Sentiment Analysis of Indonesia 2024 Election with a Comparison of Naive Bayes and KNN Algorithms on Twitter

Mespin Andayani [1], Fitri Marisa [1], Rangga Pahlevi Putra [1]

[1] *Widyagama University of Malang, Department of Informatic Engineering, Malang, Indonesia*

Abstract **– In the 2024 General Election, the Recapitulation Information System (SIREKAP) was used to capture the vote count results electronically. However, the use of SIREKAP raises various opinions in the community, both positive and negative, regarding the accuracy of the uploaded data. This study aims to analyze public sentiment towards the use of SIREKAP in the 2024 Election through Twitter data, using the Naive Bayes and KNN algorithms. The results showed that the Naive Bayes algorithm was superior with an accuracy of 93.37%, while KNN achieved an accuracy of 77.92%. The novelty of this research is to conduct sentiment analysis and provide insight into how people perceive the use of SIREKAP in the 2024 Election through Twitter data.**

Keywords **– sentiment analysis, SIREKAP, naïve bayes, K-nearest neighbors, election.**

1. Introduction

Indonesia is a country that adheres to the democratic system [1]. In democracies, members of parliament as well as the president are elected through elections [2].

DOI: 10.18421/SAR73-06
https://doi.org/10.18421/SAR73-06

Corresponding author: Fitri Marisa,
Widyagama University of Malang, Department of Informatic Engineering, Malang, Indonesia
Email: fitrimarisa@widyagama.ac.id

Received: 16 July 2024.
Revised: 05 September 2024.
Accepted: 12 September 2024.
Published: 27 September 2024.

These elections are regulated by Law No. 7/2017 which emphasizes the principles of direct, general, free, secret, honest, and fair elections [3]. With the rapid development of technology, the electoral process has also been transformed. Many countries have adopted electronic systems to simplify the process of tallying election results and voting as a result of technological advances [4]. In Indonesia, the technology used is SIREKAP which captures the results of the Plano C calculation [5].

However, the use of the SIREKAP application faces problems such as the inaccuracy of the uploaded data, which has led to various pro and con opinions. One form of media used to convey the aspirations and thoughts of the community is through social media [1]. In recent years, social media has become an effective tool for accommodating public opinion on issues that occur [6]. Platforms such as Facebook, Twitter and Google+ provide a platform for sharing views, reviews and ratings [7]. In particular, Twitter serves as a key platform for political discussion and deliberation [8]. Twitter, as the largest news source, with more than 250 million active users per month, provides a user-friendly environment for expressing opinions and sharing viewpoints among users and can be a source of public views [9] [10]. Twitter gives users the ability to create instant messages that introduce ideas without any barriers. Twitter has about 500 million instant messages every day [11].

The massive amount of user-generated content on the twitter platform provides an opportunity to measure public sentiment towards products, services, events and famous personalities. Sentiment analysis, also known as opinion mining, plays an important role in understanding and analyzing this large amount of textual data. Sentiment Analysis can be used by following and describing the text polarity (positive, negative, and neutral) of tweets to build a general view on relevant topics [12] [13]. Based on previous research, many researchers have explored text data by classifying it to generate important information. Various machine learning methods can be applied to text classification tasks [9].

In sentiment analysis, machine learning classification methods for text such as Naïve Bayes, Support Vector Machine, Logistic Regression, and Lexicon Based, are used to find the best results [14].

In previous research [15] Sentiment analysis has been carried out on user reviews of the SIREKAP application on the Play Store using the Random Forest Classifier algorithm with 5000 data, resulting in an accuracy of 74%. Other research [4] has also analyzed the sentiment of SIREKAP application user reviews on the Play Store using the Naïve Bayes and Support Vector Machine (SVM) algorithms, with 2362 data. Both algorithms produce the same accuracy of 72%, but SVM is superior in precision and recall.

Based on these problems, this research was conducted to analyze sentiment towards the use of SIREKAP in the 2024 elections by comparing the performance of two algorithms, namely Naïve Bayes and K-Nearest Neighbors (KNN), with the aim of understanding people's perceptions and reactions to the use of SIREKAP in the 2024 elections and evaluating the performance of the two algorithms in classifying sentiment. The research data to be used amounted to 2793, which was obtained from Twitter or X.

