G-Tech: Jurnal Teknologi Terapan
Vol. 9, No. 3, July 2025, p. 1279-1288
E-ISSN: 2623-064X | P-ISSN: 2580-8737

Comparative Study of Naïve Bayes Classifier and Support Vector Machine Methods in Public Sentiment Analysis of Prabowo-Gibran's Free Lunch Program

Fikri Febrian[1]✉, Zaehol Fatah[2], Achmad Baijuri[3]

[1] Information Technology, Science and Technology, Universitas Ibrahimy, Indonesia
[2,3] Information Systems, Science and Technology, Universitas Ibrahimy, Indonesia

✉ **Corresponding Author:** Fikri Febrian (e-mail: febrianfikri21@gmail.com)

Article Information

Article History

Received : June 3, 2025
Revised : June 12, 2025
Published : July 04, 2025

***Keywords*:**

Sentiment Analysis;
Naïve Bayes classifier;
Support Vector Machine;
Social Media;
Free Lunch Program.

ABSTRACT

In today's digital era, social media has become the main platform for people to voice their opinions on social and political issues. One of the most discussed topics is the free lunch program of President-elect Prabowo Subianto and Vice President-elect Gibran Rakabuming Raka. The program triggered various public reactions, making it relevant for sentiment analysis. The purpose of this study is to compare the performance of two text classification algorithms-Naïve Bayes and Support Vector Machine (SVM)-in classifying public sentiment towards the program. The dataset was obtained from Kaggle, with 657 initial data. After preprocessing, 156 data remained, consisting of 127 negative sentiments and 31 positive sentiments. Data processing followed the CRISP-DM framework, with Python and Scikit-learn used in model training. The results showed that the naive bayes classifier performed better with 84.38% accuracy, 86.90% precision, and 84.38% recall. Support Vector Machine showed lower performance in all metrics. In addition, the Naive Bayes Classifier was able to classify sentiments in a more balanced manner. The analysis was performed using Jupyter Notebook, and the final model was implemented through a Streamlit-based web interface.

INTRODUCTION

In today's digital era, social media has become one of the main means for the public to express opinions and responses to social issues and government policies. Platforms such as Twitter, Facebook, and other digital media record various public expressions on various events, including national policies. One policy that has captured widespread public attention is the free lunch program promoted by President Prabowo Subianto and Vice President Gibran Rakabuming Raka. The program generated a wide spectrum of responses, ranging from positive support to sharp criticism, which were recorded massively through digital channels.

The high level of public attention to the free lunch program requires an in-depth and measurable understanding of the public opinion that spreads on social media, because the analysis of public opinion data will affect national policies concerning public welfare. The government and policy makers need an analytical tool that is not only able to capture public sentiment quickly, but also accurately. Without the right approach, complex and diverse public opinion can easily be misinterpreted or ignored, which ultimately results in inaccurate strategic decision-making.

Based on this problem, sentiment analysis is a strategic approach to understanding public perception of an issue in a systematic and measurable way. Sentiment analysis, as part of natural language processing, plays a role in classifying opinions into categories such as positive, negative,

or neutral. (Miftahusalam et al., 2022)This technique enables information extraction from large-scale text data, making it easier for stakeholders to assess public opinion. (Fitriyah et al., 2020) Along with the increasing complexity of data sourced from social media, the use of data mining techniques in sentiment analysis is becoming increasingly relevant.(Darwis et al., 2021a) Several common methodologies such as KDD, SEMMA, and CRISP-DM have been developed to structure the information extraction process systematically.(Amri, 2020)CRISP-DM was chosen in this study because of its comprehensive and iterative framework.

Among the popular algorithms in text classification, Naive Bayes Classifier (NBC) and Support Vector Machine (SVM) stand out for their high performance in previous studies. NBC is known for its simple and efficient probabilistic approach, (Darwis et al., 2021b)while SVM excels in handling complex and linearly disjoint data.(Arsi & Waluyo, 2021) Although both have their own advantages, the comparison of their performance in the context of Indonesian socio-political policy issues has not been widely studied, especially with a systematic approach such as CRISP-DM.

Therefore, this study has both theoretical and practical significance. From the theoretical side, this study contributes to the development of public sentiment analysis studies through a comparative approach of two popular classification methods. While from the practical side, the results of this study are expected to be a reference for the government, social media analysts, and public policy researchers in choosing the right classification method to analyze public opinion on strategic policies. Thus, this research not only enriches the scientific treasure in the field of text mining, but also provides direct benefits in the context of data-based policy making.

