Survey of 𝕏 Platform Sentiment Analysis Methods

Sana Shamma
Software Engineering Deptment
Uiversity of Prince Mugrin
Medina, Saudi Arabia
sanashamma3@gmail.com

Salwa Shamma
Software Engineering Deptment
Uiversity of Prince Mugrin
Medina, Saudi Arabia
salwaramezshamma@gmail.com

Samah Shamma
Software Engineering Deptment
Uiversity of Prince Mugrin
Medina, Saudi Arabia
Samahr.shamma@gmail.com

Marwa Alrehili
Computer Science Department
Uiversity of Prince Mugrin
Medina, Saudi Arabia
m.alrehili@upm.edu.sa

***Abstract*—Nowadays, social media is an essential part of our daily lives, and this fact highlights the importance of considering them as a good source of data when we want to understand society's opinions, thoughts, and insights about a certain topic. To elaborate more, one way for companies to discover customer satisfaction with their products is by monitoring the customer emotions on their social media accounts. In general, text mining is a high-demand and challenging domain that requires technical knowledge and up-to-date technology. For this reason, this paper studies how to automatically understand the emotions of social media users on 𝕏 platform, which is a microblogging service that welcomes sharing opinions and ideas. There has been much research illustrating many methods to perform sentiment analysis 𝕏 platform by categorizing the opinions as positive, negative, or neutral, and this research compares seven well-known methods in this field which are Support Vector Machines (SVM), Random Forest, K-Nearest Neighbors (KNN), Valence Aware Dictionary and sentiment Reasoner (VADER), Naive Bayes, Adaboost, and Maximum Entropy using a dataset called "Twitter Tweets Sentiment Dataset" from Kaggle. At the end of the paper, it states that among the previous seven algorithms, Maximum Entropy (ME) achieved the highest results with an accuracy of 81%, so it is considered a good candidate for performing sentiment analysis in future tasks on 𝕏 platform with conducted further work to enhance the results more.**

Keywords— Data Mining, Sentiment, Sentiment Analysis, Machine Learning, Twitter dataset, 𝕏 Platform (Twitter), TF-IDE, SVM, Random Forest, KNN, VADER, Naive Bayes, Adaboost, Maximum Entropy.

I. Introduction

Social media has become an integrated component of everyday life, performing a variety of functions such as advertising, sharing social opinions and financial trends, gathering user feedback on products, and broadcasting news. Through postings, comments, messages, and likes, it offers users a virtual community where they can interact, share their thoughts, and build relationships. Presently, numerous social media platforms operate worldwide, with three major players being Facebook, LinkedIn, and 𝕏 [1].

𝕏, formerly known as Twitter, has emerged as a significant player in social media. It provides an opportunity for millions to express their daily activities and thoughts in real-time via brief messages. Researchers in a variety of subjects can find a hidden gold mine in this never-ending supply of knowledge created by users.

By analyzing the massive volume of data on 𝕏, researchers can gain insights into a wide range of human behaviors. From predicting fluctuations in stock markets to analyzing public opinion on societal issues, 𝕏's data enables us to gauge the collective mood and social trends.

The 𝕏 platform allows users to express themselves not just via words but also through images and videos, facilitating exciting and constantly changing exchanges. This rich material further motivates sentiment analysis, a method that examines the emotions expressed in user postings. Through the examination of these emotions, scientists can gain insight into how the public views certain subjects.

Sentiment analysis techniques can be utilized to recognize emotions in 𝕏's posts. It assesses whether a text conveys sentiments that are positive or negative or neutral. To identify emotions in text, sentiment analysis methods often rely on the existence of English opinion lexicons and emotion-evoking words (like, hate, and love). In this study, we offer an initial analysis that attempts to identify and analyses emotions in postings. Our analysis utilizes a dataset comprising 27,226 posts sampled from 𝕏 feeds. Our objectives are twofold: first, to identify the most effective techniques for detecting emotions and collective mood, and second, to investigate and visualize the underlying patterns in emotional expression.

The remainder of this paper is organized as follows: Section two provides a comprehensive review of related work in the field of sentiment analysis. Section three offers a brief description of the dataset and data preprocessing. Section four outlines the methodology employed for evaluating the aforementioned methods. Section five presents and analyzes the experimental results. Finally, Sections six and seven conclude the paper, highlighting key takeaways and avenues for future research.