2. Methodology

In this research, Naïve Bayes and K-Nearest Neighbors algorithms are used for the classification of public sentiment towards the SIREKAP application. The steps involved are illustrated in Figure 1. The research begins with the collection of Twitter data related to SIREKAP, and then pre-processing and labeling stages are performed to categorize the data as positive, negative, or neutral. Next, feature extraction is performed using the TF-IDF method to determine the importance of words in a document relative to the entire document set. After that, the data is divided into two parts: training data and testing data. Classification is performed using Naïve Bayes and K-Nearest Neighbors algorithms. After that, the model is tested and evaluation results are obtained. Visualization is used to provide a deeper understanding of the sentiment analysis results of the SIREKAP application on Twitter.

A. Data Collection

The data collection process was done through crawling with the Tweet Harvest tool, using Twitter authorization tokens. Crawling is a process used to collect data from the internet domain. The process involves automated software called a web crawler or spider, which systematically browses the web, following links between web pages, and collecting information along the way [16]. Data crawling began on February 14, 2024, coinciding with the simultaneous elections, until March 20, 2024, when the election results were announced.

B. Pre-Processing

The pre-processing stage in this research aims to remove tweets that are problematic during data collection and select features in the form of words or terms. The pre-processing process is done using Python on Google Colab and involves several steps: Cleaning, Case Folding, Normalization, Stopword, Tokenization and Stemming. The following is an explanation of the pre-processing steps applied in this research [1]:

a. Cleaning
It is the process of removing unnecessary variables, such as URLs, hashtags, usernames, and duplicated data.
b. Case Folding is a step to convert all letters in a sentence into lowercase letters.
c. Normalization is the process of adjusting inappropriate or non-standard words into standard word forms based on the Big Indonesian Dictionary (KBBI), for example to correct typos.
d. Stopword
This process aims to remove words that have no significant meaning or are not related to the topic under discussion. Stopword processing is done using the NLTK library for Bahasa Indonesia available in Python.
e. Tokenization is the process of dividing a sentence into a series of words. Each sentence is broken down into words separately, and symbols and numbers are removed.
f. Stemming is the process of finding the root word of a word. In this stage, all affixes such as prefixes, inserts, suffixes, and duplications are removed. The goal is to obtain the basic form of the word.

C. TF-IDF

TF-IDF is one of the most commonly used weighting methods to measure the relatedness between words and documents. This technique is widely used in word feature extraction. However, TF-IDF is an unsupervised method that does not utilize known class information from the training dataset when assigning weights to a term, so the resulting weights may not fully reflect the importance of the term in the classification task [17].

D. Naïve Bayes

Naïve Bayes is a classification method that uses Bayes' Theorem to predict the category of data. This method works under the assumption that every feature in the data is not related to each other.

Simply put, Naive Bayes assumes that the presence of a feature in a category does not affect the presence of other features in that category [8]. This method performs a simple probability calculation to estimate the likelihood of a data belonging to a certain category [18].

