Comparison of SVM & Naïve Bayes Methods in Sentiment Analysis of Electric Vehicle Subsidy Policy Based on X Data

IWD Wiguna[1*], DV Waas[1], IKAG Wiguna[1], ML Radhitya[1]

[1]*Program Studi Teknik Informatika, Institut Bisnis dan Teknologi Indonesia, Jl. Tukad Pakerisan No.97, Panjer, South Denpasar, Denpasar City, Bali, Indonesia 80225*

[*]*Email: darmatohpati3@gmail.com*

Article Information:

Received:
03 May 2024

Received in revised form:
25 May 2024

Accepted:
24 June 2024

Volume 6, Issue 1, June 2024
pp. 23 – 30

http://dx.doi.org/10.23960/jesr.v6i1.158

Abstract

The policy of subsidizing electric vehicles has become a widely discussed issue on social media platform X. The Indonesian government's provision of electric vehicle subsidies aims to stimulate higher adoption of electric vehicles, aiming to mitigate air pollution. However, electric vehicle subsidies continue to elicit both support and opposition among the public. Social media platform X has a wealth of data suitable for text mining, particularly concerning the current hot topic of electric vehicle subsidies. This research aims to compare the performance of the Support Vector Machine (SVM) and Naïve Bayes methods in conducting sentiment analysis on discussions related to the electric vehicle subsidy policy on social media platform X. The testing technique involves using 20% of the total dataset, comprising 5553 data points, and employing 10-fold cross-validation. The 20% test data results indicate that the Support Vector Machine (SVM) method's confusion matrix performance is superior, with the highest values achieved using the RBF kernel: accuracy 83.02%, precision 84.61%, and recall 83.02%. In the performance evaluation testing with 10-fold cross-validation, the SVM method outperforms, especially the RBF kernel, yielding an average accuracy of 82.88% over ten iterations.

Keywords: *Electric Vehicle Subsidy Policy; Sentiment Analysis; SVM; Naïve Bayes; Social Media X*

I. INTRODUCTION

MOTORIZED vehicles are an integral part of daily life in Indonesia. The number of motor vehicle ownership in Indonesia undergoes constant changes. As of May 2023, the total number of motor vehicles in Indonesia has reached 154 million units [1]. Motorized vehicles contribute significantly to carbon emissions and air pollution. Air pollution caused by motorized vehicles includes carbon monoxide (CO), various hydrocarbons, sulfur, various nitrogen oxides (NOx), and particulate matter [2]. Indonesia ranks 17th out of 118 countries in the world for air quality, indicating poor air quality [3].

The government is making efforts to shift from traditional fuel vehicles to electric vehicles. The Indonesian government has issued regulations related to electric vehicles to drive the automotive industry towards electrification. Presidential Regulation Number 55 of 2019 on the Acceleration of Battery-Based Electric Motor Vehicle Programs reflects the government's commitment to reducing greenhouse gas emissions [4]. To expedite the electric vehicle program, the government has issued policies outlined in Minister of Industry Regulation Number 6 of 2023 regarding Guidelines for Government Assistance for the Purchase of Two-Wheeled Battery-Based Electric Motor Vehicles [5].

Despite existing regulations and government plans, the introduction of this subsidy policy has sparked both support and opposition from the public. People can express their opinions through various social media platforms, including X, formerly known as Twitter. X is known for its rapid dissemination of user experiences and easy sharing of news related to trending issues. Additionally, X is one of the most popular social media platforms in Indonesia.

The Naive Bayes method is a classification method based on probabilities to predict future probabilities. Naive Bayes is popular for data mining due to its fast processing time [6]. It can be easily implemented with

relatively simple data structures and is highly effective. Another widely used method in research is the Support Vector Machine (SVM). SVM is a popular classification method that performs well in various domains. SVM can identify a hyperplane that separates different classes, optimizing results and maximizing the distance between data points and the hyperplane [7]. SVM learning involves using pairs of input and output data as desired targets.

