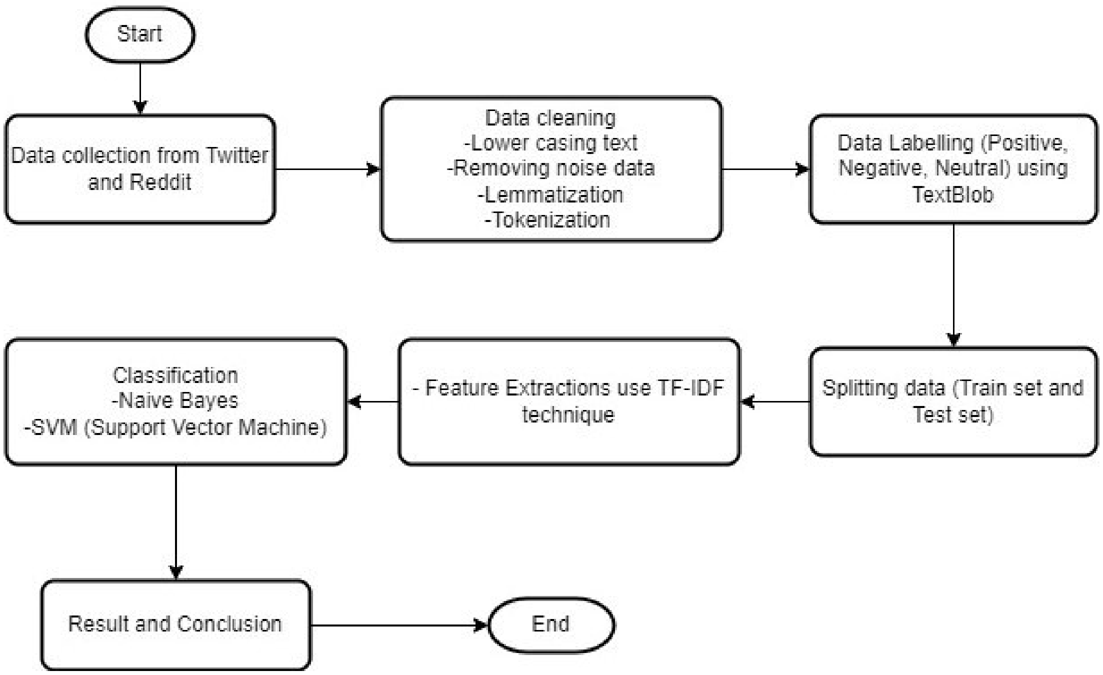

Figure 1: Methodology Workflow

3.1 Dataset

3.1.1 Data collection

As stated earlier, social media provides a platform for people to express their thoughts. As a result, the data used in this study was extracted from Twitter and Reddit, with the final data, after cleaning and several stages of data preprocessing, totaling around 100,000 statements.

- Data is collected on Twitter by scraping the comments of each tweet on the official Nasdaq Twitter account as well as the official Apple Twitter account using the Twitter API. According to Disfold.com, as of January 1, 2023 [12], Apple ranks first among other companies on the Nasdaq, so scraping for this data collection not only uses the @Nasdaq but also assists with the @AppleSupport and scraping for data collection also executes the time filter owing to Twitter's limits, so each data collection process only scraps comments for a one-year period, which is then aggregated into one and then continues at the next stage.
- The Reddit platform differs differently from Twitter in that it is more actively utilized in groups or subreddits to express the user's perspective; hence, The @nasdaq subreddit or group, which has about 5.4 thousand members and is supported by the @apple subreddit, with around 4.2 million members at the time this data was collected, makes it possible to access scraping rather than through the official Nasdaq or Apple accounts. Because each post reflects an individual perspective, this scrape includes not just the comments but also the text of the post itself.