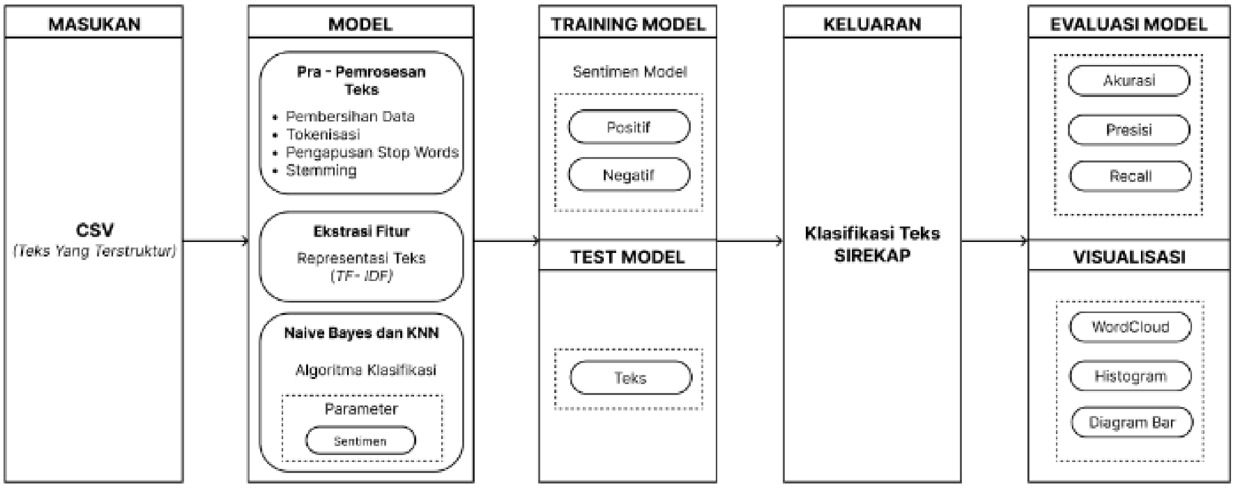

Figure 1. Research Stages

E. K-Nearest Neighbors

K-Nearest Neighbors or K-NN is a machine learning algorithm that classifies new objects based on a set of k nearest neighbors. In the K-Nearest Neighbor algorithm, the class that appears most frequently among the nearest neighbors will be the classification result [9]. The steps of this algorithm include: first, determining the parameter k, which is the number of nearest neighbors. Then, calculate the distance of the object to the training data and sort them in ascending order (from highest to lowest value). Finally, it collects the classes of the nearest neighbors according to the value of k and uses the most occurring category to predict the class [19]. KNN is one of the most popular Machine Learning algorithms applied in various fields. It classifies new test points based on the "k" nearest previously classified data points. If k=1, this approach is called nearest neighbor [20].

F. Evaluation

After the classification process is complete, the next step is to test the model using the confusion matrix. The confusion matrix is a useful and comprehensive representation of a classifier's performance on a data set that has known labels [21]. Confusion matrix is an important evaluation tool in statistical modeling, especially in the context of machine learning classification models. It helps assess the performance of a classification model by comparing the model's predicted results with the actual class of the data. Confusion matrix consists of four main components [4], namely:

1) True Positive (TP) is the number of samples that are truly positive and accurately predicted by the model.
2) True Negatives (TN) is the number of samples that are truly negative and also correctly predicted by the model.
3) False Positive (FP) is the number of samples that are actually negative but predicted as positive by the model (type I error).
4) False Negative (FN) is the number of samples that are actually positive but predicted as negative by the model (type II error).

G. Visualization

The results of sentiment analysis are displayed visually to provide an overview or general information about sentiment data related to the 2024 general election SIREKAP application, and the visualization results are displayed in the form of histograms, word clouds, and bar charts [22].

3. Results

This research divides the data into training data and test data. Training data is used to build a classification model, while test data is used to test the performance of the model. Data splitting is done using the split data method with a ratio of 90:10, 80:20, 70:30, 60:40, and 50:50 to find the best values for accuracy, precision, and recall. Sentiment analysis uses two calculation methods: Google Colab and RapidMiner Studio.

The use of these two tools is expected to improve the efficiency and final results of sentiment analysis, as previous experience with Google Colab did not result in optimal performance in terms of accuracy, precision, and recall.

A. Data Collection

The data collected were public comments about the SIREKAP application on Twitter, using the keyword "SIREKAP" and focused on tweets. The data collection process was done through crawling with the Tweet Harvest tool, using Twitter authorization tokens. Crawling results in 2,793 data that includes 15 pieces of information, such as username, tweet details, and creation date.

B. Pre-Processing

The following are the results about the pre-processing stages applied in this study:

a. Cleaning

In the cleaning process, the amount of initial data totaling 2793 changed to 2775 data. The results of data cleaning can be seen in Table 1.

Table 1. Cleaning Process Result

Before	After
PEMILU 2024 Bawaslu Sarankan KPU Hentikan Penayangan Informasi Sirekap Bawaslu telah mengirimkan surat berisi saran kepada KPU untuk menghentikan penayangan Sirekap. Baca Kompas https://t.co/pkIBLhPuvz @hariankompas	PEMILU 2024 Bawaslu Sarankan KPU Hentikan Penayangan Informasi Sirekap Bawaslu telah mengirimkan surat berisi saran kepada KPU untuk menghentikan penayangan Sirekap Baca Kompas

b. Case Folding

Case Folding is a step to convert all letters in a sentence into lowercase letters. The following results of case folding can be seen in Table 2.

