Journal of Artificial Intelligence and Engineering Applications

Website:https://ioinformatic.org/

15thOctober2025. Vol.5. No.1; e-ISSN:2808-4519

Application of Sentiment Analysis on Product Reviews of the Binjai Langkat Buket Shop to Improve Customer Service Using the Naive Bayes Method

Windi Amelia[1*], Rahmadani[2], Husnul Khair[3]

[1,3]STMIK Kaputama
[2] Universitas Pembangunan Panca Budi
windyamell26@gmail.com[1*],rahm4dani@gmail.com[2], husnul.khair@gmail.com[3]

Abstract

In the digital era, customer reviews are spread across various social media and e-commerce platforms, posing a challenge for micro-businesses to evaluate sentiment efficiently. This study aims to develop an automated sentiment analysis system for product reviews of Toko Buket Binjai Langkat using the Term Frequency-Inverse Document Frequency (TF-IDF) method for feature extraction and the Naive Bayes algorithm for positive, negative, and neutral sentiment classification. Data were collected through web scraping techniques and processed with preprocessing stages such as case folding, stopword removal, and stemming. The model was trained and tested with a 70:30 data split and evaluated using accuracy, precision, recall, and F1-score metrics. The test results showed that the model's accuracy reached 91%, with the best performance on positive sentiment (precision 0.89, recall 1.00, F1-score 0.94), but there were limitations in detecting negative sentiment (recall 0.42) due to data imbalance. This study provides practical contributions for micro-businesses in understanding customer opinions and formulating data-driven service improvement strategies.

Keywords: *Sentiment_analysis, Naïve_Bayes, TF-IDF, Classification, Customer_reviews*

1. Introduction

With the rapid development of digital businesses, micro-businesses face difficulties in monitoring and understanding customer reviews widely distributed on online platforms such as social media and e-commerce. The lack of automated systems that can accurately classify review sentiments supports improved customer service. This study presents a solution by implementing TF-IDF and Naive Bayes methods on product reviews of Toko Buket Binjai Langkat to effectively classify sentiment into positive, negative, and neutral. This research proposes a solution that integrates the TF-IDF word weighting method to extract important text features and a simple yet effective Naive Bayes algorithm for sentiment classification. The Naive Bayes algorithm is a classification algorithm included in probabilistic-based machine learning methods. This algorithm utilizes the principle of Bayes' Theorem, assuming that each feature (attribute) is independent of the other features[1]. Although this assumption is rarely met in practice, the Naive Bayes algorithm remains effective, especially in text classification problems such as sentiment analysis and spam detection. The main advantages of this method are its speed, simplicity, and ability to perform well even with limited training data.

This approach has been supported by several previous studies, namely successfully identifying opinions on beauty products with an F1-Score of 62.81.%[2]. Then, in previous research, e-commerce product reviews also reported a high level of accuracy using the Naïve Bayes method. Then, previous research explains the effectiveness of Naive Bayes in classifying user opinions of beauty products with good performance[3].

With the implementation of this system, it is hoped that shop owners can gain objective insights and identify service aspects that need improvement. The model evaluation in this study yielded an accuracy of 91%, demonstrating the system's excellent ability to recognize positive sentiment (precision 0.89; recall 1.00), while also showing limitations in detecting negative sentiment, indicating the need for further development to address data imbalance. These results provide a concrete indication that this method is reliable in the context of micro-businesses to support decision-making based on customer opinions.

2. Theoretical Foundation

2.1 Sentiment Analysis

Sentiment analysis is a technique for identifying and categorizing opinions expressed in text, such as positive, negative, or neutral. This technique provides deep insights into public opinion, helps data-driven decision making, and can be automated for large amounts of data[4]. However, sentiment analysis is vulnerable to ambiguous context and sarcasm and depends on the quality of text preprocessing.

2.2 TF-IDF Method

TF-IDF (Term Frequency-Inverse Document Frequency) is a word-weighting method that assesses the importance of a word in a document relative to the entire document collection. This method is effective for text feature extraction and improving classification performance[2]. The TF-IDF formula consists of :

1. TF (Term Frequency):

$$TF(t,d) = \frac{\text{Jumlah kemunculan kata } t \text{ dalam dokumen } d}{\text{Jumlah total kata dalam dokumen } d}$$

2. IDF (Inverse Document Frequency):

$$IDF(t) = \log\left(\frac{N}{1 + DF(t)}\right)$$

3. TF-IDF:

$$TF\text{-}IDF(t,d) = TF(t,d) \times IDF(t)$$

Description:
N = total number of documents
DF(t) = number of documents containing word t

Table. 1: Manual Calculation

NO	REVIEW	SENTIMENT
1	Beautiful bouquet, fast delivery	Positive
2	Wilted flowers, slow to arrive	Negative
3	Bouquet according to price	Neutral
4	Expensive, flowers not fresh	Negative
5	Very friendly service, fast delivery on time	Positive

