

Decision Tree for Sentiment Analysis of Facebook Social Media Posts Related to Traffic Congestion in Palembang City

Sukemi ^{1*}, Ahmad Fali Oklilas ^{2**}, Hatta Efrizal ^{3*}

* Computer Systems, Faculty of Computer Science, Universitas Sriwijaya

** Doctoral Program in Engineering Science, Faculty of Engineering, Universitas Sriwijaya
sukemi@ilkom.unsri.ac.id ¹, fali@ilkom.unsri.ac.id ², hattaef@gmail.com ³

Article Info

Article history:

Received 2026-03-04

Revised 2026-04-26

Accepted 2026-05-05

Keyword:

*Decision Tree,
Sentiment Analysis,
Traffic Congestion,
Facebook Social Media.*

ABSTRACT

This study aims to analyze public perception of traffic congestion in Palembang City through sentiment analysis on Facebook using the Decision Tree algorithm. Data were collected from public comments related to traffic over 32 months using web scraping techniques. Text data were processed through preprocessing stages including case folding, tokenization, stemming, and stopword removal, followed by TF-IDF feature extraction with unigram representation. The model classifies sentiments into positive, negative, and neutral categories. The results show an accuracy of 90.42%. However, the model tends to perform better on the neutral class, influenced by imbalanced data distribution. Therefore, evaluation metrics such as precision and recall are also considered to provide a more comprehensive analysis.



This is an open access article under the CC-BY-SA license.

I. INTRODUCTION

Traffic congestion has become a critical issue affecting urban mobility, economic productivity, and public service quality in many developing cities, including Palembang, Indonesia. The rapid increase in vehicle ownership, combined with limited transportation infrastructure and inefficient traffic management, contributes to daily congestion problems that impact public activities and accessibility [1], [2]. Infrastructure development and transportation service improvements are essential for supporting public mobility and reducing operational challenges in urban areas. However, the complexity of congestion issues requires analytical approaches that incorporate not only quantitative traffic data but also public perception and social response.

Social media platforms, particularly Facebook, have become valuable sources of real-time public opinion, where users frequently express complaints, experiences, and suggestions regarding traffic conditions [3], [4]. Sentiment analysis enables the extraction and classification of opinions into positive, negative, and neutral categories, providing insight into the emotional responses of the public. Sentiment analysis techniques have been investigated using various ML methods, such as Naïve Bayes, support vector machine (SVM), random forest, and decision tree [5]. Many previous studies have shown that the choice of techniques and

algorithms plays a significant role in determining the quality of classification results in sentiment analysis, particularly in the realm of Natural Language Processing (NLP). The literature explains that the accuracy of the method used and the model's ability to process the complexity of natural language influence the success rate of sentiment analysis [6], [7], [8]. Machine learning algorithms such as decision trees and random forests have been widely used to determine the best classification process performance. Several studies have shown that decision trees can produce classification results that can be directly evaluated using a confusion matrix [9]. Research findings show that sentiment analysis can effectively support decision-making by identifying trends in public sentiment and dominant issues in transportation contexts [10].

Decision trees are widely used in classification tasks, especially in text sentiment analysis, because of their simplicity, interpretability, and high performance [11]. However, research focusing on the application of sentiment analysis to traffic congestion perception in Indonesian cities, particularly Palembang, is still limited. Therefore, this study aims to analyze public sentiment related to traffic congestion in Palembang using the decision tree algorithm to classify Facebook comments and posts. This study provides insights into sentiment distribution, evaluates decision tree

performance in sentiment classification, and delivers recommendations that support data-driven transportation planning and policy development.

II. METHOD

A. Data collection technique

Data collection and retrieval were performed using an application programming interface (API) in previous research. Data will be retrieved from platform X using Tweet Harvest, a tool used to retrieve data from searches on platform X based on keywords, language, and the time span of an issue [12], [13]. However, in this study, it was used for data crawling, using the Apify Console to collect Facebook social media comment data using an actor called the Facebook Comments Scraper. The URL of the Facebook social media post is then entered to scrape the comments, and then the actor is run, resulting in the raw data that can be exported as a Comma Separated Values (CSV) file. Figure 1 shows the flow of data collection.

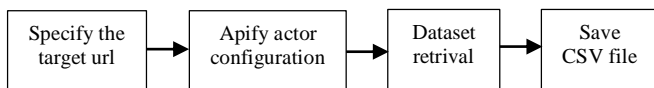


Figure 1. Flow of Text Data Collection

The raw data obtained from web scraping will be used as a basis for sentiment analysis. Using Facebook as a data source was deemed relevant because the platform has many active users in Indonesia and is frequently used by the public to express opinions on public issues, including traffic congestion. The collected data were then stored in CSV format to facilitate further processing using Python. Each row of data contains a comment text column, the upload time, and the anonymous account's identity. This data collection process has been widely applied in research on social media-based sentiment analysis.

B. Preprocessing Data Text and Filtering

The preprocessing stage aims to prepare text data for optimal processing by machine learning algorithms. Reducing noise in text is done in the pre-processing stage before the classification process. This stage follows related research, which states that good preprocessing improves model accuracy in sentiment analysis [12]. Raw web scraping data typically contains many irrelevant characters, such as punctuation, emojis, links, and meaningless words. Therefore, a text cleaning process is performed to produce cleaner and more representative data before sentiment analysis. The preprocessing flow can be seen in Figure 2.

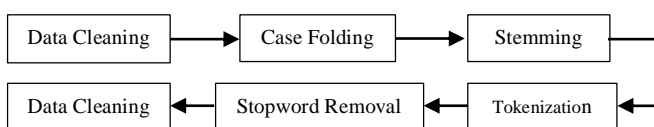


Figure 2. Text Data Preprocessing Flow

In this study, the preprocessing process consisted of several steps: case folding, tokenization, stopwords removal, stemming, and normalization. Case folding standardizes lowercase letters, while tokenization breaks sentences into single words to facilitate analysis [14]. The next step, stopwords removal, removes common words that have no impact on meaning, such as "the," "and," or "in".