Table 2. Case Folding Result

Before	After
PEMILU 2024 Bawaslu Sarankan KPU Hentikan Penayangan Informasi Sirekap Bawaslu telah mengirimkan surat berisi saran kepada KPU untuk menghentikan penayangan Sirekap Baca Kompas	pemilu 2024 bawaslu sarankan kpu hentikan penayangan informasi sirekap bawaslu telah mengirimkan surat berisi saran kepada kpu untuk menghentikan penayangan sirekap baca kompas

c. Normalization

This process uses Python operators and dictionaries to manage the dataset by normalizing abbreviations and slang. The dictionary used is a dictionary created by researchers based on the findings of typo words from the dataset in the ".txt" file format. The following results of normalization can be seen in Table 3.

Table 3. Normalization Result

Before	After
pemilu 2024 bawaslu sarankan kpu hentikan penayangan informasi sirekap bawaslu telah mengirimkan surat berisi saran kepada kpu untuk menghentikan penayangan sirekap baca kompas	pemilu 2024 bawaslu sarankan kpu hentikan penayangan informasi sirekap bawaslu telah mengirimkan surat berisi saran kepada kpu untuk menghentikan penayangan sirekap

d. Stopword

Stopword processing is done using the NLTK library for Indonesian available in Python. In addition, the researcher also performed a manual stopword removal process on the dataset to get maximum results, because automatic processing using Python is not always accurate or in accordance with the specific needs of the dataset. Table 4 shows the results of this word filtering process.

Table 4. Stopword Result

Before	After
pemilu 2024 bawaslu sarankan kpu hentikan penayangan informasi sirekap bawaslu telah mengirimkan surat berisi saran kepada kpu untuk menghentikan penayangan sirekap	pemilu 2024 bawaslu sarankan kpu hentikan penayangan informasi sirekap bawaslu mengirimkan surat berisi saran kpu menghentikan penayangan sirekap

e. Tokenization

Each sentence is broken down into words separately, and symbols and numbers are removed. However, in this research, numbers have important meaning and should not be removed. Therefore, the tokenization process will be performed without removing numbers, in order to retain relevant information and support the analysis. The results of this process are shown in Table 5.

Table 5. Tokenization Result

Before	After
pemilu 2024 bawaslu sarankan kpu hentikan penayangan informasi sirekap bawaslu mengirimkan surat berisi saran kpu menghentikan penayangan sirekap	['pemilu', '2024', 'bawaslu', 'sarankan', 'kpu', 'hentikan', 'penayangan', 'informasi', 'sirekap', 'bawaslu', 'mengirimkan', 'surat', 'berisi', 'saran', 'kpu', 'menghentikan', 'penayangan', 'sirekap']

f. Stemming

In this study, we used the Python library Sastrawi to stem the text in the dataset, which allowed us to reduce word variation and focus on analyzing the deeper meaning of the text. The results of this process are shown in Table 6.

Table 6. Stemming Result

Before	After
['pemilu', '2024', 'bawaslu', 'sarankan', 'kpu', 'hentikan', 'penayangan', 'informasi', 'sirekap', 'bawaslu', 'mengirimkan', 'surat', 'berisi', 'saran', 'kpu', 'menghentikan', 'penayangan', 'sirekap']	['pemilu', '2024', 'bawaslu', 'saran', 'kpu', 'henti', 'tayang', 'informasi', 'sirekap', 'bawaslu', 'kirim', 'surat', 'isi', 'saran', 'kpu', 'henti', 'tayang', 'sirekap']

C. TF-IDF

This stage assigns a value to each word that shows how often the keyword or other terms appear in the document. The following results of the feature extraction process using TF-IDF can be seen in Figure 2.

	aamiin	abad	abah	abai	abal	abang	abbyy	abdi	abdiyanto	abdul	...	yusril	yusuf	zakat	zalim	zaman
0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0
1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0
2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0
3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
2771	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0
2772	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0
2773	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0
2774	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0
2775	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0

Figure 2. TF-IDF Result

D. Naïve Bayes

In this study, the results of sentiment analysis using Naïve Bayes algorithm are compared between two platforms, namely Google Colab and RapidMiner, based on three metrics: accuracy, precision, and recall. RapidMiner showed superior performance by achieving an accuracy of 93.37%, which is much higher than the 50.00% accuracy achieved by Google Colab. In terms of precision, RapidMiner was also superior with a precision rate of 93.60%, while Google Colab only reached 70.00%. In addition, RapidMiner showed superiority in recall with a value of 94.05%, compared to 39.00% on Google Colab.