II. Related Work

Feature extraction is a significant step in sentiment analysis problems because it determines how much our algorithm can classify people’s opinions correctly. Due to its clarity and effectiveness, the TF-IDF models are often used for many natural language techniques. The TF-IDF algorithm aids sentiment analysis by analyzing sentiment text corpora. This approach follows a mathematical-statistical approach, which determines the significance of a word to the corpus in a text document [2]. TF-IDF approach obtains the most influential features based on the word frequency of each document from text data. TF-IDF is the combination of two sections, namely, TF and IDF. The values of TF and IDF are calculated using formula number (1) and (2). After we have extracted the important features, we use the textblob function from the natural language toolkit to get the polarity of each tweet, and we set a 1 for positive tweets, -1 for negative tweets, and 0 for neutral tweets.

Kumar etal. (2021) presented a paper titled "Performance Analysis of Sentiments in Twitter Dataset Using SVM Models." The authors compared the performance of SVM radial kernel, SVM linear grid, and SVM radial grid and found that SVM linear grid outperformed the other SVM models. Additionally, the proposed framework was illustrated in Fig.1. The process began with data collection, where 1000 tweets were selected for analysis and divided into a 70% training set and a 30% test set, with the classes being neutral, negative, and positive. Subsequently, pre-processing was conducted, which involved removing stop words, unnecessary white spaces, and other irrelevant elements. Next, Boruta, a random forest-based method, was employed for feature selection. Finally, SVM models were applied, and the accuracy was assessed [3].

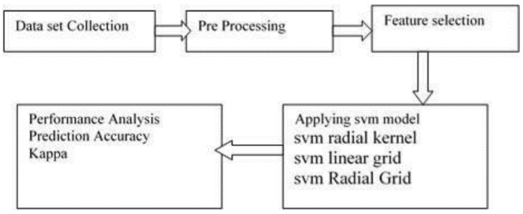

Fig. 1. Proposed framework.

"Sentiment Analysis and Classification of Indian Farmers' Protest Using Twitter Data" is the title of a work written by Sanjay et al. (2021). The study looks at several models that were applied to classify and examine the views extracted from a dataset of about 20,000 tweets related to the farmers' protest. By using both Bag of Words and TF-IDF methodologies in their investigation, they discovered that Bag of Words performed better than TF-IDF. The researchers also used Support Vector Machines, Random Forests, Decision Trees, and Naive Bayes, and they finally found that Random Forests had the best accuracy in sentiment classification [4].

Cindo and colleagues (2020) delved into analyzing news articles in Kazakhstan. They noticed a bias towards positive sentiments in the data and used methods like undersampling, oversampling, and SMOTE to balance it. By converting text into numerical vectors and applying various machine learning models, they found that oversampling and SMOTE, particularly with logistic regression, decision trees, and KNN, yielded strong classification results. This underscores the value of employing diverse ML techniques in sentiment analysis [1].

The paper "Unveiling Twitter Sentiments: Analyzing Emotions and Opinions through Sentiment Analysis on Twitter Dataset" written by Ferdoshi et al. (2023). They highlight the importance of automatically classifying 𝕏 platform emotions into positive, negative, and neutral for both the business and research worlds. In their research, they compare two machine learning algorithms: VADER and Transformers-RoBERTa, to detect different sentiments of tweets. They found that VADER achieved higher results than Transformers-RoBERTa on data from a website called "Mendeley". Despite this, they clarify that if the purpose is to detect sentiment for nuances of text and languages, RoBERTa can perform better than VADER as it is considered an advanced deep learning model [5].

In their paper "Comparison of Naive Bayes and SVM Algorithm based on Sentiment Analysis Using Review Dataset," Rahat et al (2019) analyze the sentiment on 𝕏 platform for 10,000 posts containing passenger reviews, whether negative, positive, or neutral, for an airline company to determine the quality of the services. They use two supervised machine learning algorithms in their paper: Support Vector Machine (SVM) and Naive Bayes (NB), and they found that both algorithms yielded good results in the case of airline reviews on 𝕏 platform posts. Therefore, they both are considered good choices for performing sentiment analysis on 𝕏 platform posts. Finally, in terms of high accuracy, SVM achieved higher results than Naive Bayes [6].