RESEARCH METHODS

Crisp-DM

The CRISP-DM method serves as a systematic analysis strategy to effectively solve problems in research and business challenges. (Fitriani et al., 2022) The CRISP-DM method was chosen in this study because it provides a systematic and structured framework in the mini data process, so the researcher decided to use it as the main approach. The CRISP-DM method was developed by a consortium of companies at the initiative of the European Commission in 1996 and is now widely used as a process standard in data mining applications. (Cholifah Sastya & Nugraha, 2023)

Datasets & Data Splitting

Data division is done by separating 80% as training data and 20% as testing data. Training data is used to build patterns in the model, while testing data is used to evaluate the performance of the model.(Alrasyid et al., 2024)

Cross-validation method

Cross-validation is a data resampling technique that aims to estimate how much prediction error may occur in the model, as well as used to optimize model parameters to produce better performance. (Wijiyanto et al., 2024)In this research, the K-Fold Cross Validation method is used with a value of k = 5, which means the data is divided into 5 parts (fold). Each part is used alternately as test data, while the other part is used as training data. This process is repeated 5 times, and the evaluation results from each iteration are averaged to get the final score.

Naïve Bayes Classifier

In the world of data classification, a simple yet effective approach is often the first choice, especially when computational efficiency is needed. One method that meets this criteria is the Naïve Bayes Classifier (NBC). Naïve Bayes is a classification algorithm in supervised learning that bases its probability estimates on Bayes' Theorem, with the assumption that each feature is independent of each other.(Rosyida et al., 2023.)This algorithm classifies data based on a simple probabilistic approach by applying Bayes' Theorem. This method assumes that each feature in the

data is independent, making calculations faster and more efficient (Singgalen, 2023). This method is able to estimate the possibility of an event occurring in the future by relying on patterns and experiences from previous events.(Dwiramadhan et al., 2022) Derived from Bayes' theorem, is a popular classification method that performs well even when training data is limited, because its parameter estimation is simple. Assuming independent features, this method only requires the variance of each feature within a class, without having to calculate the entire covariance matrix, making the classification process more efficient.(Felicia Watratan et al., 2020)

Support Vector Machine

Support Vector Machine (SVM) is a classification and prediction algorithm that aims to identify the optimal hyperplane that separates many data classes. This method is effective for non-linear data that cannot be separated by a straight line.(Acuan Supian et al., 2024) The SVM algorithm is also known to determine the most optimal hyperplane in the input space, which plays a role in effectively separating one class from another.(Rani Yunita, 2023)This technique trains on labeled data (supervised learning) to find the optimal hyperplane that can classify new data. (Gaffar et al., 2024)

$$f(x_d) = \sum_{\{i=1\}_i^{\{n_s\}\alpha}} {}_{y_i}\langle x_i, x_d\rangle + b \quad (1)$$

Information:

x_d = Predictable test data

x_i = i-th support vector data

y_i = Label from x_i

x_i, x_d = between x_i, x_d , reflects the linear similarity

b = Bias or error value

Model Evaluation

The performance of each model is assessed using an evaluation framework that includes key metrics like Accuracy, Precision, and Recall. This assessment is conducted through a confusion matrix, which helps analyze and compare the model's predicted outcomes with the true data labels. Essentially, the confusion matrix provides detailed information on the number of correctly and incorrectly classified data points, facilitating the measurement of how accurately and reliably the model performs its analysis.(Suryati et al., 2023)

Table 1. Confusion Matrix

Actual	Sentiment	
	Positive	**Negative**
Positive	True Positive (TP)	True Negative (TN)
Negative	False Positive (FP)	False Negative(FN)

Accuracy

Accuracy is the proportion of correct predictions compared to the total predictions, which indicates the overall performance of the classification model.

$$\boldsymbol{Accuracy} = \frac{\boldsymbol{TP+TN}}{\boldsymbol{TP+TN+FP+FN}} \quad (2)$$

Recall

Recall shows the ability of the system to find all the correct relevant data. Its value is calculated by comparing the number of data successfully found (true positive) with the total relevant data available (true positive + false negative).(Rihan Maulana,2023.)

$$\boldsymbol{Recall} = \frac{\boldsymbol{TP}}{\boldsymbol{TP+FN}} \quad (3)$$

Precission

Precision measures how many positive predictions are correct out of all positive predictions made by the model. This metric reflects the level of accuracy of the model in classifying a class.