Stemming is then performed to return each word to its base form. For example, "walk" becomes "road." This process is adapted to Indonesian language rules to ensure more accurate analysis results [15]. Furthermore, normalization is performed to standardize variations in spelling of words with similar meanings, such as "jametnya," and "jamet."

The filtering stage refines the preprocessing results so that only relevant data worthy of analysis is retained. This process eliminates duplicate text, empty comments, and data containing non-text elements, such as links, numbers, or special characters. With filtering, dataset quality can be improved because only clean data will be used in the weighting and classification processes.

Filtering also includes selecting text length, as comments that are too short or too long can bias the results of the sentiment analysis. The authors emphasize the importance of data filtering, as irrelevant text can reduce the accuracy of machine learning models. Furthermore, contextual appropriateness checks were performed to ensure that each comment addressed traffic congestion in the city of Palembang.

C. Splitting Data

In previous research, the data was divided into two parts, namely training data and testing data, with a ratio of 80% and 20% [16]. However, in this study, the data was divided into three parts: training set, the validation set, and the testing set. This division aims to ensure that the developed model is not only capable of recognizing patterns from the training data but also can be evaluated gradually before being tested using new data that the model has never seen before.

The data is divided using the stratified holdout method, dividing the data into training, testing, and validation sets so that each cluster is represented according to the calculated proportion, ensuring a balanced distribution of features. Implementation: Apply stratified separation within each cluster, separating the data according to the desired ratio of 70% for the training set, 15% for the validation data, and 15% for the testing set [17], [18].

D. TF – IDF

The TF-IDF (Term Frequency – Inverse Document Frequency) weighting stage is performed to convert filtered text data into a numerical representation that can be processed by a classification algorithm. The TF-IDF method functions to give weight to each word based on its level of occurrence in a document and compared with the entire corpus. The more frequently a word appears in a particular document but rarely in others, the higher its TF-IDF weight will be.

$$TF - IDF_{(t,d)} = TF_{(t,d)} \times \log\left(\frac{N}{df(t)}\right) \tag{1}$$

Where tf is the frequency of word t in a document, df is the number of documents containing word t , and N is the total number of documents in the dataset.

The TF value describes how often a word appears in a single document, while the IDF indicates how rarely the word appears across documents. By multiplying these two values, TF-IDF is able to highlight the most representative words in each comment text. This approach is considered effective because it can suppress the influence of common words with no specific meaning.

The topic model approach used is a class-based TF-IDF clustering and variation technique to produce a coherent topic representation. The first step involves document embedding using a pre-trained language model to capture document-level information. The second step involves reducing the dimensionality of the document embedding before clustering semantically similar documents, each representing a different topic. The third step, to anticipate the emergence of a centroid-based perspective, includes a class-based TF-IDF method to extract topic representations from each topic. These three independent steps allow the development of a flexible topic model that can be used in a variety of implementation cases, such as dynamic topic modeling [19].

E. Classification of Decision Tree Algorithms (DTA)

In previous research, the K-Nearest Neighbor (KNN) algorithm determined sentiment based on most KNN using Euclidean Distance, where the model searches for text similarity based on the distance between data. KNN performance depends on selecting the optimal K value for sentiment classification [16]. However, in this research, the decision tree algorithm was used. This algorithm is a supervised learning method used to group data based on attributes and feature values generated from the TF-IDF weighting process. Construct a model in the form of a tree structure in a decision trees, can be seen in Figure 3.

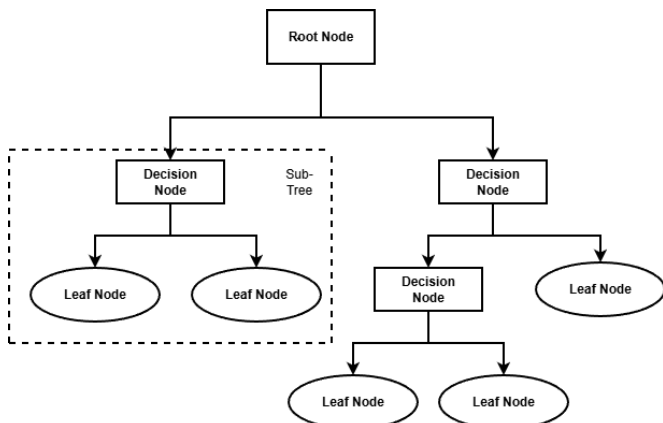


Figure 3. Decision Tree Node for Data Groups from Attributes and TF-IDF Weighting

Each node represents an attribute, a branch represents the attribute test result, and a leaf node represents the sentiment class label.

One of the advantages of decision tree is its ability to produce easily interpretable models. The resulting classification rules can be written in the form of if-then rules, simplifying the analysis of the results. Furthermore, this algorithm performs well on medium-sized datasets and can handle text data that has been converted into numeric formats, such as TF-IDF.

The decision tree model was used to classify public comments into three sentiment categories: positive, negative, and neutral. The training process was conducted using training data, while the test data was used to measure the generalizability of the model. The output of this stage was a sentiment label representing the public perception of traffic congestion in Palembang City.

III. RESULT AND DISCUSSION

1. Data Collection Results

The data for this study were obtained through a web scraping process of public comments on Facebook discussing traffic congestion in Palembang City. Data collection was conducted on several posts from local news pages and community forums that actively discuss road conditions. The web scraping technique was chosen because it is able to automate the process of retrieving large amounts of data from web pages without having to do it manually [20]. This can be seen in Figure 4.

	postTitle	text
0	Hujan 🌧 yang mengguyur kota Palembang malam in...	Kasialan bng .. ttp amngat yohh 🤔🤔
1	Hujan 🌧 yang mengguyur kota Palembang malam in...	semoga hujan membawa berkah
2	Hujan 🌧 yang mengguyur kota Palembang malam in...	Ijin Hadir kk
3	Hujan 🌧 yang mengguyur kota Palembang malam in...	Selamat malam bg 🤗🤗
4	Hujan 🌧 yang mengguyur kota Palembang malam in...	Banjir dk kk
...
7695	Orang Palembang pasti tau dgn Underpass ini , ...	Keren
7696	Orang Palembang pasti tau dgn Underpass ini , ...	Mantap min
7697	macet nian bestie.....	Yani Maryaminami iya Bun...
7698	Tolong pak bawak mobil derek, angkutke galo mo...	Boy Fern Kosongsatu Yuhar enak bikin maksuba
7699	Kota Palembang Saat iniKondisi Jalan Lancar ...	Mar Diana \nAseekk

7700 rows × 2 columns

Figure 4. Facebook Raw Data

The obtained raw data were stored in CSV format. All 32 months of scraped data, from February 2023 to June 2025, was stored in the CSV file. A total of 294 links to posts about traffic in Palembang were collected, with 7,700 comments.