“Sentiment Analysis on Twitter Data: Comparative Study on Different Approaches” authored by Abdur Rahman et al. (2021). They investigate sentiment analysis on Twitter, a prominent social media platform. Utilizing Support Vector Machines (SVM) and Adaboost with different kernel functions and feature extraction techniques like Chi-square and CPD, the research aims to optimize sentiment classification models [7].

Cindo et al (2020) highlight the exponential growth of social media has provided abundant data for research in social mining, with Twitter emerging as a key platform for sentiment analysis. While previous studies have made strides, performance optimization remains a challenge. While Support Vector Machine (SVM) has been popular, recent research suggests Maximum Entropy may offer superior efficiency. Their study compares SVM and Maximum Entropy across two datasets—general opinions and airline-specific sentiments—utilizing four feature extraction techniques. Maximum Entropy outperforms SVM, achieving 85.8% accuracy on general opinions and 92.6% on airline sentiments, highlighting the importance of diverse classification methodologies in sentiment analysis on Twitter [1].

III. DATASET AND DATA PREPROCESSING

A. Dataset

In this paper, we used the "Twitter Tweets Sentiment Dataset" from the Kaggle platform, which consists of Twitter tweets with different sentiment characteristics. The dataset includes 27,226 tweets. The data is presented in raw form with four different features. The first feature, called "textID," is a unique ID for each piece of text; the second feature, called "text," is the text of the tweet; the third feature, "selected_text," is a subset of the text that contains keywords from the tweet; the fourth feature, called "sentiment," represents the general sentiment of the tweet, which can be either positive, negative, or neutral.

B. Data Preprocessing

As it is known, performing data preprocessing on a dataset to clean it or to make it consistent helps to get better results when algorithms are applied to it. For that, many preprocessing steps are applied, such as checking if there are null values to handle them by removing them and converting the text to lowercase to get a uniform form of data. In addition, remove garbage words in the context of sentiment analysis, like URLs, special characters, and stopping words, and so on to avoid adversely affecting the results. Also, we convert the text to stemming (the base form of the word) thus helping in standardizing the data. After all that, we convert the text form to an encoding form to be ready to handle it by all different algorithms. Lastly, we split the data into two different groups, one for train data and one for test data.

IV. Methodology

In this section endeavors to evaluate the effectiveness of detecting emotions in tweets by employing a diverse array of methods. The evaluation encompasses a comparison of eight distinct techniques, each offering a unique approach to discerning emotions. These techniques encompass analyzing tweets based on sentiment polarity, Support Vector Machines (SVM), Random Forest, K-Nearest Neighbors (KNN), Valence Aware Dictionary and sentiment Reasoner (VADER), Naive Bayes (NB), Term Frequency-Inverse Document Frequency (TF-IDF), Adaboost, and Maximum Entropy (ME). Through the evaluation and contrast of these methodologies, the study seeks to glean insights into their respective strengths and weaknesses, thereby advancing the field of emotion detection in tweets.

Our methodology unfolds through several meticulous steps aimed at ensuring the accuracy and reliability of our findings. Initially, we curated a dataset comprising 27,226 tweets utilizing the Python programming language. The tweets were sourced from the Kaggle website, providing a diverse and representative sample for analysis. Subsequently, to predict the emotion conveyed in each tweet, we introduced an additional column presenting sentiment numerically. Here, a value of 0 denotes negative sentiment, 1 denotes neutral sentiment, and 2 denotes positive sentiment.

A. Frequency-Inverse Document Frequency (TF-IDF)

In sentiment analysis, feature extraction plays a crucial role as it directly impacts the accuracy of categorizing individual thoughts. Among various methods employed in natural language processing, TF-IDF models stand out for their clarity and effectiveness. TF-IDF, which stands for Term Frequency-Inverse Document Frequency, is commonly utilized in sentiment analysis tasks. This method evaluates the relevance of words within a text document based on a mathematical-statistical approach. By analyzing word frequency across documents, TF-IDF identifies the most significant features for sentiment analysis [2].