Therefore, this research compares the Naive Bayes and SVM methods for sentiment analysis related to electric vehicle subsidy policies. It compares these methods because they have respective advantages, as they can perform optimally even when trained with a small amount of data. [8][9]. Sentiment analysis is a process of analyzing a digital text to determine the emotional tone of the message, whether positive, negative, or neutral. Keywords used in the research include "subsidi kendaraan listrik" (electric vehicle subsidies), "subsidi mobil listrik (electric car subsidies)" and "subsidi motor listrik (electric motorbike subsidies)" with data collection conducted from May to August 2023. The collected data from the three keywords amount to 9003, which is then preprocessed into 4229 data. The data is labeled into three classes, resulting in 1590 positive, 1851 negative, and 858 neutral data. SMOTE up-sampling is performed to avoid imbalanced data, resulting in a balanced dataset of 1851 data for each class.

This research analyzes sentiment by comparing the Naive Bayes and SVM methods on 20% of test data with a balanced number of classes. Model evaluation is performed with 10-fold cross-validation for both methods to avoid overfitting. This comparison determines which method performs better in analyzing sentiment regarding electric vehicle subsidy policies.

II. MATERIALS AND METHODS

A. Research Stages

Data was collected using the Python programming language run on Google Collaboratory in the form of X tweets. The Python programming language uses the tweet-harvest library. There are 3 keywords used, namely "subsidi kendaraan listrik", "subsidi motor listrik", and "subsidi mobil listrik". Data collection was carried out from 1 May 2023 to 31 August 2023.

B. Text Preprocessing

Text Preprocessing is transforming the format of textual data into a more structured form by eliminating unnecessary data, making it easier for the system to process. Preprocessing is crucial in creating sentiment analysis models, especially when the research subject is social media containing unstructured textual data that can cause disturbances [10]. The following are the stages in text preprocessing:

a. Cleaning is a stage to clean the dataset from punctuation, symbols, numbers, URLs, and hashtags. Text on platform X often contains non-alphanumeric characters and links that do not provide important information in the analysis process.
b. Lowercasing is a process of standardizing characters in the document to lowercase to avoid unnecessary casing differences.
c. Stopword Removal: Tweet data often contains unimportant words that can make it less effective in the analysis process. In the Stopword Removal stage, actions such as removing words without significant meaning are taken.
d. Normalization is a process of changing or correcting abbreviated words to the same words according to the official Indonesian dictionary (KBBI).
e. Stemming is the next step in text preprocessing, reducing the number of indices in data so derivative words return to their base form.
f. Tokenizing separates text into words, phrases, symbols, or other meaningful elements called tokens.

C. Labeling Dataset.

Labeling the dataset categorizes data into several sentiment categories used in the research. Labeling automatically uses the Indonesia Sentiment Lexicon dataset to classify tweets into positive, negative, and neutral sentiment categories. [11]. If the number of positive words is greater than the number of negative words, the tweet will be labeled as positive. If the number of negative words is greater than the number of positive words, the tweet will be labeled as negative. The tweet will be labeled as neutral if the number of positive and negative words is equal.

D. Word Weighting TF-IDF.

Word Weighting is a process of assigning weight values to a word based on its frequency of occurrence. Word weighting using TF-IDF (Term Frequency-Inverse Document Frequency) can identify infrequently occurring words, providing relevant information about the importance of words while disregarding common words that do not contribute significantly to sentiment analysis [8].

$$TF\ (d,t) = f(d,t) \quad (1)$$

$$IDF\ (t) = 1 + log\left(\frac{Nd}{df(t)}\right) \quad (2)$$

$$IDF\ (t) = TF(d,t).IDF(t) \quad (3)$$

Where:

$f(d,t)$ = Frequency of the term *(t)* appearing in the document *(d)*
Nd = Total number of documents
$df(t)$ = Number of documents containing the term *(t)*

E. SMOTE Up Sampling.

The labeling results of the dataset have produced an imbalance in the number of data classes. The quantity of data between positive, negative, and neutral classes is significantly different, which can lead to varying precision results for each class when applied to classification methods [12]. To address this issue, SMOTE (Synthetic Minority Over-sampling Technique) Up-Sampling is implemented to handle imbalanced data.