$$\boldsymbol{Precission} = \frac{\boldsymbol{TP}}{\boldsymbol{TP+FP}} \quad (4)$$

Application Development with Stremlit

Streamlit is an open-source framework developed using the Python programming language. This framework is specifically designed to facilitate the creation of data science and machine learning applications with interactive user interfaces.(Ayu Syafarina & Kalimantan Muhammad Arsyad Al Banjari Banjarmasin,2023 .) Streamlit integrates with popular frameworks and libraries in the Python ecosystem, such as NumPy, Pandas, Matplotlib, and Scikit-learn. This integration makes Streamlit a highly sought-after choice among data science and machine learning practitioners, as it supports integrated and efficient data analysis and model development workflows. This framework was created to enable the development of data mining applications that leverage machine learning and data science systems.

RESULTS AND DISCUSSION

Design Sistem

The comparison between these methods began with searching for data on the Keagle platform using the keyword "free lunch," which was then processed and classified using two commonly used classification methods, namely the naïve Bayes classifier and support vector machine.

Stages of the sentiment analysis

process flowThis section describes the flow of the sentiment analysis process, which can be seen in Figure 1 below.

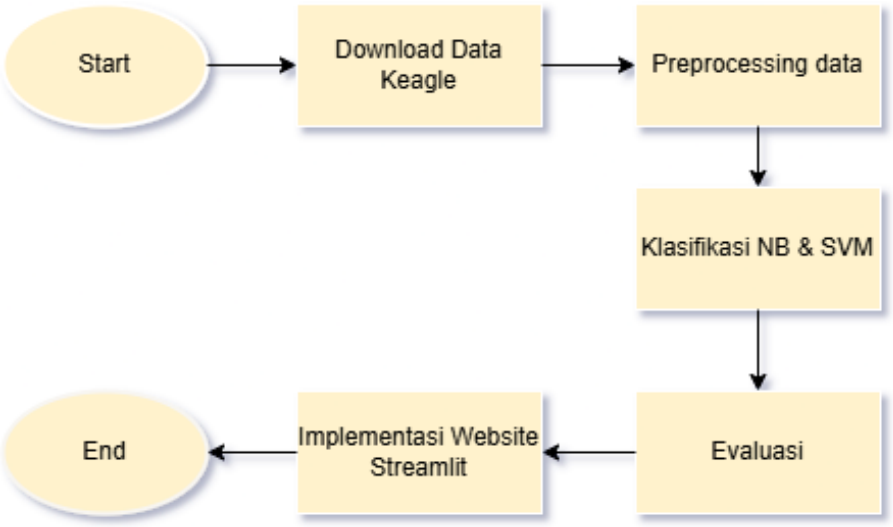

Figure 1. Research Process Flow

Download Kaggle Data

This study utilizes the Keagle platform to obtain data that will be used for sentiment analysis and to compare the naïve bayes classifier and support vector machine methods. A total of 656 sentiment data points were obtained. The data obtained is as shown in the following table.

Table 1. Downloaded data results

No	Full Text, Label
1	PKS: Kalau Dana BOS Dipakai Makan Siang Gratis, Terus Guru Honorer Mau Digaji Pakai Apa? https://t.co/1uStKOmGb1,N
2	@MrsTitins @ZackXChristy @akunbarublek @ignasbowo Gak baca programnya apa gimana ya? Targetnya aja makan siang gratis buat anak sekolah, yg putus sekolah ya otomatis gak dapet dong,P
3	@AnggaPutraF Yang dibutuhkan saat ini makan siang gratis,P
4	@Ipaahu Makan siang Gratis tapi otaknya jadi tahi,N

Preprocessing

The purpose of data preprocessing is to improve and adjust the data to the needs of the algorithm. In this study, there are several stages of data preprocessing, including;

a) Text normalization and cleaning

This process begins with normalizing lowercase letters and cleaning the text of non-informative characters such as URLs, mentions, numbers, and punctuation marks using regular expressions. As shown in the following table;