These results show that RapidMiner provides better performance for all three metrics measured. Therefore, it can be concluded that RapidMiner has significant advantages in terms of accuracy, precision, and recall compared to Google Colab, making it a better choice for sentiment analysis using the Naïve Bayes algorithm.

E. K-Nearest Neighbors

The results of sentiment analysis using the K-Nearest Neighbor (KNN) algorithm between two platforms, namely Google Colab and RapidMiner, based on three metrics: accuracy, precision, and recall. The KNN algorithm is used to classify new objects based on the attributes and training samples closest to the object, with the selected k values being 21, 23, and 25.

The results show that RapidMiner produces better performance with an accuracy rate of 77.92%, which is significant compared to the 55.00% accuracy achieved by Google Colab. In terms of precision, RapidMiner also excelled with a precision rate of 79.19%, while Google Colab only achieved 56.00%. RapidMiner also performed better in recall, achieving 73.62% compared to the 50.00% achieved by Google Colab.

From these results, it can be concluded that the model implemented using RapidMiner (with a 50:50 ratio and a K=21 value) performs better than the model implemented using Google Colab (with an 80:20 ratio and a K=23 value) in all measured metrics: accuracy, precision, and recall.

Therefore, for the datasets and experiments conducted, RapidMiner provided superior and consistent performance in all aspects of evaluation. The selection of K=21 proved to be the optimal choice for this dataset, indicating that the KNN model was able to capture the patterns in the data well and provide a good balance between classification accuracy and completeness.

F. Evaluation

The performance evaluation results show that RapidMiner provides the best performance for both algorithms used. RapidMiner excels at managing big data and provides features that make sentiment analysis easy, making it a highly efficient tool. Therefore, algorithm comparisons were conducted on the RapidMiner application to ensure the most accurate and efficient results.

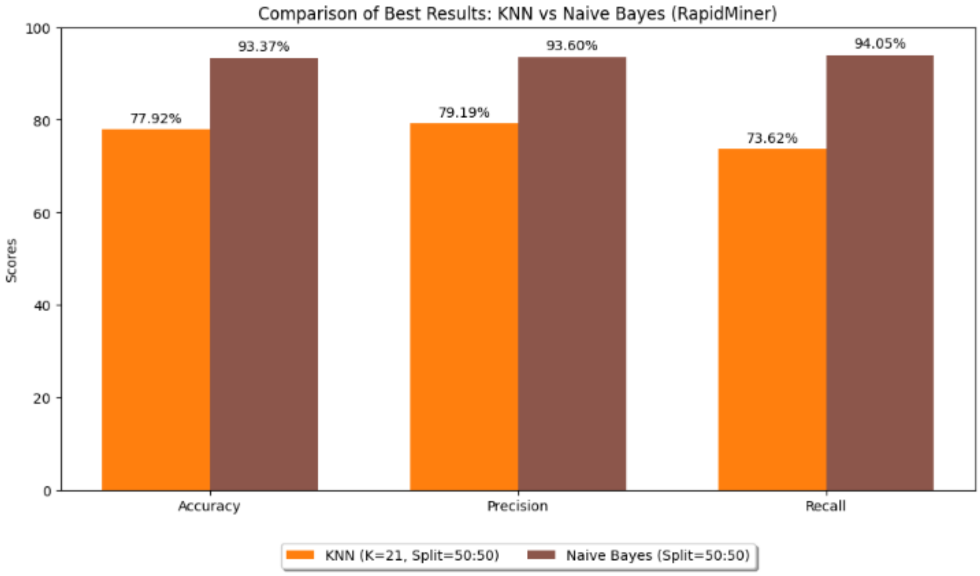

Figure 3. Comparison of Naive Bayes and KNN Algorithms

Figure 3 shows the comparison of the best results between K-Nearest Neighbors (KNN) and Naïve Bayes algorithms in RapidMiner application based on three evaluation metrics: accuracy, precision, and recall. In this study, the Naïve Bayes algorithm showed better performance than the previous study, with accuracy reaching 93.37%, precision 93.60%, and recall 94.05%. On the other hand, the results of the K-Nearest Neighbors (KNN) algorithm showed less optimal performance, with an accuracy of 77.92%, precision of 79.19%, and recall of 73.62%. This study uses data from Twitter and applies Naïve Bayes and KNN algorithms that are different from previous research. The findings highlight the superiority of Naïve Bayes in sentiment analysis, particularly in the context of Twitter data, and underscore the importance of selecting the appropriate algorithm to achieve optimal results in data analysis.