2. Preprocessing Data Text and Filtering Results

The preprocessing and filtering stages were carried out to prepare the collected comment data for processing in sentiment analysis. This process involved several text cleaning steps, including removing punctuation, numbers, URLs, double spaces, and non-alphabetic characters, with the

aim of eliminating elements that do not provide contextual meaning. The preprocessing process in this study consisted of several steps: case folding, tokenization, stopword removal, stemming, and normalization. These steps are shown in Table 1.

TABLE I
EXAMPLE OF TEXT DATA PREROCCESSING

No	Original Text	Tokenized	Normalization
1	didepan RSU Muhamaad Husein min sering macet disitu	[“depan,” “rsu,” “muhamad,” “husein,” “min,” “sering,” “macet,” “situ”]	rsu muhamad husein min macet situ
2	Percuma bae uba sano ubah sini mase macet tulah kalo jam sekolah kecuali agak malem agak sepi jalan itu	[“bae,” “uba,” “sano,” “ubah,” “mase,” “macet,” “tulah,” “kalo,” “jam,” “sekolah,” “kecuali,” “malem,” “sepi,” “jalan”]	bae uba sano ubah mase macet tulah kalo jam sekolah kecuali malam sepi jalan
3	Tiap hri lwtat sini dak pagi dak sore macet trus klu ank2 kumbang sdh keluar smp2 di pinggir jln mobil2 jemputan nyo bebaris SMP separuh jln kecuali hri sabtu	[‘hri’, ‘lwtat’, ‘dak’, ‘pagi’, ‘sore’, ‘macet’, ‘trus’, ‘klu’, ‘ank’, ‘kumbang’, ‘sdh’, ‘smp’, ‘pinggir’, ‘jln’, ‘mobil’, ‘jemput’, ‘nyo’, ‘bebar’, ‘smp’, ‘paruh’, ‘jln’, ‘kecuali’, ‘hri’, ‘sabtu’]	hari lwtat indak pagi indak sore macet terus kalau anak kumbang sudah sampai pinggir jalan mobil jemput nya bebar sampai paruh jalan kecuali hari sabtu
4	Uji wong tambah macet....parkiran kumbang itu harus di relokasi	[‘uji’, ‘macet’, ‘parkir’, ‘kumbang’, ‘relokasi’]	uji macet parkir kumbang relokasi
5	Tiap hari aq lewat sini...pagi tadi masih ado belokan di depan kumbang	[aq, pagi, ado, belok, kumbang]	aku pagi ado belok kumbang

Table 1 shows the results of the tokenization and normalization stages for comments from residents of Palembang regarding traffic congestion. Tokenization produces a list of words ready for use in the weighting process, while normalization converts nonstandard words into more uniform forms without altering the meaning of the sentences.

The filtering stage was conducted after the preprocessing process to ensure that the data used were truly relevant to the research topic, namely, public perceptions of traffic congestion in Palembang City. Comments unrelated to traffic, promotions, spam, or out-of-context conversations were removed from the dataset at this stage. Sentiment results for each sentence where every sentence containing the word "jam" will be included in the negative class and every

sentence containing the word "lancar" will be included in the positive class if the sentence does not contain the words jam and lancar but is still in accordance with the topic of traffic it will be included in the neutral class. This can be seen in Figure 5.

	postTitle	normalization	class
0	Hujan yang mengguyur kota Palembang malam in...	selamat rehat kak Palembang ada guyur hujan smlmn	netral
1	Suasana arus lalu lintas di kota Palembang, Ra...	mantap lancar	positif
2	Suasana arus lalu lintas di kota Palembang, Ra...	mantap bucan moga hasil muas	positif
3	Kepadatan lalu lintas kota Palembang	sudah rame ya	netral
4	Kepadatan lalu lintas kota Palembang	kota Palembang maju aku pindah cipondoh kota L...	netral
...
2356	macet nian bestie.....	apa segalo bucu Palembang juk	netral
2357	macet nian bestie.....	macet banget bun	negatif
2358	Lah sering di posting cuma dak pernah selesai ...	agoes verza benar nian gara ado ogah panti kay...	negatif
2359	Orang Palembang pasti tau dgn Underpass ini ...	mantap min	positif
2360	Tolong pak bawah mobil derek, angkutke galo mo...	boy fern kosongsatu yuhar enak bikin maksuba	positif

Figure 5. Public Perceptions Data Filtering

Of the initial 7,700 comments, 5,339 were removed for irrelevance, leaving 2,361 clean entries for analysis. After the data were declared valid, a manual sentiment labeling process was performed, categorizing them into three categories: positive, negative, and neutral. This classification was based on the expression of opinion in the text, such as words with positive emotional tones ("smooth," "good") or negative ("jamet," "severe"), as well as the overall context of the sentence. With a total of 2,361 rows for the negative class, there are 743 rows for the neutral class, 839 rows for the positive class, and 779 rows for the positive class. So, for negative sentiment, the number of rows is 31.46%, for neutral sentiment, the number of rows is 35.53%, and for positive sentiment, the number of rows is 32.99%. This indicates that the percentage of the number of classes is quite balanced.

3. Data Splitting Results

The data-splitting stage is performed after the filtering and labeling processes are complete. The goal is to divide the dataset into several parts so that the machine learning model can be trained, validated, and tested objectively. In this study, the data is divided into three parts: training_data (70%), validation_data (15%), and testing_data (15%).