Table 2. Results of data normalization and cleaning

No	Full Text	Normalisasi
1	PKS: Kalau Dana BOS Dipakai Makan Siang Gratis, Terus Guru Honorer Mau Digaji Pakai Apa? https://t.co/1uStKOmGb1,	pks kalau dana bos dipakai makan siang gratis terus guru honorer mau digaji pakai apa
2	@MrsTitins @ZackXChristy @akunbarublek @ignasbowo Gak baca programnya apa gimana ya? Targetnya aja makan siang gratis buat anak sekolah, yg putus sekolah ya otomatis gak dapet dong,	gak baca programnya apa gimana ya targetnya aja makan siang gratis buat anak sekolah yg putus sekolah ya otomatis gak dapet dong
3	@AnggaPutraF Yang dibutuhkan saat ini makan siang gratis,	yang dibutuhkan saat ini makan siang gratis
4	@Ipaahu Makan siang Gratis tapi otaknya jadi tahi	makan siang gratis tapi otaknya jadi tahi

b) Tokenizing dan Stopword

After that, tokenization is performed to break the sentence into words, followed by stopword removal using the NLTK library. As shown in the following table;

Table 3. Tokenization and Stopword data results

No.	Full Text	Normalisasi
1	pks kalau dana bos dipakai makan siang gratis terus guru honorer mau digaji pakai apa	['pks', 'dana', 'bos', 'dipakai', 'makan', 'siang', 'gratis', 'terus', 'guru', 'honorer', 'mau', 'digaji', 'pakai',]
2	gak baca programnya apa gimana ya targetnya aja makan siang gratis buat anak sekolah yg putus sekolah ya otomatis gak dapet dong	['gak', 'baca', 'programnya', 'gimana', 'targetnya', 'aja', 'makan', 'siang', 'gratis', 'buat', 'anak', 'sekolah', 'yg', 'putus', 'sekolah', 'otomatis', 'gak', 'dapet', 'dong',]
3	yang dibutuhkan saat ini makan siang gratis	['dibutuhkan', 'makan', 'siang', 'gratis',]
	makan siang gratis tapi otaknya jadi tahi	['makan', 'siang', 'gratis', 'tapi', 'otaknya', 'jadi', 'tahi']

c) Stemming

The words are then destemmed to return to their original form. As in the following table;

Table 4. Data Stemming Results

No.	Full Text	Normalisasi
1	pks kalau dana bos dipakai makan siang gratis terus guru honorer mau digaji pakai apa	['pks', 'dana', 'bos', 'pakai', 'makan', 'siang', 'gratis', 'terus', 'guru', 'honorer', 'mau', 'gaji', 'pakai',]
2	gak baca programnya apa gimana ya targetnya aja makan siang gratis buat anak sekolah yg putus sekolah ya otomatis gak dapet dong	['gak', 'baca', 'program', 'gimana', 'target', 'aja', 'makan', 'siang', 'gratis', 'buat', 'anak', 'sekolah', 'yg', 'putus', 'sekolah', 'otomatis', 'gak', 'dapet', 'dong',]
3	yang dibutuhkan saat ini makan siang gratis	['butuh', 'makan', 'siang', 'gratis',]
4	makan siang gratis tapi otaknya jadi tahi	['makan', 'siang', 'gratis', 'tapi', 'otak', 'jadi', 'tahi']

d) Labeling

This is the process of determining whether a text is a positive or negative text. As in the following table;

Table 5. Data Labeling Results

No.	Text	Label
1	['pks', 'dana', 'bos', 'pakai', 'makan', 'siang', 'gratis', 'terus', 'guru', 'honorer', 'mau', 'gaji', 'pakai',]	Positive
2	['gak', 'baca', 'program', 'gimana', 'target', 'aja', 'makan', 'siang', 'gratis', 'buat', 'anak', 'sekolah', 'yg', 'putus', 'sekolah', 'otomatis', 'gak', 'dapet', 'dong',]	Negative
3	['butuh', 'makan', 'siang', 'gratis',]	Positive
4	['makan', 'siang', 'gratis', 'tapi', 'otak', 'jadi', 'tahi']	Negative

From the overall data, the sum of the numbers of each label is shown in Figure 2.