F. Method Implementation.

The sentiment classification process involves using the Naïve Bayes Classifier and SVM algorithms. In the SVM algorithm, four kernels will be utilized: Linear kernel, Polynomial kernel, RBF (Radial Basic Function) kernel, and Sigmoid kernel. The Naïve Bayes Classifier algorithm detects knowledge or patterns of similarity in characteristics within a specific group or class. In performing classification, the Naïve Bayes method treats features independently [13]. The Naïve Bayes Classifier algorithm has the following calculation.

$$P(C|X) = \frac{P(X|C)P(C)}{P(X)} \quad (4)$$

Where:
X = Data with an unknown class
C = Hypothesis that data *X* belongs to a specific class
$P(C|X)$ = Probability of the hypothesis given the condition
$P(C)$ = Probability of the hypothesis
$P(X|C)$ = Probability given the condition on the hypothesis
$P(X)$ = Probability of *X*

The SVM method classifies by examining interactions between features up to a certain level. SVM can identify a hyperplane that separates different classes, optimize its results, and maximize the distance between the nearest data points and the hyperplane or decision boundary. SVM is divided into two types: linear SVM and non-linear SVM. The linear SVM hyperplane can be annotated as follows [14]:

$$f(x) = w^T x + b \quad (5)$$

Where:
x = Feature vector of input data
w^T = Transpose of the weight vector
b = Bias or offset value

For non-linear kernels, they are used to handle data that cannot be linearly separated. There are several non-linear kernel functions, including the Gaussian Kernel or Radial Basis Function (RBF), Polynomial kernel, and Sigmoid kernel. The Gaussian (RBF) Kernel is expressed as:

$$K(x,y) = e^2\left(\frac{\|x-y\|^2}{2\sigma^2}\right) \quad (6)$$

Where:
$K(x,y)$ = Kernel value between two vectors
$\|x-y\|^2$ = Euclidean squared distance between vectors *x* and *y*
σ = Width of the Gaussian function

The Polynomial kernel function is expressed as follows:

$$K(x,y) = \tanh(y.(x^T y) + r)^d, y > 0 \quad (7)$$

Where:
$K(x,y)$= Kernel value between two vectors
y = Kernel parameter
r = Bias parameter
d = Polynomial degree

The Sigmoid kernel function is expressed as follows:

$$K(x,y) = \tanh(y.(x^T y) + r) \quad (8)$$

Where:
$K(x,y)$= Kernel value between two vectors
y = Kernel parameter
r = Bias parameter

In this process, there are two stages of implementation methods, as follows:

a. Implementation of SVM and Naïve Bayes train-test split models, where the dataset is divided into an 80% ratio for training data and 20% for test data, with an equal distribution of classes. Subsequently, the model is trained using 80% of the training data and tested using 20% of the test data.
b. Evaluation of SVM and Naïve Bayes methods with 10-fold cross-validation, used to evaluate the model's performance by dividing the dataset into 10 folds. The model is trained on nine folds and tested on one fold. This process is repeated 10 times to obtain a more stable performance estimation [15].

III. RESULTS AND DISCUSSIONS

A. Data Collection Result.

There were 9001 collected tweet data during the data crawling process from May 1, 2023, to August 31, 2023 (**Table 1**). After data processing using Microsoft Excel, the dataset shrank to 4551 tweet data. This reduction occurred because all processed data went through the "remove duplicate" stage, where identical tweets were eliminated, leaving only one unique tweet data. During the data crawling process, many duplicate data were obtained because the author used three keywords: "subsidi kendaraan listrik", "subsidi motor listrik", and "subsidi mobil listrik".

Additionally, the author performed data crawling multiple times due to limitations on how much data could be obtained in one crawling session. This led to the crawling process, sometimes acquiring the same content. Only tweet data will be used for the sentiment analysis process. Therefore, only the data from the "full_text" column is taken as sentiment data.

Table 1. A Collection of Data in the Form of X Tweets

full_text
@FashionIin @KompasTV Nanti polusi udara ribut lagi ya. Ga ngerti sih maksudnya subsidi kendaraan listrik tujuan nya utk apa.
@KompasTV tp getol kasih subsidi buat yg kaya ut beli kendaraan listrik
@mariadi63425147 @yaniarsim Iya juga ya..... LPG 3kg mau dicabut subsidinya yg jelas2 ini sangat vital dan menyentuh sekali buat rakyat kecil disisi lain kendaraan listrik dapat subsidi puluhan juta.... LOGIKANYA DIMANA DIMANA INI PEMERINTAH
Pemerintah mendorong masyarakat untuk melakukan konversi kendaraan BBM menjadi kendaraan listrik dengan memberikan subsidi Rp 7 juta per unit. https://t.co/rD9gjFt1jh
@tempodotco Mau siapapun Dirut Pertaminanya.... niscayalah harga BBM cenderung naik terus dan BBM subsidi lama2 bakal dihapus juga krn duit modal subsidinya (dari APBN) dialihin utk proyek2 lain misal IKN atau kendaraan. listrik.