3.1.2 Data cleaning

Each statement may contain terms that are neither meaningful nor useful for sentiment analysis. For example, some tweets contain URLs, tags for other users, or symbols that have no meaning. In order to better determine a tweet's sentiment score, preprocessing is required. Preprocessing is the transformation of unstructured data into structured data. This stage seeks to provide a clean dataset with better outcomes by doing the cleaning using the Natural Language ToolKit (NLTK) for Python. The preprocessing processes in this research are as follows:

- *Lowercase text:* All text in the data set is converted to lowercase to ensure a consistent format because this sentiment analysis processing is executed in the case-sensitive programming language Python.
- *Removal of Emojis:* Emojis now make up a considerable portion of text data. As a result of the widespread adoption of digital communication. This emoji must be removed from the data as long as the TextBlob lexicon is used to label it. Because TextBlob's sentiment analysis and other text processing procedures do not analyze the meaning or sentiment associated with the emojis,
- *URL removal:* URLs in comments that direct visitors to other websites are unrelated to the sentimental meaning of the content, may be spam, and will be filtered out completely.
- *Removing Twitter symbols:* Tweets frequently include extra symbols such as "@" or "#" as well as URLs. The term after the "@" symbol (mention) on Twitter is always a username, which is likewise excluded because it adds no value to the text during the sentiment analysis process. Only the "#" (hashtag) sign is deleted from the data; however, the words after the "#" (hashtag) are not filtered since they may provide significant data about the emotion of the tweet.
- *Removing special Character and digits:* The code used in this procedure searches for all non-word characters, i.e., non-alphabetical or numeric characters, which are subsequently eliminated since they can be deemed noise when entering the labelling process.
- *Taking out punctuation:* Taking out punctuation will help you treat each text equally. Punctuation is also deleted since it brings no value to sentiment analysis. This is also an important step for tokenization convenience.
- *Removing stopwords:* Stop words are functional terms that lack the sentiments that are commonly used. The Natural Language ToolKit Library in Python provides a dictionary of stopwords, which are lists of words with neutral meanings that are not suited for sentiment analysis, such as "from," "to," "or," "a," "of," "the," "I," "it," "you," and "and," and so on. Stopwords can be eliminated without affecting other words that frequently appear in texts.
- *Lemmatization:* A similar fundamental "stem or root" is what lemmatization attempts to achieve for a given word. When both terms are present in the data, the effort required by algorithms to interpret the sentiments of words increases, such as "evaluate" and "evaluation," where the root word for "evaluation" is "evaluate." Lemmatization of tokens to root types is therefore required to decrease processing time and comment complexity, thereby improving model performance.
- *Tokenization of Sentences:* This method reduced the comment to a single sentence. The labeling procedure will then be applied to a sentence rather than the comments, making it easier to determine whether the sentence is positive, negative, or neutral.

3.1.3 Data labelling

After cleaning the data, the next step in preprocessing is labeling the data. The TextBlob Lexicon is used to mark data. TextBlob is a Python package for text processing. It provides a basic API for looking deeper into standard natural language processing (NLP) activities. One of the NLP procedures includes a feature known as POS Tagging, or part of speech tagging. This approach assigns the given word its POS before proceeding with sentiment analysis labeling, which examines the text's polarity and subjectivity criteria to determine the sentiment of the text.

The TextBlob function returns the sentence's polarity and subjectivity. The polarity ranges from -1 to 1, with -1 representing negative sentiment and 1 expressing positive sentiment. The data has three labels at this point: 'POSITIVE,' 'NEGATIVE,' and 'NEUTRAL.' If the polarity value produced by a TextBlob polarity is larger than (>) 0, the sentence has a POSITIVE sentiment; if the polarity value is less than (<) 0, the sentence has a NEGATIVE sentiment; and if the resultant value is 0, the sentence has a NEUTRAL sentiment.

3.2 Sentiment Analysis

3.2.1 Machine learning

Classifier algorithms utilize advanced mathematical and statistical approaches to create predictions about the chance of the input data being categorized in a certain way. The classification techniques that will be employed in this research are Naive Bayes and

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

SVM (Support Vector Machine). Both are supervised machine learning techniques that are utilized for classification tasks such as text classification, which will be performed in this research.