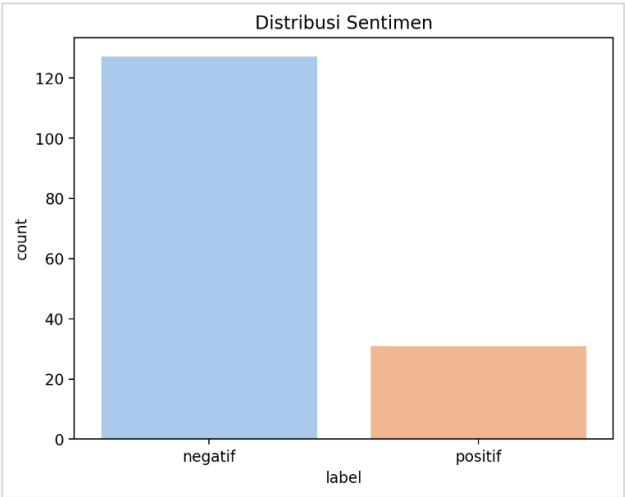

Figure 2. Data Labeling Result

The collected data was divided into 80% for training and 20% for testing, with 127 data used for training and 31 data for testing.

Classification

a) Naïve Bayes Classifier

The implementation of the resulting data set is done using the Python programming language with Jupyter Notebook for classification using the naive bayes classifier algorithm. In the algorithm, a value, namely " MultinomialNB()," produces a confusion matrix in the image below.

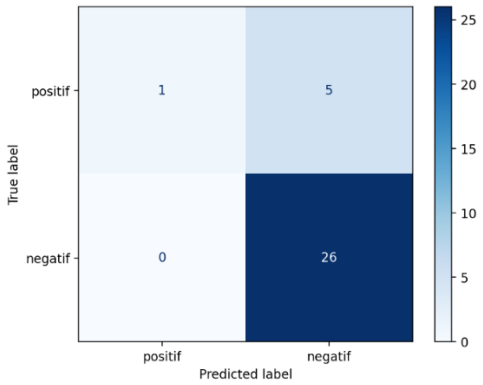

Figure 3. Naïve Bayes Classifier confusion matrix result value

$$\boldsymbol{Accuracy} = \frac{\mathbf{26+120}}{\mathbf{26+120+7+5}} \quad \boldsymbol{84{,}38\%}$$

$$\boldsymbol{precision} = \frac{\mathbf{26}}{\mathbf{26+5}} \quad \boldsymbol{86{,}90\%}$$

$$\boldsymbol{Recall} = \frac{\mathbf{26}}{\mathbf{26+7}} \quad \boldsymbol{84{,}38\%}$$

There were 26 correct positive predictions and 120 correct negative predictions, while 7 negative data were misclassified as positive and 5 positive data were not detected. This reflects that the model performs very well with high precision and recall rates, although there are still a few classification errors that can be minimized.

b) Support Vector Machine

In this classification project, Python programming language is used with the help of Jupyter Notebook as an interactive development environment. The approach used is the Support Vector Machine (SVM) algorithm, which is known to be effective in solving classification problems, especially when the data is not too large and the features are quite clearly separated. In the implementation, the SVM model utilizes the kernel parameter set to 'linear', based on the assumption that the data is linearly separable within the feature space.

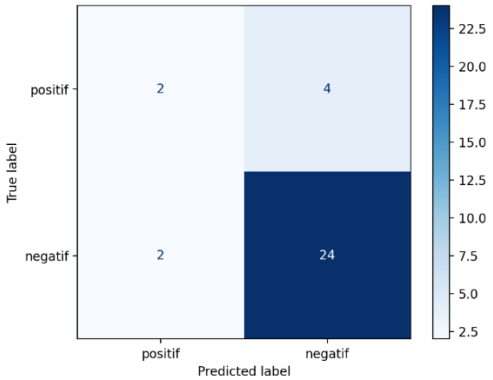

Figure 3. Support Vector Machine confusion matrix result value

$$Accuracy = \frac{25+118}{25+118+9+6} \quad 81.25\%$$

$$presisi = \frac{25}{26+9} \quad 79.02\%$$

$$Recall = \frac{25}{25+6} \quad 81.25\%$$

There were 25 correct positive predictions and 118 correct negative predictions, while 9 negative data were misclassified as positive and 6 positive data were not recognized. Although its performance was quite good, the SVM model showed slightly lower accuracy than Naive Bayes, so it still needs improvement in terms of classification precision.

Comparison Algorithms & Model Evaluation

To determine the performance of both algorithms, calculations were performed based on three evaluation metrics, namely Accuracy, Precision, and Recall, as shown in the following table.