B. Text Preprocessing

The text preprocessing process is carried out on the data to make it more structured and eliminate unnecessary information in the analysis process (**Table 2**).

A cleaning stage is performed to remove unnecessary information from the dataset. The data obtained, generally in the form of tweets, often contains special characters, links, numbers, and punctuation that must be removed to improve data quality. Next, a lowering case stage is conducted to ensure consistency in the dataset during analysis because uppercase and lowercase letters are considered the same.

A stopword removal stage is implemented using the Indonesian stopword dictionary created by Oswin Rahadiyan Hartono to reduce irrelevant words in the dataset. Non-formal words or tokens in the dataset are normalized to become standardized and identical words. The reference dataset used in the normalization process is the "Kamus Alay" which consists of unique colloquial words or non-standard language [16].

A stemming process is applied to the dataset to transform variations of word forms by converting derivative words into their base form. The stemming process on the dataset is carried out using the Sastrawi library in the Python programming language [17].

Finally, the dataset is broken down into word units or tokens because the sentiment labeling process is done word by word.

Table 2. Preprocessed Dataset

full_text
['polusi', 'udara', 'ribut', 'erti', 'maksud', 'subsidi', 'kendaraan', 'listrik', 'tuju']
['getol', 'kasih', 'subsidi', 'kayak', 'beli', 'kendaraan', 'listrik']
['cabut', 'subsidi', 'vital', 'sentuh', 'rakyat', 'sisi', 'kendaraan', 'listrik', 'subsidi', 'puluh', 'juta', 'logika', 'mana', 'mana', 'pemerintah']
['pemerintah', 'dorong', 'masyarakat', 'konversi', 'kendaraan', 'kendaraan', 'listrik', 'subsidi', 'juta', 'unit']
['dirut', 'pertaminanya', 'niscaya', 'harga', 'cenderung', 'subsidi', 'hapus', 'duit', 'modal', 'subsidi', 'apbn', 'dialihin', 'proyek', 'kendaraan', 'listrik']

C. Labeling Dataset

Every word in the tokenized dataset will be labeled positive, negative, or neutral (**Table 3**). Dataset labeling is used to train the model and serves as a reference during the testing process of the trained model. The dataset labeling algorithm involves assigning labels to each word based on the Indonesian Sentiment Lexicon, which consists of 3409 positive and 6609 negative words in the Indonesian language. The Indonesian Sentiment Lexicon approach surpasses existing basic methods, achieving the highest accuracy of 65.78% in sentiment classification. If a word is in the positive lexicon, its value is set to 1; for the negative lexicon, the value is -1, and if it is in neither, the value is neutral. To assign labels to tweets, the overall label is determined based on the labels of the previously calculated words. If the calculation is greater than 0, the label is positive; if less than 0, the label is negative; and if 0, the label is neutral (**Figure 1**).

Table 3. Labeling Dataset

Tweet Tokenize	Word Weight	Tweet Weight	Labels
['polusi', 'udara', 'ribut', 'erti', 'maksud', 'subsidi', 'kendara', 'listrik', 'tuju']	[-1, -1, 0, 0, 1, 0, 0, 0, -1]	-2	negatif
['getol', 'kasih', 'subsidi', 'kayak', 'beli', 'kendara', 'listrik']	[-1, 1, 0, -1, -1, 0, 0]	-2	negatif
['cabut', 'subsidi', 'vital', 'sentuh', 'rakyat', 'sisi', 'kendara', 'listrik', 'subsidi', 'puluh', 'juta', 'logika', 'mana', 'mana', 'perintah']	[-1, 0, -1, -1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0]	-2	negatif
['perintah', 'dorong', 'masyarakat', 'konversi', 'kendara', 'kendara', 'listrik', 'subsidi', 'juta', 'unit']	[0, 0, 0, 0, 0, 0, 0, 0, 0, 1, -1]	0	Netral
['dirut', 'pertaminanya', 'niscaya', 'harga', 'cenderung', 'subsidi', 'hapus', 'duit', 'modal', 'subsidi', 'apbn', 'dialihin', 'proyek', 'kendaraan', 'listrik']	[0, 0, 0, 1, -1, 0, -1, 0, -1, 0, 0, 0, 0, 0, 0]	-2	negatif