The data is scraped from Twitter and Reddit, totaling roughly 100,000 sentences, and is then utilized as a train set and test set in the Naive Bayes and SVM classifications. Before classifying, we transform text data to numerical using the TF-IDF feature extraction, or Term Frequency-Inverse Document Frequency. It is a numerical statistic used to assess the significance of a word or phrase in a collection or corpus of documents.

Furthermore, the Naive Bayes and SVM classification training processes are carried out. In the Naive Bayes classification, we employ the Multinomial classification since this classification is commonly used in cases utilizing natural language processing [13], whereas the SVM classification takes longer to process data than the Naive Bayes classification. Because SVM has difficulty with big data, Naive Bayes takes only a few minutes to advance to the next level, which is the prediction stage, but SVM takes more than an hour. According to Mohammad Hassan Almaspoor et al.'s study, Support Vector Machines in Big Data Classification: A Systematic Literature Review [14], special processes will be needed to analyze big data.

The following metrics are obtained through this classification process:

1. *Accuracy*: Accuracy is the most fundamental measure of how accurate a model's predictions are. It is calculated as the percentage of correct predictions to the total number of predictions.
   Accuracy = (Number of Correct Predictions) / (Total Number of Predictions)
2. *Precision*: Precision is measured by how many of the model's positive predictions were accurate. Out of all instances predicted as positive, it quantifies the percentage of correctly predicted positive instances. When the cost of false positives is high, precision is useful.
   Precision = (True Positives) / (True Positives + False Positives)
3. *Recall* (Sensitivity or True Positive Rate): The model's capacity to correctly identify positive instances is measured by recall. It calculates the percentage of positive instances that were accurately predicted out of all positive instances. When the cost of false negatives is high, recall is crucial.
   Recall = (True Positives) / (True Positives + False Negatives)
4. *F1 Score*: A metric called the F1 score combines recall and precision into a single number. When the classes are unbalanced, it is especially helpful because it strikes a balance between precision and recall. The harmonic mean of recall and precision is used to calculate the F1 score.
   Skor F1 = 2 * ((Precision * Recall) / (Precision + Recall))

By considering various aspects of its predictions, these metrics assist in evaluating the effectiveness of a classification model.

3.2.2 ChatGPT

ChatGPT one of the chatbots developed by OpenAI, has made waves in natural language processing (NLP) since its release in 2019. ChatGPT can communicate natural or human responses in a very natural manner. The replies offered are sometimes accurate, as are the instructions written by the users. ChatGPT has acquired appeal among the general population because of its outstanding capacity to create natural-sounding cohesive prose. That is why this chatbot is currently so popular and constantly discussed on social media.

ChatGPT has garnered over 1 million users just one week after its inception, outpacing other prominent online platforms in terms of adoption rate [15] ; nevertheless, little consideration has been given to sentiment analysis of Twitter messages on ChatGPT [16]. This chatbot is part of the Large Language Model (LLM) [17], which is a computer program that can identify, summarize, translate, predict, and produce text. The model is powered by the GPT series of generative pre-trained transformers [18]. This paper will also use ChatGPT to categorize the data that has been collected before depending on the sentiment of the statement, whether positive, negative, or neutral.

This OpenAI chatbot is extremely helpful in some cases in saving time to gain answers, but it should be noted that this AI has some limitations, including a limited ability to comprehend, and the answers given are not always accurate [19]. As a result, a range of

Journal of Theoretical and Applied Information Technology
15th July 2024. Vol.102. No. 13

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

factors can influence the accuracy of the labels provided.

4. RESULT AND ANALYSIS

To collect data from Twitter and Reddit, this study uses the Jupyter Notebook environment. Our Reddit data collection tool uses the "praw" Python library, and we filter posts based on the year they were posted as well as the subreddits or groups they belong to. The year that we used is 2023 and the subreddit are @nasdaq and @apple. The filter used does cause data to be collected multiple times. However, as long as the data is scraped from the subreddit, then the post and the comments needed to be collected and stored, because both are represented the user sentiment. We use the Python "tweepy" library on Twitter, which uses Twitter's API. Just like Reddit does, Twitter also uses a year filter, with the years 2022 and 2023 being used, the accounts we use are the official Twitter accounts @AppleSupport and @Nasdaq and the obtained data is then compiled into a single CSV file.