G. Visualization

In this research, the visualization stage uses the Matplotlib and Seaborn libraries. The following visualization results in this research can be seen in Figure 4.

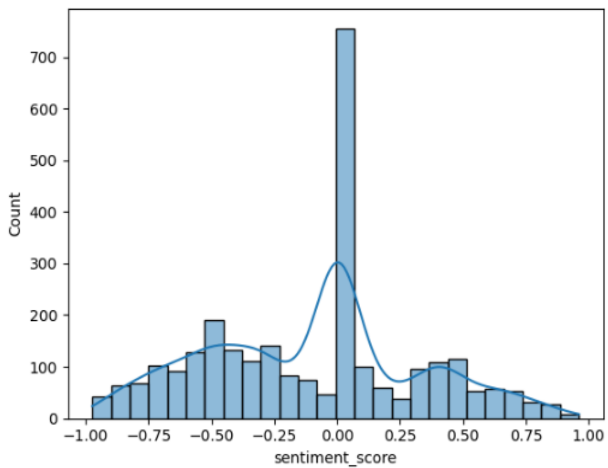

Figure 4. Histogram of Sentiment Score

Figure 4 displays a histogram of the sentiment score distribution, with kernel density estimation (KDE) curves to smoothen the distribution picture. The horizontal axis (X) shows the sentiment scores from -1.0 (negative) to 1.0 (positive), with zero as the neutral sentiment. The vertical axis (Y) shows the frequency of sentiment scores within each bin of the histogram.

Figure 5. WordCloud Positive Sentiment

Figure 5 shows the words that frequently appear in texts with positive sentiments. Larger words such as "sirekap, "no", "vote", "kpu", and "two" show high frequency in positive texts.

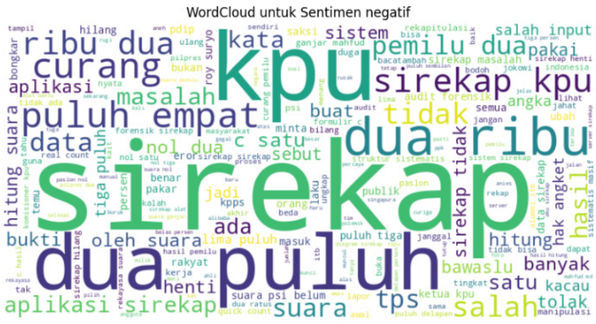

Figure 6. WordCloud Negative Sentiment

Figure 6 displays the frequency of words in the text with negative sentiments related to a particular topic. Larger words such as "sirekap", "kpu", "two", "twenty", and "thousand" show high frequency tendencies in negative contexts.

Figure 7. WordCloud Neutral Sentiment

Figure 7 displays the frequency of words in the text with neutral sentiment. Larger words such as "sirekap", "kpu", "two", "twenty", and "thousand" show high frequency in the neutral context.

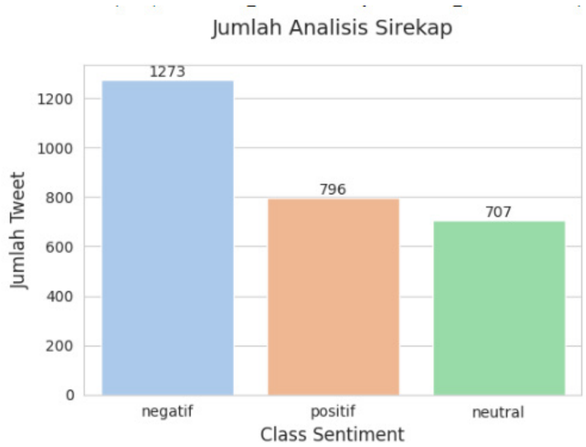

Figure 8. Number of Each Sentiment

Figure 8 shows the results of the sentiment analysis on "sirekap". This diagram divides the sentiment into three categories: negative, positive and neutral. Negative sentiment has the highest number of tweets, at 1273, indicating a majority of negative sentiment towards SIREKAP. Positive sentiment recorded 796 tweets, indicating a significant amount of positive sentiment although less than negative sentiment. Neutral sentiment recorded 707 tweets, indicating that there is also neutral sentiment towards SIREKAP. This diagram provides an overview of how SIREKAP is received by the public based on the sentiment analysis of the collected tweets.