TF-IDF comprises two components: TF (Term Frequency) and IDF (Inverse Document Frequency). These values are computed using the following formulas (1) and (2):

$$TF = \frac{Number\ of\ times\ terms\ appear\ in\ a\ document}{Number\ of\ terms\ in\ a\ document} \quad (1)$$

$$IDF = \frac{Total\ number\ of\ documents}{Total\ number\ of\ documents\ each\ term\ appears} \quad (2)$$

B. Support Vector Machines (SVM)

Support Vector Machine (SVM) is a tool that helps us draw lines to separate different groups of data. We use something called a "kernel" to do some math on our inputs. With a linear kernel, we have a simple formula:

$$f(x) = \ \beta(0) + sum\ (a_i * (x, x_i)) \qquad (3)$$

For linear kernel calculation, when we get a new input, we calculate its position compared to all the training data. Then, we add up some numbers ($\beta(0)$β(0) and aiai) we figured out from the training data to make a prediction [1].

C. Random Forest (RF)

Random Forest (RF) is a powerful ensemble machine learning algorithm that combines multiple decision trees to make accurate predictions in classification tasks. It works by training individual trees on random subsets of both the data and features, which helps to reduce overfitting and improves the model's ability to generalize to new data. During prediction, each tree in the forest contributes its output through majority voting. RF is robust, capable of handling various data types and noisy data, making it a popular choice for real-world machine learning problems. Additionally, RF provides insights into feature importance, aiding in feature selection and understanding the most influential variables in the dataset [2].

D. K-Nearest Neighbors (KNN)

A well-known non-parametric approach for categorizing data instances is called k-nn. The points are assigned to the class of their k nearest neighbor points once the distances between the vectors are calculated. In general, this method categorizes texts using the most used distance measurement, known as the Euclidean distance, which is defined as:

$$d\ (x, y) = \sqrt{\sum_{i=1}^{N} (a_{ix} - a_{iy})^2} \qquad (4)$$

Here, d (x, y) represents the distance between two data instances, aix and aiy are the weights of the ith terms in the data instances x and y respectively, and N is the total number of unique features in a set of data instances [8].

E. Vader

The VADER tool is pacifically designed for analyzing sentiment in social media text, particularly tweets. It evaluates each tweet and assigns scores indicating positive, negative, neutral, and compound sentiment. The output dataset containing these sentiment scores offers valuable insights into the sentiment polarity of each tweet, enhancing subsequent analysis and visualization efforts. Utilizing VADER enables a more profound comprehension of sentiment patterns and user behavior within our dataset [9].

F. Naive Bayes (NB)

Based on the Bayes theorem, an NB classifier is a probabilistic machine learning model used for classification tasks. It is simple to use and has encouraging outcomes. The conditional probability that a document d belongs to a class c serves as the basis for the classifier. At the heart of the algorithm is the Bayes formula. The classifier's formula for sentiment analysis is provided by [8]:

$$P\ (c|d) = \frac{P(c) \times P\ (d|c)}{P\ (d)} \qquad (5)$$

G. Adaboost

For text categorization, boosting is a highly effective machine learning technique. In order to create a powerful classifier and achieve high accuracy, many performing classifiers are merged. The Adaboost method iteratively ensures accurate predictions of uncommon observations by setting the classifier weight and training the data sample. A basic learning classifier based on decision trees is utilised. A single decision tree with a predetermined weight dependent on the decision tree's correctness was discovered in a single cycle. Ultimately, the following equation combines all of these hypothesis' decisions:

$$f(x) = \sum_{i=1}^{T} a_t h_t(x) \qquad (6)$$

Here, T is the number of hypotheses; t is the index of the hypothesis ht with weight at. Then, based on the sign of f(x) the sample is classified into a class [10].

H. Maximum Entropy

In Maximum Entropy classification, no assumptions are made regarding the relationships between features. The primary goal is to maximize entropy within the system by predicting the conditional distribution of labels for each class. Unlike other methods, such as logistic regression, this classification method can effectively handle overlapping features. The resulting distribution, known as MaxEnt, is characterized by its lack of assumptions regarding features. MaxEnt can be expressed as:

$$P_{ME}(c|d,\lambda) = \frac{\exp[\sum_i \lambda_i f_i(c,d)]}{\sum_c \exp[\sum_i \lambda_i f_i(c,d)]} \qquad (7)$$

Here, c represents a class, b denotes a feature (e.g., Twitter), and λ signifies the weight vector [1].