The following stage is data preparation, which includes data cleaning and labeling as described in the dataset section. We utilize the Python libraries "re," "emoji," and "nltk" for data cleansing. The initial step is to convert all text to lowercase, followed by noise removal, lemmatization, and sentence tokenization. We employ the TextBlob lexicon in data labeling by leveraging the Python "Textblob" library and also "nltk" to aid in text data processing. This TextBlob has a polarity value ranging from -1 to 1, therefore in each sentence, TexBlob assigns a value to each word, which is then summed together to yield the final polarity result of the sentence. Based on the polarity of the sentence, we labeled polarity less than zero (0) as "Negative (-1)," polarity equal to zero (0) as "Neutral (0)," and polarity more than zero (0) as "Positive (1)." We were able to obtain 119980 rows of data from Twitter, for a total of 6495, with the remaining 113485 rows coming from Reddit. TexBlob labeled data from Twitter with a positive value of 3757, 555 rows of data were negative, and 2182 data were neutral (Fig.2), whereas data from Reddit had a positive value of 52141, 21715 data rows were negative, and 39630 were neutral (Fig.3). Even on Twitter and Reddit, the social sentiment with the highest score is still positive, indicating that most of these social media users have favorable views of the Nasdaq company, especially Apple products. Furthermore, the data labeled using TexBlob is saved in a CSV file, and a classification procedure is carried out utilizing Naive Bayes and SVM classification.

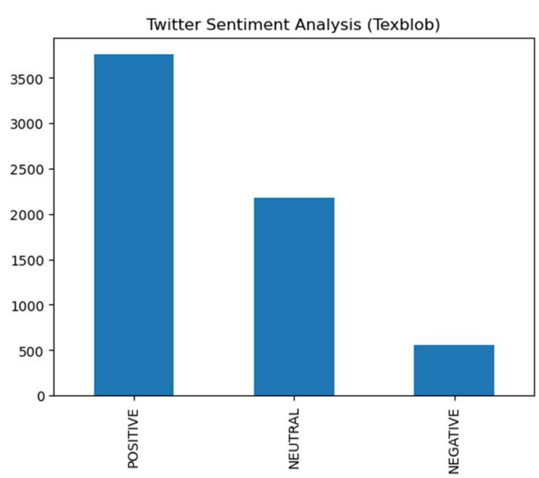

Figure 2. Twitter TexBlob lexicon labeled data

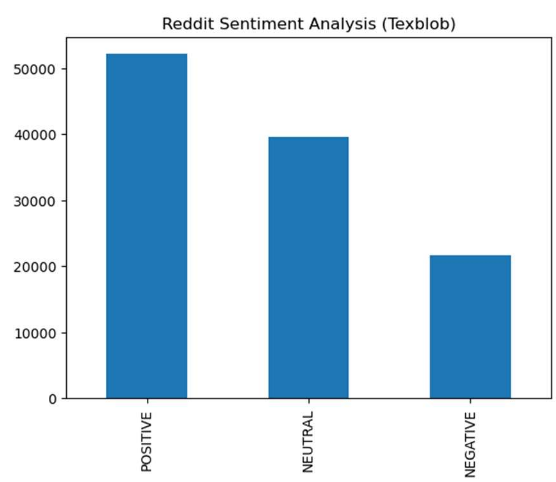

Figure 3. Reddit TexBlob lexicon labeled data

SVM and Naive Bayes are used to classify sentiment-labeled data, both of which make use of the Python module "sklearn." The data is split into two sections, text comments as X and data labels as Y, and then divided into a training set and a testing set using "train_test_split." Data must be transformed into numerical features using TF-IDF before proceeding with classification training. In this phase, the SVM classifier encounters an issue, which is, it takes more than an hour to train SVM classification. The results show that SVM has greater accuracy than Naive Bayes in processing huge stock