Table 2. Comparison of the two algorithms

Algorithm	Accuration	Precission	Recall
Naive Bayes Classifier	84,38%	86,90%	84,38%
Support Vector Machine	81,25%	79,02%	81,25%

The comparison results between the Support Vector Machine and Random Forest algorithms show that, with rounded values, the Naïve Bayes algorithm has higher values, namely an accuracy of 84.31%, precision of 86.90%, and recall of 84.38%, or an average score of 74.42%. These results are also reliable because, based on the cross-validation method, the naive bayes method is indeed superior, producing an accuracy value of 78.51%, precision of 68.98%, and recall of 71.69%, in contrast to the svm method, which only produces an accuracy of 74.66%, precision of 68.44%, and recall of 71.21%.

This advantage can theoretically be explained by the probabilistic and generative nature of Naïve Bayes, which is capable of directly modeling class distributions. Additionally, this algorithm is highly effective for high-dimensional and sparse data, such as text vector representations. The assumption of feature independence, though simple, often aligns well with text classification tasks, making Naïve Bayes more stable and efficient in handling natural language variations and limited datasets.

Comparison sentiment analysis using streamlit

The developed Support Vector Machine model is then imported using the Pickle module, so that it can be integrated and used on the website via the Streamlit library.

Figure 4. Model implementation using streamlit

CONCLUSIONS AND SUGGESTIONS

Conclusions

Based on performance evaluation results, the Naïve Bayes algorithm demonstrated superior performance compared to Support Vector Machine and Random Forest in the task of sentiment analysis for the free lunch program. This superiority is evident from the higher accuracy, precision, and recall achieved, both in direct testing and cross-validation. This indicates that the probabilistic characteristics and its ability to handle high-dimensional text data make Naïve Bayes highly suitable for sentiment classification. These findings have important implications in the context of public policy. Naïve Bayes-based sentiment analysis can be utilized to monitor public opinion and perceptions of government policies in real-time through social media, enabling faster, more accurate, and data-driven policy-making.

Suggestions

Based on the research results, the Naïve Bayes sentiment analysis model is recommended not only for research, but also for implementation in real applications. This application can be developed as an automated public opinion monitoring system integrated with social media data and capable of real-time sentiment classification. This solution has the potential to help government agencies, civil society organizations, and policy research institutions understand public perceptions, evaluate policy impacts, and respond to social issues in a faster and data-driven manner.

ACKNOWLEDGEMENTS

he authors would like to thank all those who have supported this research, especially the supervisor, colleagues for their valuable input, as well as open data contributors and tool developers such as Python, Scikit-learn, and Streamlit who have been very helpful in carrying out the research.

REFERENCES

Alrasyid, H., Homaidi, A., Kom, M., & Fatah, Z. (2024). *Comparison Support Vector Machine and Random Forest Algorithms in Detect Diabetes* (Vol. 1, Issue 1).

Amri, S. (2020). *Information Science and Library Perbandingan Kerangka Model Klasifikasi untuk Pemilihan Metode Kontrasepsi dengan Pendekatan CRIPS-DM Info Artikel _____ e-ISSN*. *1*(1), 14–23. https://doi.org/10.26623/jisl.v1i1

Arsi, P., & Waluyo, R. (2021). *ANALISIS SENTIMEN WACANA PEMINDAHAN IBU KOTA INDONESIA MENGGUNAKAN ALGORITMA SUPPORT VECTOR MACHINE (SVM)*. *8*(1), 147–156. https://doi.org/10.25126/jtiik.202183944

Ayu Syafarina, G., & Kalimantan Muhammad Arsyad Al Banjari Banjarmasin, I. (2023). Implementasi Framework Streamlit Sebagai Prediksi Harga Jual Rumah Dengan Linear Regresi. *Metik Jurnal*, *7*, 2023. https://doi.org/10.47002/metik.v7i2.680

Cholifah Sastya, N., & Nugraha, D. I. (2023). *Penerapan Metode CRISP-DM dalam Menganalisis Data untuk Menentukan Customer Behavior di MeatSolution*. http://ejournal.unis.ac.id/index.php/UNISTEK

Darwis, D., Siskawati, N., & Abidin, Z. (2021a). Penerapan Algoritma Naive Bayes untuk Analisis Sentimen Review Data Twitter BMKG Nasional. *Jurnal Tekno Kompak*, *15*(1).

Darwis, D., Siskawati, N., & Abidin, Z. (2021b). Penerapan Algoritma Naive Bayes untuk Analisis Sentimen Review Data Twitter BMKG Nasional. *Jurnal Tekno Kompak*, *15*(1).