V. Experiments & Results

In this section, we aim to evaluate the effectiveness of detecting emotions in tweets by employing a diverse array of methods. These techniques encompass analyzing tweets using various methods, including Support Vector Machines (SVM), Random Forest, K-Nearest Neighbors (KNN), Valence Aware Dictionary and Sentiment Reasoner (VADER), Naive Bayes, Term Frequency-Inverse Document Frequency (TF-IDF), Adaboost, and Maximum Entropy.

Fig.2 shows the accuracy of these various ML algorithms during sentiment analysis of tweets. Vader exhibited the lowest accuracy at 56%, while Maximum Entropy achieved the highest accuracy of 81%. SVM and Random Forest secured the second position with accuracies of 80.2% and 80%, respectively.

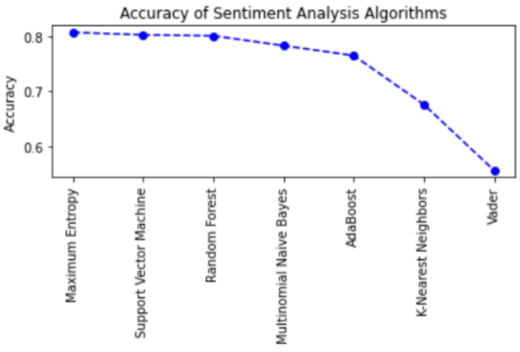

Fig. 2. Accuracy of Sentiment Analysis Algorithms.

The word cloud was generated from the classified tweets, providing a visual representation of the commonly used words in the dataset. By analyzing the entire word cloud, we can gain insights into the most frequently used words. The size of each word in the cloud indicates its frequency, with larger fonts representing more commonly used words. [4]

In the positive tweets category, words such as "love," "thank," "good," "great," and "happy" dominate, indicating associations with positive emotions and experiences as shown in Fig.3.

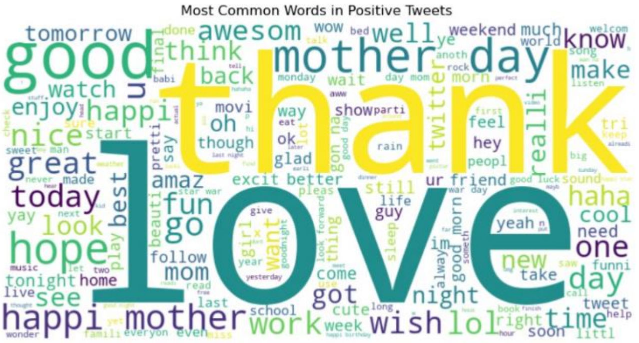

Fig. 3.Most common words in positive tweets.

On the other hand, the negative word cloud contains scattered words like "miss," "sad," "bore," and "ugh," reflecting dissatisfaction or unpleasant states expressed in the tweets as shown in Fig.4.

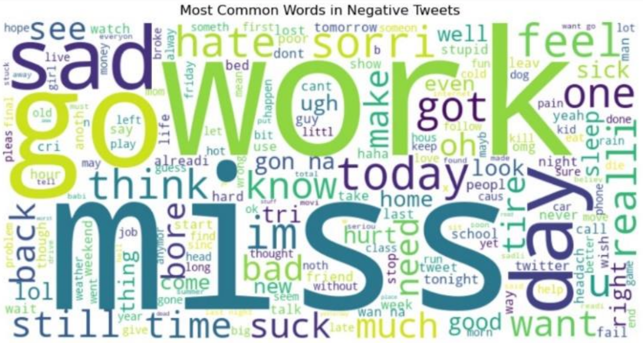

Fig. 4. Most common words in negative tweets.

In the neutral tweets category, a mixture of less emotionally charged terms is observed, such as "work," "today," "going," and "watch." This category encompasses a variety of subjects without a strong emotional inclination as shown in Fig.5.

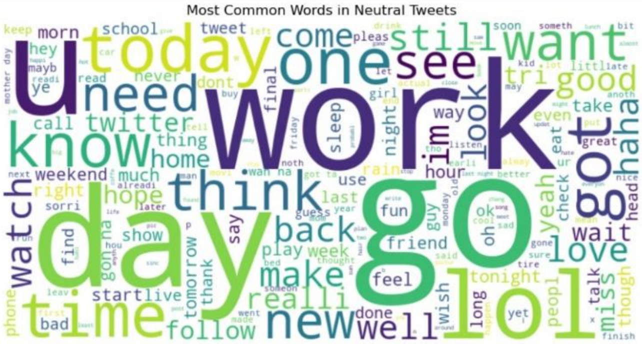

Fig. 5. Most common words in neutral tweets.