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

sentiment data, reaching 93.5%, whereas Naive Bayes only gets an accuracy rate of 61.4% [20] [21] [22]. Next, we run the classification process with lesser amounts of data to check if SVM retains its high accuracy while working with larger amounts of data, or if Naive Bayes, which can be superior to SVM, may be used instead. We use SVM and Naive Bayes to categorize the data obtained from Twitter, a total of 6495 rows of data. As a consequence, the accuracy of Naive Bayes reaches 76.2%, indicating that it has improved while working with smaller data sets, although SVM still exceeds Naive Bayes with an accuracy rate of 86.9%, and SVM has no problem with the time spent during classification training.

Machine Learning Evaluation Metrics		
Algorithm	Naïve Bayes	SVM
Accuracy	0.614	0.935
Precision	Pos: 0.55	Pos: 0.94
	Neg: 0.96	Neg: 0.93
	Neut: 0.96	Neut: 0.93
Recall	Pos: 0.99	Pos: 0.96
	Neg: 0.12	Neg: 0.80
	Neut: 0.36	Neut: 0.98
F1-Score	Pos: 0.71	Pos: 0.95
	Neg: 0.21	Neg: 0.86
	Neut: 0.53	Neut: 0.95

Figure 4: SVM and Naive Bayes result

Based As demonstrated in the Naive Bayes and SVM results table (Fig.4), SVM outperforms Naive Bayes in terms of overall accuracy, meaning that the SVM model delivers more trustworthy predictions in general. SVM values predict everything with 93% precision, whether positive, negative, or neutral. However, in Recall metric a positive value of Naive Bayes outperforms SVM by 3%, with SVM having a recall rate of 93% and Naive Bayes having a recall rate of 61% overall. A high F1 score implies a good balance of accuracy and recall, with SVM outperforming Nave Bayes by 93%. In terms of total and per-metric performance, whether working with big data or smaller data, SVM exceeds Naive Bayes. Although SVM takes longer to train models, the outcomes are more accurate.

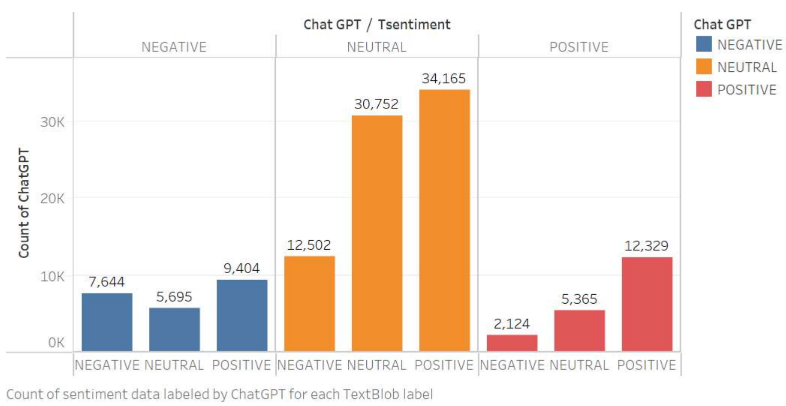

Figure 5: ChatGPT result belongs to TextBlob label.

We employ a different process for data labeling with ChatGPT, and we handle everything by hand. First, using the collected data, we copy every two hundred lines of text (due to ChatGPT's limits) to the ChatGPT website and ask ChatGPT to identify them one by one. This procedure took a long time; we spent around a week finishing the 19980 rows of data we had. We determined how much data is the same as the sentiment label with the data that was previously tagged with Textblob using the labels obtained from the GPT conversation. The findings are displayed in Fig.4. There are 7644 negative values accurately categorized by ChatGPT, 30752 neutral values, and 12329 positive values. The accuracy is then calculated manually using the formula,

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Accuracy = (Number of Correct Predictions) / (Total Number of Predictions)

Yielding a result of 42.2%, indicating that ChatGPT has the lowest accuracy value when compared to Naive Bayes and SVM.