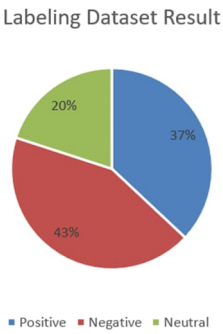

Figure 1. Labeling Dataset Result

D. Word Weighting TF-IDF.

After the dataset has been labeled, the data will be assigned weights to each word using the TF-IDF method, integrating term frequency (TF) and inverse document frequency (IDF). The process involves calculating the TF-IDF representation in vector form to measure the importance of features in the dataset before applying them to the classification algorithm. To compute the weight of each word, the TfidfVectorizer function from the scikit-learn library is used [18].

E. SMOTE Up Sampling.

The labeling results of the previous dataset produced an imbalance in the number of data instances across classes. The quantity of data for positive, negative, and neutral classes differs significantly, leading to varied precision results for each class when applied to a classification method. To address this issue, SMOTE (Synthetic Minority Over-sampling Technique) Upsampling is applied to handle the imbalanced data (**Table 4**).

Table 4. SMOTE Up Sampling

Class	Initial Data	SMOTE Up Sampling
Positive	1590	1851
Negative	1851	1851
Neutral	858	1851
Total	**4299**	**5553**

F. Method Implementation.

Before applying the dataset to SVM and Naïve Bayes algorithms, the dataset will be divided into training and testing data with an 80:20 ratio based on the Pareto principle [19]Dataset splitting is performed to evaluate the classification model's performance by dividing the dataset that has already undergone SMOTE Up Sampling (**Table 5**).

Table 5. Dataset Splitting

Class of Data	Total Data	Train Data (80%)	Test Data (20%)
Positive	1851	1480	371
Negative	1851	1480	371
Neutral	1851	1480	371
Total	5553	4440	1113

The initialization of models in both classification methods is performed using the training data. The SVM method uses four kernel configurations: linear, RBF (Radial Basis Function), Polynomial, and Sigmoid. Meanwhile, The Naïve Bayes model is implemented using the Multinomial Naïve Bayes classification (**Table 6**).

Table 6. Model Functions

Models	Function
svm_model_linear	SVC(kernel='linear')
svm_model_rbf	SVC(kernel='rbf')
svm_model_poly	SVC(kernel='poly', degree=1)
svm_model_sigmoid	SVC(kernel='sigmoid')
naive_bayes_model	MultinomialNB()

a. Implementation of SVM and Naïve Bayes train-test split models

The train-test split method compares the accuracy, precision, and recall between the Naïve Bayes and SVM models for each kernel (**Table 7**) (**Figure 2**).

Table 7. Result of Implementation of SVM and Naïve Bayes train-test split models

Object	Naïve Bayes	SVM			
		Linear	RBF	Polynomial	Sigmoid
Accuracy	71.07%	81.13%	83.02%	81.49%	78.26%
Precision	72.28%	82.18%	84.61%	82.64%	79.08%
Recall	71.07%	81.13%	83.02%	81.49%	78.26%

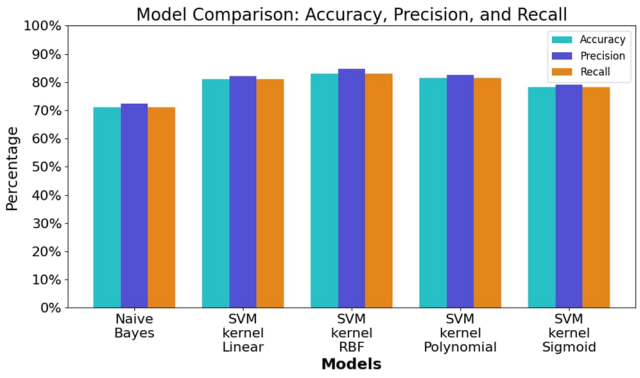

Figure 2. Comparison Accuracy, Precision, and Recall of Models

The comparison results indicate that the SVM method achieves higher accuracy, precision, and recall values than the Naïve Bayes method. The accuracy, precision, and recall values for each SVM kernel consistently outperform those of the Naïve Bayes method. In the SVM model, the RBF kernel shows the highest accuracy, precision, and recall values, namely 83.02%, 84.61%, and 83.02%, respectively. Meanwhile, the Naïve Bayes model obtains 71.07% for accuracy, 72.28% for precision, and 71.07% for recall.

b. Evaluation of SVM and Naïve Bayes methods with 10-fold cross-validation

The comparison of the 10-fold cross-validation method assesses the accuracy in each iteration and the average results between the Naïve Bayes and SVM models for each kernel (**Table 8**) (**Figure 3**).