Fig. 6 shows confusion matrix representing the performance of Maximum Entropy (ME) using TF-IDF vectorizer that has the highest accuracy. The confusion matrix shows the relationship between the actual sentiment of tweets (true label) and the predicted sentiment of tweets (predicted label).

According to the confusion matrix, a percentage of tweets were predicted with the same sentiment as their true label is 81%. This is calculated by adding up the values in the matrix: 1173+ 1835+1385 = 4393. On the other hand, a percentage of tweets were predicted with a wrong sentiment is 19%. This is calculated by adding up the values in the matrix corresponding to the incorrect predictions: 320+ 78+ 171+ 122+ 116+ 246= 1053. Based on these results, we can conclude that approximately 81% of the tweets were predicted correctly.

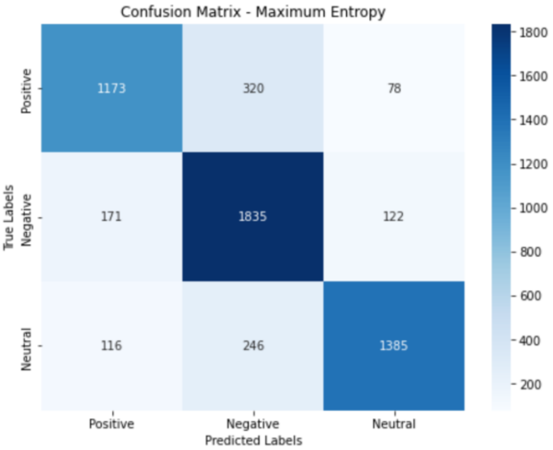

Fig. 6. Confusion matrix for maximum entropy.

In Fig.7, we can see the confusion matrix representing the performance of Multinomial Naive Bayes (MNB) using TF-IDF vectorizer. This model achieved the highest accuracy in detecting negative sentiment tweets, which accounted for approximately 44% out of tis accuracy. To provide further details, the model successfully identified around 18% of positive tweets, 35% of negative tweets, and 25% of neutral tweets. Dividing 35 by 78 yields a percentage of 44%.

According to the confusion matrix, a percentage of tweets were predicted with the same sentiment as their true label is 78%. This is calculated by adding up the values in the matrix: 1237+ 2389+ 1702= 5328. On the other hand, a percentage of tweets were predicted with a wrong sentiment is 22%. This is calculated by adding up the values in the matrix corresponding to the incorrect predictions: 562+ 181+ 113+ 174+ 42+ 407= 1479. Based on these results, we can conclude that approximately 78% of the tweets were predicted correctly.

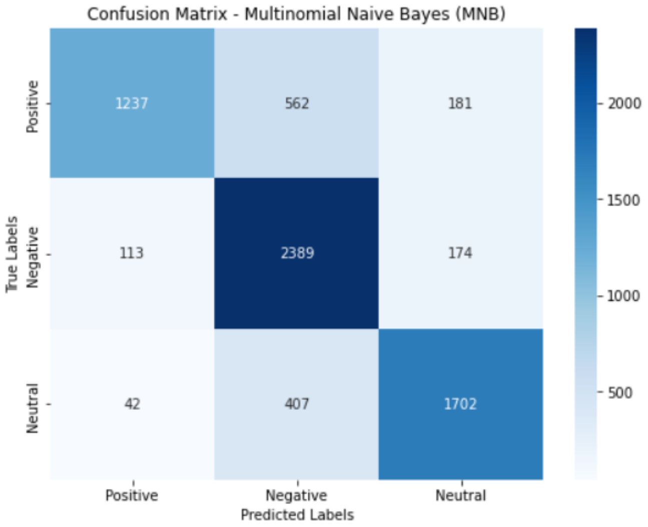

Fig. 7. Confusion matrix for multinomial naive bayes.

Fig.8 shows confusion matrix representing the performance of K-Nearest Neighbors (K-NN) using TF-IDF vectorizer that has the highest accuracy in detecting natural tweets which weigh is 26 over 68 to be 38. The confusion matrix shows the relationship between the actual sentiment of tweets (true label) and the predicted sentiment of tweets (predicted label).