Dwiramadhan, F., Wahyuddin, M. I., & Hidayatullah, D. (2022). Sistem Pakar Diagnosa Penyakit Kulit Kucing Menggunakan Metode Naive Bayes Berbasis Web. *Jurnal Teknologi Informasi Dan Komunikasi)*, *6*(3), 2022. https://doi.org/10.35870/jti

Felicia Watratan, A., Puspita, A. B., Moeis, D., Informasi, S., & Profesional Makassar, S. (2020). Implementasi Algoritma Naive Bayes Untuk Memprediksi Tingkat Penyebaran Covid-19 Di Indonesia. In *JOURNAL OF APPLIED COMPUTER SCIENCE AND TECHNOLOGY (JACOST)* (Vol. 1, Issue 1). http://journal.isas.or.id/index.php/JACOST

Fitriani, M., Nama, G. F., & Mardiana, M. (2022). Implementasi Association Rule Dengan Algoritma Apriori Pada Data Peminjaman Buku UPT Perpustakaan Universitas Lampung Menggunakan Metodologi CRISP-DM. *Jurnal Informatika Dan Teknik Elektro Terapan*, *10*(1). https://doi.org/10.23960/jitet.v10i1.2263

Fitriyah, N., Warsito, B., Asih, D., & Maruddani, I. (2020). ANALISIS SENTIMEN GOJEK PADA MEDIA SOSIAL TWITTER DENGAN KLASIFIKASI SUPPORT VECTOR MACHINE (SVM). *JURNAL GAUSSIAN*, *9*(3), 376–390. https://ejournal3.undip.ac.id/index.php/gaussian/

Gaffar, A. W. M., Halis, A. M., Purnawansyah, P., & Jabir, S. R. (2024). Penerapan Algoritma Support Vector Machine untuk Klasifikasi Stunting pada Balita di Kabupaten Enrekang. *Jurnal Minfo Polgan*, *13*(1), 286–292. https://doi.org/10.33395/jmp.v13i1.13620

Kinerja Naïve Bayes, P., Supian, A., Tri Revaldo, B., Marhadi, N., & Efrizoni, L. (2024). *Acuan Supian Perbandingan Kinerja Naïve Bayes dan SVM pada Analisis Sentimen Twitter Ibukota Nusantara*. https://github.com/syenirasheila/Sentiment-Analysis-IKN-

Miftahusalam, A., Febby Nuraini, A., Khoirunisa, A. A., & Pratiwi, H. (2022). Perbandingan Algoritma Random Forest, Naïve Bayes, dan Support Vector Machine Pada Analisis Sentimen Twitter Mengenai Opini Masyarakat Terhadap Penghapusan Tenaga Honorer. *Seminar Nasional Official Statistics 2022*, 563. https://prosiding.stis.ac.id/index.php/semnasoffstat/article/view/1410

Rani Yunita, M. K. (2023). IJCS_12_5+Yunita. *Indonesian Journal of Computer Science*.

Rihan Maulana, Apriade Voutama, & Taufik Ridwan. (2023). lukman,+609-Analisis+Sentimen+Ulasan+Aplikasi+MyPertamina+pada+Google+Play+Store+menggunakan+Algoritma+NBC. *Jurnal Teknologi Terpadu*, *09*.

Rosyida, T., Putro, H. P., & Wahyono, H. (2023). ANALISIS SENTIMEN TERHADAP PILPRES 2024 BERDASARKAN OPINI DARI TWITTER MENGGUNAKAN NAÏVE BAYES DAN SVM. *Teknokris*, *26*. www.apjii.or.id

Singgalen, Y. A. (2023). Analisis Sentimen Pengunjung Pulau Komodo dan Pulau Rinca di Website Tripadvisor Berbasis CRISP-DM. *Journal of Information System Research (JOSH)*, *4*(2), 614–625. https://doi.org/10.47065/josh.v4i2.2999

Suryati, E., Ari Aldino, A., Penulis Korespondensi, N., & Suryati Submitted, E. (2023). *Analisis Sentimen Transportasi Online Menggunakan Ekstraksi Fitur Model Word2vec Text Embedding Dan Algoritma Support Vector Machine (SVM)*. *4*(1), 96–106. https://doi.org/10.33365/jtsi.v4i1.2445

Wijiyanto, W., Pradana, A. I., Sopingi, S., & Atina, V. (2024). Teknik K-Fold Cross Validation untuk Mengevaluasi Kinerja Mahasiswa. *Jurnal Algoritma*, *21*(1). https://doi.org/10.33364/algoritma/v.21-1.1618