4. Discussion

This study uses different data and algorithms from previous studies, namely Twitter data and Naïve Bayes and KNN algorithms. The results emphasize the superiority of Naïve Bayes in sentiment analysis, especially in the context of Twitter data, and show the importance of selecting the right algorithm to achieve optimal results in data analysis.

In addition, few previous studies have conducted sentiment analysis on the SIREKAP application, so this research makes a significant contribution in expanding the understanding of sentiment analysis on the platform. Furthermore, by using Twitter data as the basis of analysis, this research is able to capture sentiment dynamics and patterns that may not be visible in data from other sources, adding practical value and relevance to the results obtained.

5. Conclusion

This research successfully developed and implemented a sentiment analysis model of SIREKAP using Naive Bayes and K-Nearest Neighbors (KNN) algorithms. The results show that Naive Bayes is superior with 93.37% accuracy, 93.60% precision, and 94.05% recall, compared to KNN which has 77.92% accuracy, 79.19% precision, and 73.62% recall.

Out of 2,793 data, negative sentiment dominates with 1,273 data, followed by positive sentiment with 796 data, and neutral sentiment with 707 data. This shows that user reviews are more critical of SIREKAP.

This research emphasizes the importance for SIREKAP developers to pay attention to users' criticism to improve the quality of the platform. The contribution of this research is significant in sentiment analysis in Indonesia, especially in monitoring and evaluating digital applications such as SIREKAP. The results can help developers understand user feedback more deeply and quickly, and provide a basis for further research in improving the performance of classification models and other more adaptive techniques. It is hoped that further development will involve more diverse data and other machine learning techniques for more optimal and comprehensive results.

References:

[1]. Wenando, F. A., Hayami, R., & Novermahakim, A. Y. (2020). Tweet sentiment analysis for 2019 Indonesia presidential election results using various classification algorithms. In *2020 1st International Conference on Information Technology, Advanced Mechanical and Electrical Engineering (ICITAMEE),* 279-282. IEEE.

[2]. Kristiyanti, D. A., & Umam, A. H. (2019). Prediction of indonesia presidential election results for the 2019-2024 period using twitter sentiment analysis. In *2019 5th International Conference on New Media Studies (CONMEDIA)*, 36-42. IEEE. doi:10.1109/CONMEDIA46929.2019.8981823

[3]. Yulfa, R. I., Setiawan, B. H., Lourensius, G. G., & Purwandari, K. (2023). Enhancing Hate Speech Detection in Social Media Using IndoBERT Model: A Study of Sentiment Analysis during the 2024 Indonesia Presidential Election. In *2023 International Conference on Computer and Applications (ICCA)*, 1-6. IEEE.

[4]. Setiyawan, R., & Mustofa, Z. (2024). Comparison of the performance of naive bayes and support vector machine in sirekap sentiment analysis with the lexicon-based approach. *Journal of Soft Computing Exploration, 5*(2), 122-132. Doi: 10.52465/joscex.v5i2.367

[5]. Kusumadinata, A. A. (2024). Sosialisasi Sirekap Sebagai Alat Bantu Pemilu Nasional Tahun 2024. *Jurnal Masyarakat Madani Indonesia, 3*(2), 138-146. Doi: 10.59025/js.v3i2.213

[6]. Nugroho, D. K. (2021). US presidential election 2020 prediction based on Twitter data using lexicon-based sentiment analysis. In *2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, 136-141. IEEE.