Of the total 2,361 valid comment datasets, 1,652 were used as training data, 354 as validation data, and 355 as test data. The data were divided using the stratified holdout method, dividing the data into training, testing, and validation sets so that each cluster was represented according to the calculated proportions, ensuring a balanced feature distribution.

4. TF – IDF Results

The feature extraction process is carried out using the Term Frequency (TF) and Term Frequency-Inverse Document Frequency (TF-IDF) methods. TF values are calculated using CountVectorizer from the scikit-learn library with default parameters. At this stage, the feature representation uses a unigram approach (ngram_range =

(1,1)), where every single word in the document is considered a feature. In addition, stopwords are not removed so that all words in the document are still considered in the weighting process. The following is an example of text data used in the TF-IDF weighting calculation. To calculate the weight of each word in the data, the weighting begins by calculating the TF value, then the IDF value, and then multiplying the TF and IDF values. The equation above can be implemented in the example text data shown in Table 2.

TABLE II
DATA TRAINING

No	Text
1	lancar jaya
2	macet ya bun
3	kangen Palembang
4	faktor dari macet sekarang jemput anak sekolah nak pakai mobil galo kadang plat merah jemput anak sekolah pergi kantor bawa mobil sikok sikok mobil isi orang wajar bae macet yang bawa mobil perempuan pada saman kalau bawa motor nak menang dewek sudah dio salah lawan kenyat jalan pulang tidak ubah macet lebar jalan macet karena banyak yang pakai mobil
5	lemakla tutup lancar nian jalan
:	:
1648	kotah wajar bos ku daerah muara enim macet gara api batu bara mobil batu bara pernah demo masyarakat stop total herman deru kreta api batu barang yang bikin macet enggak respon perintah nya atur nya jangan pengguna jalan kb muara enim gagu pasien sakit tuju rs bunda jm wb simpang belimbing enggak masuk akal
1649	afryl filiang geografis beda orang bilang begitu ya serah nama jalan enggak bawa bawa Palembang kota letak geografis sudah beda sudah enggak bawa nama kota tolol teriak tolol
1650	yang enggak tau aturannya MBL pribadi parkir senak sudah macet salah MBL tkp tuding macet
1651	mantap
1652	mulyadi mul susah karena negra gengsi budaya pegawai bus mana yang tau mobil

In the second sentence in Table 2, "Jamet ya bun" has 3 words. TF calculates the value of each word in the document. For example, the word "jamet" has 1 word, so the TF calculation is as follows:

$$TF = \frac{\text{word count in } d}{\text{total words in } d} = \frac{1}{3} = 0,333$$

So the TF value for the word "jamet" in document 1 is 0.333. Meanwhile, the IDF calculation for the word "jamet" contained in the 10 documents in Table 2 is as follows:

$$idf_t = \ln \frac{D + 1}{Df_t + 1} + 1 = \ln \frac{10 + 1}{9 + 1} + 1 = \ln \left(\frac{11}{10} \right) + 1 = 1,095$$

After the TF and IDF values are obtained, multiply the TF and IDF results as follows:

$$w_{t,d} = 0,333 \times 1,095 = 0,364$$

So to find the TF-IDF value for the word "jamet" is 0.364.

Text data visualization can provide an overview of the frequency of a word's appearance within a document. The more frequently a word appears, the larger the visual text will appear. A visualization of the text data can be seen in Figure 6. below.

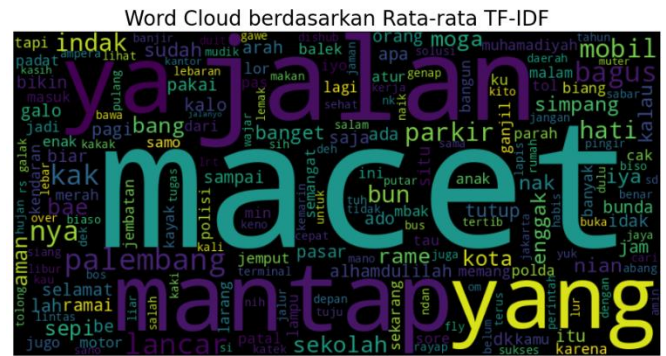


Figure 6. Word Cloud Data Text

This word cloud reflects public conversations about traffic conditions in Palembang. The most prominent words are "jam" and "jalan," confirming that the discussion focuses heavily on congestion and traffic flow. On the other hand, the words "mantap" and "lancar" also appear in relatively large size, indicating positive comments when traffic feels light. Furthermore, words like "rame," "parking," "car," "anak," and "sekolah" indicate links to daily activities, such as congestion caused by school drop-offs or parking issues. Other words like "polis," "junction," "jembatan," and "fly" describe specific traffic points or locations frequently discussed, while "ganjil" and "genap" refer to traffic regulations that have also been highlighted. The presence of the word "palembang" emphasizes the specific regional context, while greetings like "ya," "bu," and "kak" reflect netizens' relaxed and friendly communication style.

5. Decision Tree Algorithm Classification Results

After going through the preprocessing stage, assigning sentiment labels, and dividing the data (split 70%:15%:15%), the model was trained using the 'decision tree algorithm'. This model is saved in the best_fitting_model.pkl file format, which contains the structure and the best parameters of the training results. The parameters used include criterion = gini, max depth = 15, min samples split = 10, and min samples leaf = 5. These settings aim to prevent overfitting and produce a stable model that is able to generalize to new data. Displays a visualization of the decision tree structure. Figure 7. displays a decision tree with max depth = 3.