Table 8. Result Accuracy of SVM and Naïve Bayes methods with 10-fold cross-validation

Iteration	Naïve Bayes	SVM			
		Linear	RBF	Polynomial	Sigmoid
1	65.99%	79.27%	81.08%	79.72%	76.57%
2	68.91%	79.05%	79.95%	76.72%	76.80%
3	68.69%	80.63%	82.43%	81.53%	77.25%
4	71.39%	82.88%	84.45%	82.65%	78.37%
5	70.04%	82.20%	84.00%	81.98%	79.72%
6	74.09%	79.95%	84.23%	80.18%	76.12%
7	67.34%	77.47%	81.75%	77.47%	76.57%
8	70.94%	80.40%	86.29%	81.08%	77.92%
9	72.74%	81.75%	83.10%	82.20%	79.05%
10	67.34%	79.27%	81.53%	79.50%	75.90%

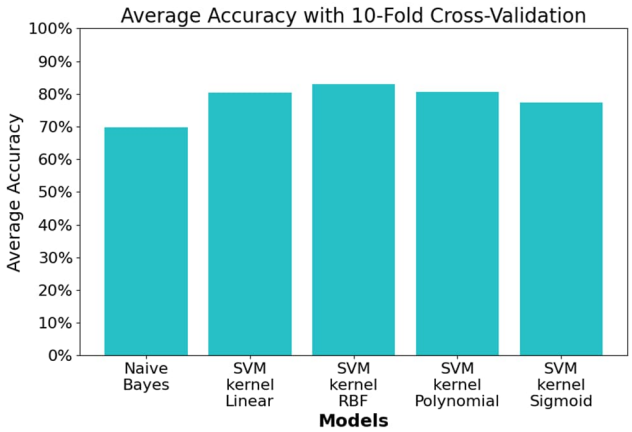

Figure 3. Average Accuracy Every Model with 10-Fold Cross Validation

The comparison results indicate that the SVM method obtains a higher average accuracy value in 10-fold cross-validation than the Naïve Bayes method. The average accuracy values for each SVM kernel consistently surpass those of the Naïve Bayes method. The RBF kernel achieves the highest average accuracy in the SVM model, 82.88%. Meanwhile, the Naïve Bayes method obtains an average accuracy of 69.75%. With the margin maximization principle, SVM yields superior generalization capability on data, reduces the risk of overfitting, and provides more reliable performance in conditions where data features are interrelated.

IV. CONCLUSIONS

Based on the comparison results between the SVM and Naïve Bayes methods, comprehensive evaluation consistently indicates that SVM outperforms Naïve Bayes regarding accuracy, precision, and recall. The SVM model with the RBF kernel consistently achieves the highest values for accuracy, precision, and recall. The evaluation results from the 10-fold cross-validation consistently show higher values for the SVM method than Naïve Bayes, where the average accuracy with the RBF kernel in SVM is higher than that of Naïve Bayes. Text classification methods like Naive Bayes and SVM can be tailored to the characteristics of the data. Naive Bayes is suitable for data with many independent features, while SVM is better for data with complex relationships between features and a strong need for class separation. Method selection is based on the characteristics of the data and the analysis goals.

REFERENCES

[1] Korlantas Polri, "Dashborad ERI," 2023. http://rc.korlantas.polri.go.id:8900/eri2017/laprekappolres.php?kdpolda=18&poldanya=LAMPUNG (accessed May 01, 2023).

[2] A. Toha, P. Purwono, and W. Gata, "Model Prediksi Kualitas Udara dengan Support Vector Machines dengan Optimasi Hyperparameter GridSearch CV," *Bul. Ilm. Sarj. Tek. Elektro*, vol. 4, no. 1, pp. 12–21, 2022.

[3] IQAir, "Informasi Indeks Kualitas Udara (AQI) dan Polusi Udara di Indonesia | IQAir." 2023. [Online]. Available: https://www.iqair.com/id/indonesia

[4] Perpres, "PERPRES No. 55 Tahun 2019 tentang Percepatan Program Kendaraan Bermotor Listrik Berbasis Baterai (Battery Electric Vehicle) untuk Transportasi Jalan [JDIH BPK RI]," *2019*, 2019. https://peraturan.bpk.go.id/Home/Details/116973/perpres-no-55-tahun-2019 (accessed May 01, 2023).