According to the confusion matrix, a percentage of tweets were predicted with the same sentiment as their true label is 67%. This is calculated by adding up the values in the matrix: 1098+1171+1414= 3683. On the other hand, a percentage of tweets were predicted with a wrong sentiment is 19%. This is calculated by adding up the values in the matrix corresponding to the incorrect predictions: 359+114+490+467+ 94+239= 1763. Based on these results, we can conclude that approximately 67% of the tweets were predicted correctly.

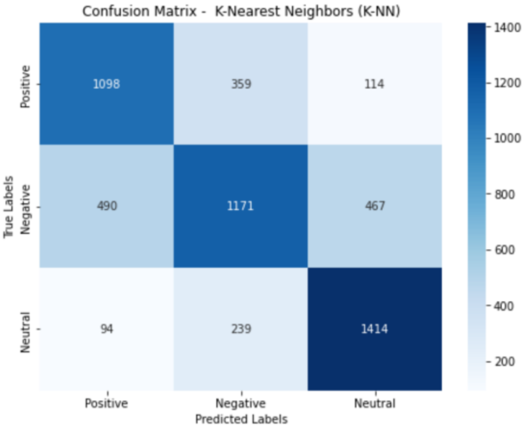

Fig. 8. Confusion matrix for k-nearest neighbors.

The remaining algorithms, namely Support Vector Machines (SVM), Random Forest, Adaboost, and Valence Aware Dictionary and Sentiment Reasoner (VADER), also outperform in detecting negative sentiment. This suggests that detecting negative tweets is relatively easier. Furthermore, it is worth noting that none of the proposed models achieved the highest accuracy in detecting positive sentiment, making the detection of positive tweets more challenging.

VI. DISCUSSION

As Maximum Entropy (ME) algorithm shows, its effectiveness between Random Forest, K-Nearest Neighbors (KNN), Support Vector Machines, Valence Aware Dictionary Sentiment Reasoner (VADER), Naive Bayes, and Adaboost in analysing the sentiment of 𝕏 platform users, this result can be generalized to apply on another platform that has the same nature as 𝕏 platform, like Reddit, Threads, and it is expected to perform with the same efficiency. At the end, the result of this research can be useful for those studies that concern knowing how to do sentiment analysis in areas close in nature to 𝕏 platform, a microblogging service.

VII. CONCLUSION

To sum up, the research explored all 7 algorithms, which are Support Vector Machines, Random Forest, K-Nearest Neighbors (KNN), Valence Aware Dictionary and sentiment Reasoner (VADER), Naive Bayes, Adaboost, and Maximum Entropy. Our experiment results show that Maximum Entropy (ME) has the best result with 81% accuracy in identifying the sentiment of the tweet in the "Twitter Tweets Sentiment Dataset". This result is ahead of Support Vector Machine (SVM) with accuracy 80%, Random Forest (RF) with 80%, Multinomial Naive Bayes (MNB) with 78%, AdaBoost with 74%, K-Nearest Neighbors (K-NN) with 68%, and Valence Aware Dictionary and sentiment Reasoner (VADER) with 55%. Multinomial Naive Bayes (MNB) achieved the highest accuracy in detecting negative sentiment tweets. While K-Nearest Neighbors (K-NN) has the highest accuracy in detecting natural tweets. Generally, it is also found that most algorithms yield good results as they are applied on pre-processed text "selected_text", which contains words that clearly show user emotion. This text is generated after doing words extraction, meaning the cleaner the data, the more accurate the results can be. So even if powerful algorithms are used without performing feature extraction, they cannot fully leverage the advantages of the algorithms. Overall, the paper provides valuable insights into how text mining employs machine learning in sentiment analysis on 𝕏 platform. These insights serve as a fundamental block for future research and improvements in the design and implementation of sentiment analysis algorithms on different social media platforms. Businesses also depend on sentiment analysis for making informed decisions.

VIII. FUTURE WORK

Summarizing our research, we have gained valuable knowledge regarding the effectiveness of machine learning-based sentiment analysis techniques on social media data (X platform). However, it is crucial to acknowledge the limitations of our study.

Firstly, our dataset was limited to English-language content. It would be beneficial to include a wider range of tweet types, sentiments, and additional features such as time, location, and number of likes. Furthermore, narrowing our analysis to a specific domain, such as product reviews, may restrict the applicability of our findings.