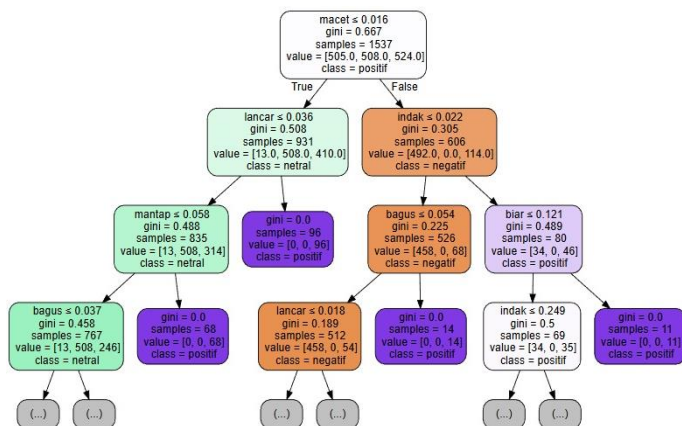


Figure 7. Decision Tree Visualization Max Depth = 3

Visualization of the decision tree structure formed from the model training results. The traffic jam feature is the main separator at the initial node, with 1,652 data samples and the majority of the class being neutral. Next, the branching is divided based on keywords such as "smooth," "good," "steady," "not," and "road," which influence sentiment classification. In general, when the frequency of the word "traffic jam" is lower, public comments tend to be classified as neutral or positive. Conversely, when the word "traffic jam" is more dominant, comments fall more into the negative class. Thus, this decision tree provides a clear picture of the keyword factors that most influence public sentiment toward traffic conditions in Palembang City.

The following summary contains information on the level of accuracy obtained at each stage, the amount of data used, and its percentage of the overall dataset. Table 3 provides a summary of the accuracy levels obtained at each stage, the amount of data used, and their percentages of the total dataset.

TABLE III
SUMMARY OF RESULTS DATA TRAINING, DATA VALIDATION, AND DATA TESTING

No	Data Types	Accuracy (%)	Amount of data	Percentage (%)
1	Training	93,77	1.652	70
2	Validation	90,40	354	15
3	Testing	90,42	355	15

Based on Table 3, it can be seen that the accuracy on the training data reached 93.77% with a total of 1,652 rows of data or approximately 70% of the entire dataset. Meanwhile, the accuracy in the validation stage was 90.40% with 354 rows of data (15%), and the accuracy in the testing stage reached 90.42% with 355 rows of data (15%). The difference in accuracy values between stages was not too significant, indicating that the decision tree model is able to generalize well to new data because the accuracy of each training, validation, and testing is not too far apart, thus categorized as best fitting.

The training data will be used to train the Decision Tree model. Meanwhile, the test data will be used to test the

Decision Tree model based on the learning results using training data with positive, neutral, and negative classes. The confusion matrix for the training data can be seen in Figure 7. below.

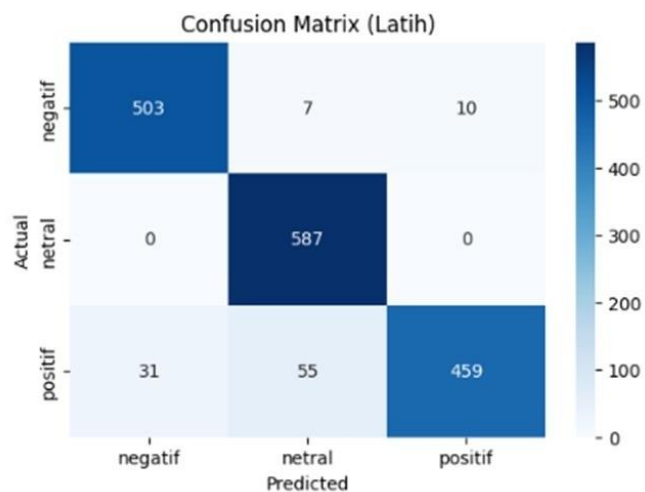


Figure 8. Confusion Matrix Data Training

The model performed very well in both the neutral and negative classes, as evidenced by the fact that almost all data were correctly classified. For the neutral class, all 587 data points were correctly predicted, while for the negative class, 503 data points were correctly classified with only 17 errors. However, for the positive class, the model's performance was still less than optimal, as 86 data points were misclassified (31 were predicted as negative and 55 were predicted as neutral). Overall, the training data accuracy reached 93.77%, and the precision, recall, and f1-score values can be seen in Figure 8 below.

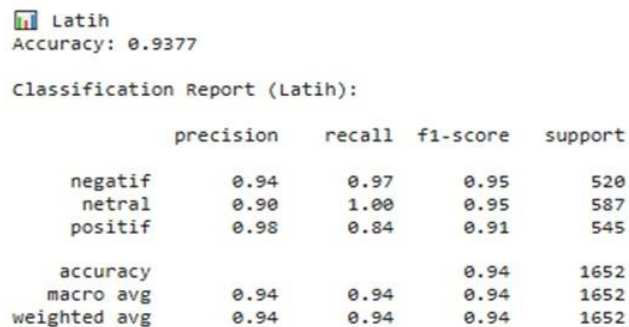


Figure 9. Classification Report Data Training

Precision values for each class. For the negative class it is 94%, neutral 90%, and positive 98%. The recall value of each class. For the negative class it was 97%, neutral 100%, and positive 84%. The f1-score value of each class. For the negative class, it is 95%, neutral 95%, and positive 91%.

Then, the model training data is evaluated on validation data to ensure the model not only performs well on the training data but also understands new data well. The confusion matrix can be seen in Figure 10. Below

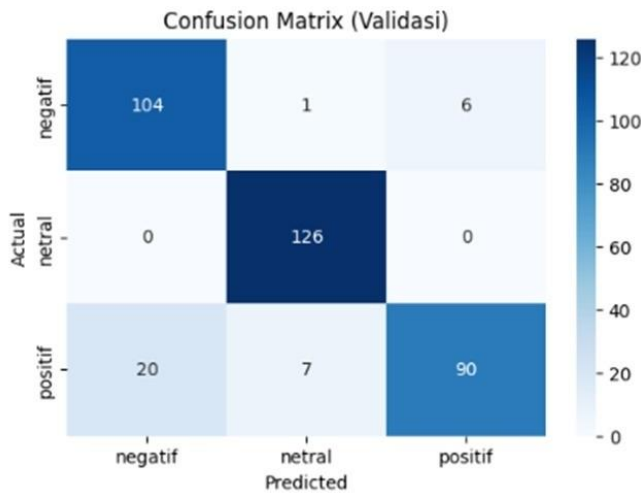


Figure 10 Confusion Matrix Data Validation

The model on the validation data showed excellent performance in both the neutral and negative classes, with a very low error rate. In the neutral class, all 126 data points were correctly predicted, while in the negative class, 104 data points were correctly classified with only 7 errors. Meanwhile, in the positive class, there were still 20 misclassified data points (20 predicted as negative and 7 predicted as neutral). Overall, the model achieved an accuracy of 90.40%, with the precision, recall, and f1-score values shown in Figure 11. below.