[5] Permenperin, "Pedoman Pemberian Bantuan Pemerintah Untuk Pembelian Kendaraan Bermotor Listrik Berbasis Baterai Roda Dua," 2023.

[6] S. Taheri and M. Mammadov, "Learning the naive Bayes classifier with optimization models," *Int. J. Appl. Math. Comput. Sci.*, vol. 23, no. 4, pp. 787–795, 2013.

[7] D. Gunawan, D. Riana, D. Ardiansyah, F. Akbar, and S. Alfarizi, "Komparasi Algoritma Support Vector Machine Dan Naïve Bayes Dengan Algoritma Genetika Pada Analisis Sentimen Calon Gubernur Jabar 2018-2023. V (1), 135–138." 2020.

[8] R. N. Devita, H. W. Herwanto, and A. P. Wibawa, "Perbandingan kinerja metode naive bayes dan k-nearest neighbor untuk klasifikasi artikel berbahasa indonesia," *J. Teknol. Inf. dan Ilmu Komput*, vol. 5, no. 4, 2018.

[9] D. Suyanto, "Data Mining untuk klasifikasi dan klasterisasi data," *Bandung Inform. Bandung*, 2017.

[10] S. Shevira, I. Made, A. D. Suarjaya, and P. Wira Buana, "Pengaruh Kombinasi dan Urutan Pre-Processing pada Tweets Bahasa Indonesia," *JITTER-Jurnal Ilm. Teknol. dan Komput.*, vol. 3, no. 2, 2022.

[11] F. Koto and G. Y. Rahmaningtyas, "Inset lexicon: Evaluation of a word list for Indonesian sentiment analysis in microblogs," *Proc. 2017 Int. Conf. Asian Lang. Process. IALP 2017*, vol. 2018-Janua, no. December, pp. 391–394, 2018, doi: 10.1109/IALP.2017.8300625.

[12] A. Nurwalikadani, "Implementasi Algoritme Smote Dan Klasifikasi Random Forest Pada Imbalanced Data Metilasi Sequence Protein Lisin," 2022.

[13] R. Amalia, M. A. Bijaksana, and D. Darmantoro, "Negation handling in sentiment classification using rule-based adapted from Indonesian language syntactic for Indonesian text in Twitter," in *Journal of Physics: Conference Series*, IOP Publishing, 2018, p. 12039.

[14] B. Santosa, "Data mining teknik pemanfaatan data untuk keperluan bisnis," *Yogyakarta Graha Ilmu*, vol. 978, no. 979, p. 756, 2007.

[15] W. A. Firmansyach, U. Hayati, and Y. A. Wijaya, "Analisa Terjadinya Overfitting Dan Underfitting Pada Algoritma Naive Bayes Dan Decision Tree Dengan Teknik Cross Validation," *JATI (Jurnal Mhs. Tek. Inform.*, vol. 7, no. 1, pp. 262–269, 2023.

[16] N. Aliyah Salsabila, Y. Ardhito Winatmoko, A.

Akbar Septiandri, and A. Jamal, "Colloquial Indonesian Lexicon," in *2018 International Conference on Asian Language Processing (IALP)*, 2018, pp. 226–229. doi: 10.1109/IALP.2018.8629151.

[17] M. A. Rosid, A. S. Fitrani, I. R. I. Astutik, N. I. Mulloh, and H. A. Gozali, "Improving text preprocessing for student complaint document classification using sastrawi," in *IOP Conference Series: Materials Science and Engineering*, IOP Publishing, 2020, p. 12017.

[18] V. Kumar and B. Subba, "A TfidfVectorizer and SVM based sentiment analysis framework for text data corpus," in *2020 national conference on communications (NCC)*, IEEE, 2020, pp. 1–6.

[19] C. P. Yanti, N. Wayan, E. Agustini, N. Luh, W. Sri, and R. Ginantra, "Perbandingan Metode K-NN Dan Metode Random Forest Untuk Analisis Sentimen pada Tweet Isu Minyak Goreng di Indonesia," vol. 7, no. April, pp. 756–765, 2023, doi: 10.30865/mib.v7i2.5900.