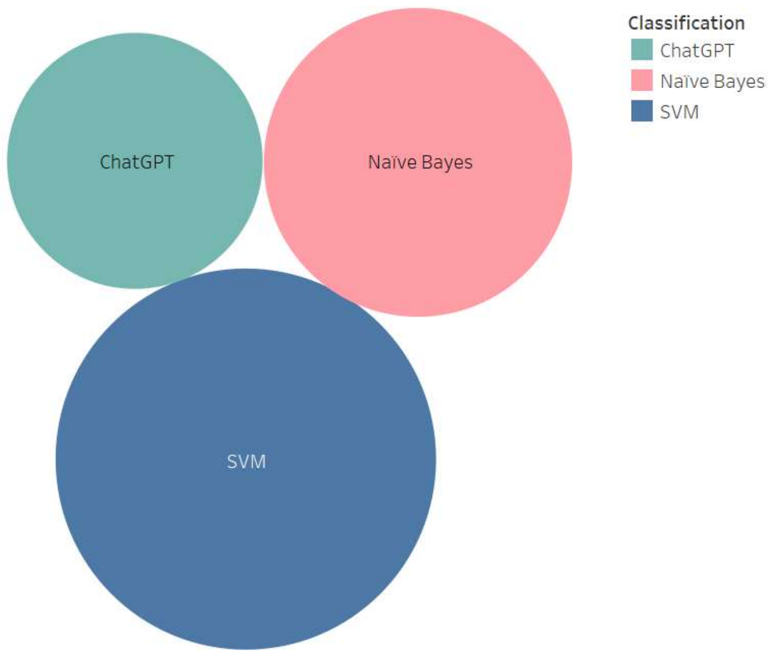

Figure 6: Accuracy Of Each Classification

5. CONCLUSION AND FUTURE WORK

This study investigates which strategy is the most accurate in predicting sentiment phrases and which technique works faster with big data in the context of stock market sentiment research, because the data used is massive, exceeding 100,000 lines. The data is derived through scraping Twitter comments on the official Twitter accounts @Nasdaq and @AppleSupport, as well as posts and comments on subreddit @nasdaq and @apple on the social media platform Reddit. After cleaning the data with lowercase text and eliminating noisy data such as stopwords, lemmatization, and sentence tokenization, the data is labeled using the TextBlob lexicon, yielding 55898 data with positive emotions, 22270 data with negative emotions, and 41812 data with neutral sentiments. The data is then separated into training and testing sets, and feature extraction with TF-IDF is used to transform text data into numerical data.

In the classification process, we utilize supervised machine learning techniques such as Naive Bayes and SVM to calculate accuracy, precision, recall, and F1-score to determine which approach is more accurate. As a consequence, SVM surpassed Naive Bayes in both overall and per-metric accuracy, with each measure achieving an average rate of 0.93%. We also use ChatGPT to manually categorize each row of data as positive, negative, or neutral, and the results are 7644 for true Negative, 12239 for true Positive, 30752 for true Neutral, and the overall accuracy is 0.422%. As a consequence, when compared to Naive Bayes and ChatGPT, SVM has the greatest overall accuracy value. In terms of time-consuming work with huge data, Naive Bayes has the shortest training time, which just takes a few minutes, while SVM takes more than an hour, and ChatGPT is the most time-consuming since everything has to be done manually, notably data verification, which took almost a week.

The limitation of this study, particularly the SVM classification, which takes longer when working with large amounts of data, is that we expect that future work will resolve this problem.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

REFERENCES

[1] T. Dergiade, "Do investors' sentiment dynamics affect stock returns? Evidence from the US economy," *ScienceDirect,* vol. 116, no. 3, pp. 404-407, September 2012.

[2] P. Gonçalves, M. Araújo, F. Benevenuto dan M. Cha, "Comparing and combining sentiment analysis methods," *ACM Digital Library,* pp. 27-38, october 2013.