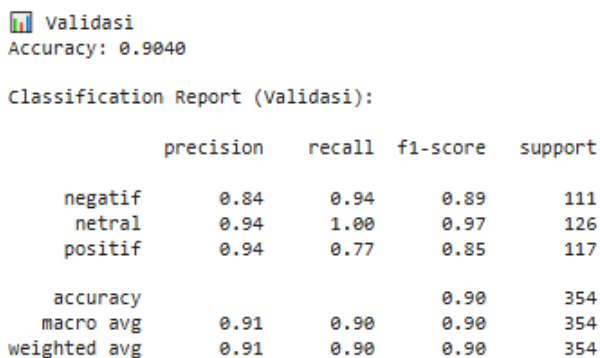


Figure 11 Classification Report Data Validation

Precision values for each class. For the negative class it is 84%, neutral 94%, and positive 94%. Recall values for each class. For the negative class it was 94%, neutral 100%, and positive 77%. The f1-score value of each class. For the negative class, it is 89%, neutral 97%, and positive 85%.

Then the evaluation of the test data can be seen in the confusion matrix in Figure 12. below.

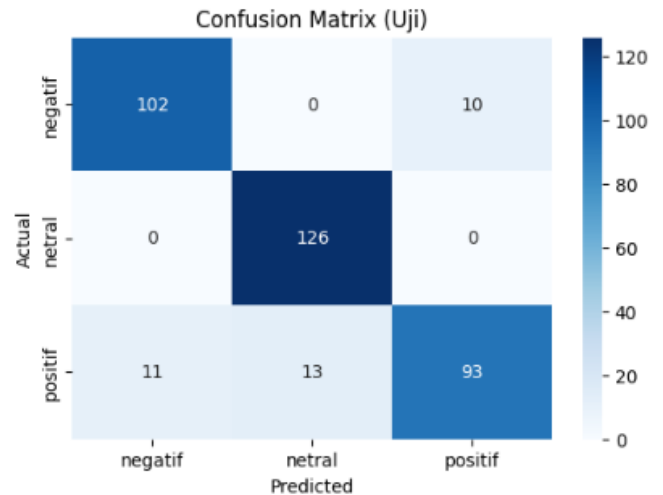


Figure 12 Confusion Matrix Data Testing

The model performed very well on the test data in both the neutral and negative classes, with very few errors. In the neutral class, all 126 data points were correctly predicted, while in the negative class, 102 data points were correctly classified with only 10 errors. However, in the positive class, there were still 24 misclassified data points (11 predicted as negative and 13 predicted as neutral). Overall, the model achieved an accuracy of 90.42%, with the precision, recall, and f1-score values shown in Figure 13. below.

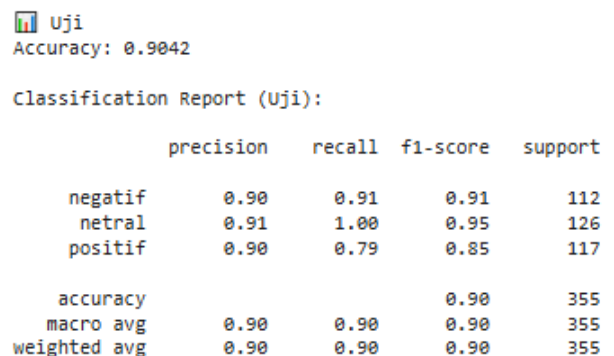


Figure 13 Classification Report Data Testing

Precision values for each class. For the negative class it is 90%, neutral 91%, and positive 90%. Recall values for each class. For the negative class it was 91%, neutral 100%, and positive 79%. The f1-score value of each class. For the negative class, it is 91%, neutral 95%, and positive 85%.

6. Analysis of the Correctness of Decision Tree Predictions on Testing Data

Next, each comment was manually reviewed to verify the accuracy of the decision tree classification results on the testing data, examining the overall meaning of the sentence and the previous post. This was done because some comments contained the word "smooth," but were actually discussing efforts to address traffic congestion in a particular area.

Therefore, a "Previous Post" column has been added to Table 4 for context.

TABLE IV
SENTIMENT CLASSIFICATION ACCURACY ANALYSIS BASED ON INITIAL POSTS ON TESTING DATA

No	Previous Post	Comment	Original Label	Prediksi DT	Label Analysis
1	Sepi Jalan Palembang Saat Tahun Baru	Rindy Rizka Ambarini samo bae ndi mngkin yang jual lagi jalan bangun siang	Netral	Netral	Positif
2	Masalah macet sering diposting tapi tak selesai	Kumbang macet jugo maklum sekolah anak kayo galo anter jemput pakai mobil	Negatif	Negatif	Negatif
3	Macet arah Kambang Iwak, parkir liar di badan jalan	Yola galo sano tukang palak kedok jukir indak tau siapa nian backingnyo jembatan ampera jugo parkir motor	Positif	Positif	Negatif
4	Jalan Sepi Saat Motoran Sendiri	Ya saman alangke seramnyo mas jon daktikduk jantung hehe	Positif	Netral	Positif
5	Setelah Hujan, Jalan Padat Merayap	Masih pagi lah padat rayap ca ya apolagi lah siang terbang be men cak daripada rayap lambat sampenyo	Netral	Netral	Netral

Based on Table 4 decision tree predicts as many as 355 rows of testing/test data, of which 113 are negative, 103 are positive, 139 are neutral. However, based on the analysis checked per row according to the context of the initial post for negative, there are no wrong predictions, but the number is less than 240 rows, so there is a prediction error of $127/240 = 52.91\%$ and the correct number is $113/240 = 47.08\%$. for the number of correct positive predictions $59/103 = 57.28\%$, then for the number of incorrect positive predictions $44/103 = 42.71\%$, for the number of correct neutral predictions $35/139 = 25.17\%$, for the number of incorrect neutral predictions $104/139 = 74.82\%$. This means that prediction errors from the decision tree, then negative sentiment is more on social media, the number of correct negative sentiments with a percentage of 43.29%, but for correct positives 57.28%, and correct neutrals 25.17% this means the truth of negative, neutral, and positive predictions is still low because it is 60% and below.