Moreover, our study employed a straightforward approach using TF-IDF for feature extraction and multiple sentiment analysis techniques, including Maximum Entropy (ME), Support Vector Machine (SVM), Random Forest (RF), Multinomial Naive Bayes (MNB), AdaBoost, K-Nearest Neighbors (K-NN), and Vader. It is important to note that there are numerous alternative techniques and models that could be explored in future research, such as the Bag of Words approach in the feature extraction stage. Different approaches may yield varying results depending on the specific data and sentiment being examined.

Despite these limitations, our research provides valuable insights into the effectiveness of machine learning-based sentiment analysis techniques on social media data within a controlled framework. By using a well-balanced dataset, employing a uniform feature extraction method, and comparing seven popular sentiment analysis techniques, we were able to evaluate the most reliable approach in a general domain and highlight the advantages of different techniques.

REFERENCES

[1] M. Cindo, D. P. Rini, and E. Ermatita, “SENTIMENT ANALYSIS ON TWITTER BY USING MAXIMUM ENTROPY AND SUPPORT VECTOR MACHINE METHOD,” Sinergi, vol. 24, no. 2, p. 87, Apr. 2020, doi: 10.22441/sinergi.2020.2.002.

[2] G. Popoola, K. Abdullah, G. S. Fuhnwi, and J. O. Agbaje, “Sentiment analysis of financial news data using TF-IDF and machine learning algorithms,” 3rd IEEE International Conference on AI in Cybersecurity (ICAIC), Feb. 2024, doi: 10.1109/icaic60265.2024.10433843

[3] L. Ramasamy, S. Kadry, Y. Nam, and M. Meqdad, “Performance analysis of sentiments in Twitter dataset using SVM Models,” International Journal of Electrical and Computer Engineering (IJECE), vol. 11, no. 3, p. 2275, Jun. 2021. doi:10.11591/ijece.v11i3.pp2275-2284.

[4] A. Neogi, K. Garga, R. Mishra, and Y. Dwivedi, “Sentiment analysis and classification of Indian Farmers’ protest using Twitter data,” International Journal of Information Management Data Insights, https://www.sciencedirect.com/science/article/pii/S2667096821000124?via%3Dihub.

[5] Jannatul Ferdoshi, Samirah Dilshad Salsabil, Ehsanur Rahman Rhythm, H. Kabir, and Annajiat Alim Rasel, “Unveiling Twitter Sentiments: Analyzing Emotions and Opinions through Sentiment Analysis on Twitter Dataset,” Oct. 2023, doi: https://doi.org/10.1109/cats58046.2023.10424206 .

[6] A. M. Rahat, A. Kahir, and A. K. M. Masum, “Comparison of Naive Bayes and SVM Algorithm based on Sentiment Analysis Using Review Dataset,” IEEE Xplore, Nov. 01, 2019. https://ieeexplore.ieee.org/document/9117512?arnumber=9117512 .

[7] A. Rahman, M. Sadat, and Md. S. Siddik, “Sentiment analysis on Twitter Data: Comparative study on different approaches,” International Journal of Intelligent Systems and Applications, vol. 13, no. 4, pp. 1–13, Aug. 2021, doi: 10.5815/ijisa.2021.04.01.

[8] G. Mutanov, V. Karyukin, and Z. Mamykova, “Multi-Class Sentiment Analysis of Social Media Data with Machine Learning Algorithms,” Computers, Materials & Continua/Computers, Materials & Continua (Print), vol. 69, no. 1, pp. 913–930, Jan. 2021, doi: 10.32604/cmc.2021.017827.

[9] J. Ferdoshi, S. D. Salsabil, E. R. Rhythm, M. H. K. Mehedi, and A. A. Rasel, “Unveiling Twitter Sentiments: Analyzing emotions and opinions through sentiment analysis on Twitter Dataset,” in Computer Applications & Technological Solutions (CATS), Oct. 29, 2023. doi: 10.1109/cats58046.2023.10424206.

[10] A. Rahman, M. Sadat, and Md. S. Siddik, “Sentiment analysis on Twitter Data: Comparative study on different approaches,” International Journal of Intelligent Systems and Applications, vol. 13, no. 4, pp. 1–13, Aug. 2021, doi: 10.5815/ijisa.2021.04.01.