[7]. Verma, M., Suryawanshi, P., & Deore, S. (2023). Election Outcome Prediction Using Sentiment Analysis On Twitter. *International Research Journal of Modernization in Engineering Technology and Science, 5*(6). Doi: 10.56726/IRJMETS41569

[8]. Gupta, S., Gaur, S. S., Sharmaa, P. R. A. T. I. B. H. A., & Gupta, A. (2024). Election Prediction Using Twitter Sentiment Analysis Using Naïve Bayes and Support Vector Machine. *SSRN,* 4754965. Doi: 10.2139/ssrn.4754965

[9]. Wisnu, G. R. G., Muttaqi, A. R., Santoso, A. B., Putra, P. K., & Budi, I. (2020). Sentiment analysis and topic modelling of 2018 central java gubernatorial election using twitter data. In *2020 International Workshop on Big Data and Information Security (IWBIS), 35-40.* IEEE. Doi: 10.1109/IWBIS50925.2020.9255583

[10]. Yavari, A., Hassanpour, H., Rahimpour Cami, B., & Mahdavi, M. (2022). Election prediction based on sentiment analysis using twitter data. *International Journal of engineering, 35*(2), 372-379. Doi: 10.5829/ije.2022.35.02b.13

[11]. Korani, W., & Mouhoub, M. (2024). Sentiment Analysis of Serious Suicide References in Twitter Social Network. In *Proceedings of the 12th International Conference on Agents and Artificial Intelligence, 2, ICAART,* 339-346, Valletta, Malta. Retrieved from: https://www.scitepress.org/Link.aspx?doi=10.5220/0008894003390346 [accessed: 10 July 2024]

[12]. IIPSeries—Conferences & Edited Books. (n.d.). *Home page.* IIPSeries. Retrieved form: https://iipseries.org/ [accessed: 10 July 2024]

[13]. Limboi, S., & Diosan, L. (2022). The Twitter-Lex Sentiment Analysis System. In *Proceedings of the 14th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2022) - KDIR*. Retrieved form: https://www.scitepress.org/Link.aspx?doi=10.5220/0011379900003335 [accessed: 11 July 2024]

[14]. AriBangsa, T., & Yudertha, A. (2024). Analisis Sentimen Opinimasyarakat Terhadap Pindahnya Ibu Kota Indonesia Dengan Menggunakan Klasifikasi Naïve Bayes. *Jurnal Teknoinfo, 18*(1), 226-238.

[15]. Herjanto, M. F. Y., & Carudin, C. (2024). Analisis Sentimen Ulasan Pengguna Aplikasi Sirekap Pada Play Store Menggunakan Algoritma Random Forest Classifer. *Jurnal Informatika dan Teknik Elektro Terapan, 12*(2). doi: 10.23960/jitet.v12i2.4192

[16]. HM, P. K., & S, Dr. S. B. (2023). AI - Based Solution for Web Crawling. *International Journal of Science and Research (IJSR), 12.* Doi: 10.21275/SR23331154330

[17]. Jin, Z., Lai, X., & Cao, J. (2020). Multi-label sentiment analysis base on BERT with modified TF-IDF. In *2020 IEEE International Symposium on Product Compliance Engineering-Asia (ISPCE-CN)*, 1-6. IEEE. Doi: 10.1109/ISPCE-CN51288.2020.9321861

[18]. Ö Şahinaslan, Ö., Dalyan, H., & Şahinaslan, E. (2022). Naive bayes sınıflandırıcısı kullanılarak youtube verileri üzerinden çok dilli duygu analizi. *Bilişim Teknolojileri Dergisi, 15*(2), 221-229. Doi: 10.17671/gazibtd.999960

[19]. Irawaty, I., Andreswari, R., & Pramesti, D. (2020). Development of Youtube Sentiment Analysis Application using K-Nearest Neighbors (Nokia Case Study). In *2020 3rd International Conference on Information and Communications Technology (ICOIACT),* 39-44. IEEE. Doi: 10.1109/ICOIACT50329.2020.9332151

[20]. Roongta, D. (2022). Double Ensemble kNN: Towards an Enhanced k-Nearest-Neighbors. *SSRN,* 4286996. Doi: 10.2139/ssrn.4286996

[21]. Krstinić, D., Braović, M., Šerić, L., & Božić-Štulić, D. (2020). Multi-label classifier performance evaluation with confusion matrix. *Computer Science & Information Technology, 1*, 1-14. Doi: 10.5121/csit.2020.100801

[22]. Salsabila, R. (2021). Penggunaan Metode Support Vector Machine untuk Proses Analisis Sentimen Mengenai Tanggapan Masyarakat Terhadap Pemberian Vaksin Covid-19. In *Prosiding Seminar SeNTIK, 5*(1), 213-220.