The following is a confusion matrix from the manual analysis calculations of the test data.

TABLE V
CONFUSION MATRIX MANUAL CALCULATION OF TEST DATA

Actual	negative	113	0	0
	neutral	85	35	19
	positive	42	2	59
		negative	neutral	positive
Predicted				

Based on Table 5. There are 355 test data based on predictions from decision tree, namely, 113 negative data, 139 neutral data, and 103 positive data. Where the TP data obtained for negative is 113, neutral is 35, and positive is 59. Meanwhile, for neutral data predicted as positive (FN) there are 19, and neutral data predicted as negative (FN) there are 85, while positive data predicted as negative (FN) there are 42 and predicted as neutral (FN) there are 2. The following is based on Table 5. Description of the data obtained in the confusion matrix.

Total Actual (horizontal) = 113, 139, 103

Total Predicted (vertical) = 240, 37, 78

True Positive, True Negative = $113 + 35 + 59 = 187$

False Positive, False Negative = $127 + 2 + 19 = 148$

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

$$\text{Accuracy} = \frac{187}{187 + 148} = 0,5582 \times 100\% = 55,82\%$$

Then, the precision is obtained from TP divided by (TP + FP).

$$\text{Precision} = \frac{TP}{TP + FP} \quad (3)$$

TABLE VI
MANUAL PRECISION CALCULATION TEST DATA

Accuracy/Prediction	Negative	Neutral	Positive
TP	113	35	59
TP + FP	240	37	78
Precision	0,47	0,94	0,75

Table 6 presents the precision values for each class. For the negative class, it was 47%, for the neutral class, it was 94%, and for the positive class, it was 75%.

Then, recall is obtained from TP divided by (TP + FN).

$$\text{Recall} = \frac{TP}{TP + FN} \quad (4)$$

TABLE VII
MANUAL RECALL CALCULATION TEST DATA

Accuracy/Prediction	Negative	Neutral	Positive
TP	113	35	59
TP + FN	113	139	103
Precision	1.00	0,25	0,57

The recall values for each class are shown in Table 7. The negative class was 100%, neutral class 25%, and the positive class 57%.

Finally, the f1-score is obtained from the result of 2 times (recall times precision) divided by (recall plus precision).

$$F1_{score} = 2 \times \left(\frac{Precision \times Recall}{Precision + Recall} \right) \quad (5)$$

TABLE VIII
MANUAL F1-SCORE CALCULATION OF TEST DATA

Accuracy/Precision	Negative	Neutral	Positive
(Precision x recall)	(0,47 x 1,00) = 0,47	(0,94 x 0,25) = 0,23	(0,75 x 0,57) = 0,42
(Precision + recall)	(0,47 + 1,00) = 1,47	(0,94 + 0,25) = 1,23	(0,75 + 0,57) = 1,42
F1-Score	2 x 0,31 = 0,62	2 x 0,18 = 0,36	2 x 0,29 = 0,58

The recall values for each class are shown in Table 8. The negative, neutral, and positive classes accounted for 62%, 36%, and 58%, respectively.

This research analysis reveals significant differences at several stages of the process compared to related research. In the crawling stage, previous studies generally used different social media data sources, such as Twitter or Instagram, which are characterized by shorter content and limited to quick conversations. Meanwhile, this study uses Facebook comment data, which has a longer and more interactive discussion structure, providing richer and more relevant data for analysis. This difference represents a novel contribution because it captures public opinion more comprehensively than previous research.

Furthermore, in the preprocessing stage, this study applies standard steps, such as case folding, cleansing, stopword removal, and stemming, which are also used in previous research. However, this study demonstrates an improvement in the dataset division method, where the data is divided into three groups: training, testing, and validation. In contrast, previous research only divided the dataset into two groups without additional validation. Dividing the dataset into three groups provides better testing quality because it allows the model to be validated before final testing, thereby increasing the prediction results' stability.

Another important difference is the classification algorithm method. Previous research has often used the KNN algorithm, which classifies data based on the proximity between data points. Although this method is simple and easy to implement, KNN has drawbacks in handling large datasets and is susceptible to noise. This study utilized the decision tree algorithm, which is capable of producing a classification model that is easier to interpret through a decision tree structure and is able to understand hierarchical data patterns based on the most influential attributes. The use of decision trees has proven to provide a more transparent and effective analytical approach, especially in the context of long text comments.

Furthermore, this study conducted a manual analysis of the decision tree model's prediction results to evaluate the

suitability of the sentiment context to the original data, rather than relying solely on the statistical accuracy value from the confusion matrix. This manual approach provides a stronger level of verification because it can identify prediction errors based on the actual meaning of the sentence, thereby improving the accuracy of the interpretation of the results and strengthening the validity of the study.

IV. CONCLUSION

This study demonstrates that the decision tree algorithm is capable of classifying Palembang residents' sentiments regarding traffic congestion with an accuracy of 93.77% (training), 90.40% (validation), and 90.42% (testing). Despite the high performance metrics, manual verification results indicate that the prediction accuracy rate is still below 60%, especially for informal and local language comments. This indicates that the model is not yet capable of understanding semantic context in depth. Future research should use deep learning approaches, such as LSTM or BERT, to improve context understanding and the accuracy of social media sentiment analysis, which changes over time.