[3] E. Talahaturuson, A. B. Gumelar, A. G. Agustinus, S. Suprihatien, H. A. Altway, C. S. N. Lestari, S. Sueb, P. Darmajanti, T. Hariyanti, S. Nilowardono, U. Z. Fanani dan S. Sulistiyani, "Exploring Indonesian Netizen's Emotional Behavior Through Investment Sentiment Analysis Using TextBlob-NLTK (Natural Language Toolkit)," *IEEE,* 17-18 September 2022.

[4] M. Belal, J. She dan S. Wong, "Leveraging ChatGPT As Text Annotation Tool For Sentiment Analysis," *Cornell University,* 18 June 2023.

[5] B. Pang dan L. Lee, "Opinion Mining and Sentiment Analysis," *ACM Digital Library,* vol. 2, no. 1-2, pp. 1-135, 01 January 2008.

[6] T. Joachims, "Text categorization with Support Vector Machines: learning with many relevant features," *ACM Digital Library,* pp. 137-142, 21 April 1998.

[7] J. Kordonis, S. Symeonidis dan A. Arampatzis, "Stock Price Forecasting via Sentiment Analysis on Twitter," *ACM digital library,* pp. 1-6, 10 November 2016.

[8] T. W. Sagala, R. Mahendra, I. Budi dan M. S. Saputri, "Stock Price Movement Prediction Using Technical Analysis and Sentiment Analysis," *ACM Digital Library,* pp. 123-127, January 2020.

[9] K. Chong dan N. Shah, "Comparison of Naive Bayes and SVM Classification in Grid-Search Hyperparameter Tuned and Non-Hyperparameter Tuned Healthcare Stock Market Sentiment Analysis," *International Journal of Advanced Computer Science and Applications(IJACSA),* vol. 13, no. 12, 2022.

[10] N. S. Sinatrya, B. A. Santoso dan I. Budi, "Classification of Stock Price Movement With," *IEEE,* 31 October 2022.

[11] E. F. Fama, "Efficient Capital Markets: A Review of Theory and Empirical Work," vol. 25, no. 2, pp. 383-417, Mei 1970.

[12] "Top NASDAQ Composite Companies Ranking by Market Cap as of Jan. 1, 2023," dalam *DISFOLD*, 2023.

[13] M. Abbas, K. Ali, S. Memon dan A. Jamali, "Multinomial Naive Bayes Classification Model for Sentiment," vol. 19, March 2019.

[14] M. H. Almaspoor, A. Salajegheh, A. Safaei dan B. Minaei, "Support Vector Machines in Big Data Classication:," June 2021.

[15] J. Sier, "ChatGPT takes the internet by storm, bad poetry and all," 2022.

[16] M. U. Haque, I. Dharmadasa, Z. T. Sworna, R. N. Rajapakse dan H. Ahmad, ""I think this is the most disruptive technology": Exploring Sentiments of ChatGPT Early Adopters using Twitter Data," 12 December 2022.

[17] D. A. Deshpande, "Optimizing Language Models for Dialogue," dalam *Digital Technology in Clinical*, D. S. V. Kulkarni, D. S. V. Kulkarni dan D. A. Modi, Penyunt., Mumbai, Indian College of Physicians, 2022, p. 62.

[18] A. Radford, K. Narasimhan, T. Salimans dan I. Sutskever, "Improving Language Understanding," 2018.

[19] A. Radford, J. Wu, R. Child, D. Luan, D. A. Amodei dan I. Sutskever, "Language Models are Unsupervised Multitask Learners," *Semantic Scholar,* 2019.

[20] S. Fahmi, L. Purnamawati, G. F. Shidik, M. Muljono dan A. Z. Fanani, "Sentiment Analysis of Student Review in Learning Management System Based on Sastrawi Stemmer and SVM-PSO," *IEEE,* 26 October 2020.

[21] S. Rana dan A. Singh, "Comparative analysis of sentiment orientation using SVM and Naive Bayes techniques," *IEEE,* 16 March 2017.

[22] G. Novilia, R. Adriman dan T. F. Abidin, "Comparison of Classification Methods Using Feature Selection for Smartphone Sentiment Analysis," *IEEE,* 06 February 2023.