REFERENCES

- [1] D. Antoni, M. I. Herdiansyah, M. Akbar, and A. Sumitro, "Pengembangan Infrastruktur Jaringan Untuk Meningkatkan Pelayanan Publik di Kota Palembang," *J. Media Inform. Budidarma*, vol. 5, no. 4, p. 1652, 2021, doi: 10.30865/mib.v5i4.3318.
- [2] Peraturan Pemerintah Republik Indonesia Nomor 30, "Peraturan Pemerintah Republik Indonesia Nomor: 30 Tahun 2021, Tentang Penyelenggaraan Bidang Lalu Lintas Dan Angkutan Jalan," *LN.2021/No.40, TLN No.6642, jdih.setkab.go.id43 hlm.*, no. 085113, pp. 1-57, 2021, [Online]. Available: <https://peraturan.bpk.go.id/Home/Details/161874/pp-no-30-tahun-2021>.
- [3] J. J. Van Bavel, C. E. Robertson, K. del Rosario, J. Rasmussen, and S. Rathje, "Social Media and Morality," *Annu. Rev. Psychol.*, vol. 75, pp. 311-340, 2024, doi: 10.1146/annurev-psych-022123-110258.
- [4] F. Bruno, A. Mautone, D. A. Ali, A. Fassima, H. Khabbache, and A. Rizzo, "Evaluating Facebook scales: A systematic review of the psychological assessment tools," *Syst. Rev. Artic. Psychol.*, vol. 2, no. 3, pp. 142-156, 2025, doi: 10.5281/zenodo.13351015.
- [5] A. Nazir, Y. Rao, L. Wu, and L. Sun, "Issues and Challenges of Aspect-based Sentiment Analysis: A Comprehensive Survey," *IEEE Trans. Affect. Comput.*, vol. 13, no. 2, pp. 845-863, 2022, doi: 10.1109/TAFFC.2020.2970399.
- [6] H. M. Sayeed, W. Smallwood, S. G. Baird, and T. D. Sparks, "NLP meets materials science: Quantifying the presentation of materials data in literature," *Matter*, vol. 7, no. 3, pp. 723-727, 2024, doi: 10.1016/j.matt.2023.12.032.
- [7] Y. Mao, Q. Liu, and Y. Zhang, "Sentiment-analysis-methods, applications, and challenges: A systematic-literature-review," *J. King Saud Univ. Comput. Inf. Sci.*, vol.36, no.4, p.102048, 2024, doi: 10.1016/j.jksuci.2024.102048.
- [8] A. Chamekh, M. Mahfoudh, and G. Forestier, "Sentiment Analysis Based on Deep Learning in E-Commerce," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 13369 LNAI, pp. 498-507, 2022, doi: 10.1007/978-3-031-10986-7_40.
- [9] E. Helmud, E. Helmud, F. Fitriyani, and P. Romadiana, "Classification Comparison Performance of Supervised Machine Learning Random Forest and Decision Tree Algorithms Using Confusion Matrix," *J.*

- Sisfokom (Sistem Inf. dan Komputer)*, vol. 13, no. 1, pp. 92–97, 2024, doi: 10.32736/sisfokom.v13i1.1985.
- [10] C.A. Nurhaliza Agustina, R. Novita, Mustakim, and N.E. Rozanda, “The Implementation of TF-IDF and Word2Vec on Booster Vaccine Sentiment Analysis Using Support Vector Machine Algorithm,” *Procedia Comput. Sci.*, vol.234, pp.156–163, 2024, doi: 10.1016/j.procs.2024.02.162.
- [11] J. Jin, “Research and construction of student management platform for special needs students with decision tree model and big data technology,” *Syst. Soft Comput.*, vol. 7, no. May, p. 200310, 2025, doi: 10.1016/j.sasc.2025.200310.
- [12] R. R. Fajriani *et al.*, “Analisis sentimen pengguna lrt jabodebek di platform x menggunakan algoritma svm,” vol. 8, no. 6, pp. 12446–12453, 2024.
- [13] M. A. Brown, A. Gruen, G. Maldoff, S. Messing, Z. Sanderson, and M. Zimmer, “Web scraping for research: legal, ethical, institutional, and Scientific Considerations,” pp.1–43, 2024, [Online]. Available: <http://arxiv.org/abs/2410.23432>, 2024.
- [14] I. Lauriola, A. Lavelli, and F. Aioli, “An introduction to Deep Learning in Natural Language Processing: Models, techniques, and tools,” *Neurocomputing*, vol. 470, no. xxxx, pp. 443–456, 2022, doi: 10.1016/j.neucom.2021.05.103.
- [15] A. Kurniasih and L. P. Manik, “On the Role of Text Preprocessing in BERT embedding-based DNNs for Classifying Informal Texts,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 13, no. 6, pp. 927–934, 2022, doi: 10.14569/IJACSA.2022.01306109.
- [16] N.Wulandari, Y.Cahyana, and H.Handayani, “Sentiment Analysis on the Relocation of the National Capital (IKN) on Social Media X-Using Naive Bayes and K-Nearest Neighbor (KNN) Methods,” vol.9, no.3, pp.724–731, 2025.
- [17] M. Heydarian, T. E. Doyle, and R. Samavi, “MLCM: Multi-Label Confusion Matrix,” *IEEE Access*, vol. 10, pp. 19083–19095, 2022, doi: 10.1109/ACCESS.2022.3151048.
- [18] N. Trung Tuyen, N. Thanh Dang, D. Hon Minh, and T. Q. Nguyen, “Enhancing human activity recognition model performance through KMeans-based stratified data splitting,” *Meas. Control (United Kingdom)*, 2025, doi: 10.1177/00202940241312873.
- [19] M. Grootendorst, “BERTopic: Neural topic modeling with a class-based TF-IDF procedure,” 2022, [Online]. Available: <http://arxiv.org/abs/2203.05794>, 2022.
- [20] A. Ayashi, A. Pourahmad, H. A. Faraji Sabokbar, S. Zanganeh Shahraki, and F. Carbone, “Introducing an intelligent web scraping (mining) method for analyzing online reviews of Tehran accommodation users,” *J. Urban Tour.*, vol.12, no.1, pp.99–118, 2025, [Online]. Available: <http://doi.org/10.22059/JUT.2025.378415.1221>