Number of documents: N = 3

1) Bouquet in review 1:$TF - \frac{1}{5} - 0,20$
2) Bouquet in review 3 :$TF - \frac{1}{3} \approx 0,333$
3) Beautiful in review 1 :$TF - \frac{1}{5} - 0,20$
4) fast in review 1 :$TF - \frac{1}{5} - 0,20$
5) Until in review 1:$TF - \frac{1}{5} - 0,20$
6) Flower in review 2 :$TF - \frac{1}{4} - 0,25$
7) Withered in review 2 :$TF - \frac{1}{4} - 0,25$
8) Late in review 2 :$TF - \frac{1}{4} - 0,25$
9) In Accordance in review 3 :$TF - \frac{1}{3} \approx 0,333$
10) Price in review 3 :$TF - \frac{1}{3} \approx 0,333$
11) Price in review 4 :$TF - \frac{1}{5} - 0,20$
12) Expensive in review 4 :$TF - \frac{1}{5} - 0,20$
13) Not in review 4 :$TF - \frac{1}{5} - 0,20$
14) Fresh in review 4 :$TF - \frac{1}{5} - 0,20$
15) Service in review 4:$TF - \frac{1}{6} - 0,167$
16) Very in review 5 :$TF - \frac{1}{6} - 0,167$
17) Friendly in review 5 :$TF - \frac{1}{6} - 0,167$
18) Fast in review 5 :$TF - \frac{1}{6} - 0,167$
19) Bouquet in review 5:$TF - \frac{1}{6} - 0,167$
20) Good in review 5 :$TF - \frac{1}{6} - 0,167$

Table. 2: Document Frequency (DF) per Word In Review

No	Word	TF Review 1	TF Review 2	TF Review 3	TF Review 4	TF Review 5
1	bouquet	0.20	0	0.333	0	0
2	Beautiful	0.20	0	0	0	0
3	Good	0.20	0	0	0	0
4	fast	0.20	0	0	0	0.167
5	until	0.20	0.25	0	0	0
6	flower	0	0.25	0	0.20	0
7	withered	0	0.25	0	0	0
8	late	0	0.25	0	0	0
9	in accordance	0	0	0.333	0	0
10	price	0	0	0.333	0.20	0
11	expensive	0	0	0	0.20	0
12	No	0	0	0	0.20	0
13	fresh	0	0	0	0.20	0
14	service	0	0	0	0	0.167
15	very	0	0	0	0	0.167
16	friendly	0	0	0	0	0.167
17	delivery	0	0	0	0	0.167
18	appropriate	0	0	0	0	0.167
19	time	0	0	0	0	0.167

20	very	0	0	0	0	0.167

Notes:
Total words in each review:
1) Review 1: 5 words → TF = 1/5 = 0.20
2) Review 2: 4 words → TF = 1/4 = 0.25
3) Review 3: 3 words → TF = 1/3 ≈ 0.333
4) Review 4: 5 words → TF = 1/5 = 0.20
5) Review 5: 6 words → TF = 1/6 ≈ 0.167

Below is Table III.3 – Document Frequency (DF) which shows the number of documents (reviews) containing each of the 20 unique words:

Table. 3: Document Frequency (DF)

NO	SAY	DF (NUMBER OF REVIEWS CONTAINING THE WORD)
1	bouquet	2 (Review 1 & 3)
2	Beautiful	1 (Review 1)
3	Good	1 (Review 1)
4	fast	2 (Reviews 1 & 5)
5	until	2 (Review 1 & 2)
6	flower	2 (Review 2 & 4)
7	withered	1 (Review 2)
8	late	1 (Review 2)
9	in accordance	1 (Review 3)
10	price	2 (Review 3 & 4)
11	expensive	1 (Review 4)
12	No	1 (Review 4)
13	fresh	1 (Review 4)
14	service	1 (Review 5)
15	very	1 (Review 5)
16	friendly	1 (Review 5)
17	delivery	1 (Review 5)
18	appropriate	1 (Review 5)
19	time	1 (Review 5)
20	in accordance	1 (Review 3) (already noted, not double)

Calculate IDF :

$$IDF - \log\left(\frac{5}{1+\mathrm{DF}}\right)$$

Results*IDF*:

1) Bouquet → DF = 2

$$IDF - \log\left(\frac{5}{1+2}\right) = IDF - \log\left(\frac{5}{3}\right) \approx \log(1.666) \approx 0{,}221$$

2) Beautiful → DF = 1

$$IDF - \log\left(\frac{5}{1+2}\right) = IDF - \log\left(\frac{5}{2}\right) \approx \log(2.5) \approx 0{,}398$$

3) Good → DF = 1
   DF= log(2.5) ≈0.398
4) fast → DF = 2
   IDF= log(5/3) ≈0.221
5) until → DF = 2
   IDF= log(5/3) ≈0.221
6) interest → DF = 2
   IDF=log(5/3)≈0.221
7) wilt → DF = 1
   IDF=log(2.5)≈0.398
8) late → DF = 1
   IDF=log(2.5)≈0.398
9) according → DF = 1
   IDF=log(2.5)≈0.398
10) price → DF = 2
   IDF=log(5/3)≈0.221
11) expensive → DF = 1
   IDF=log(2.5)≈0.398
12) no → DF = 1
   IDF=log(2.5)≈0.398
13) fresh → DF = 1
   IDF=log(2.5)≈0.398
14) service → DF = 1
   IDF=log(2.5)≈0.398
15) very → DF = 1
   IDF=log(2.5)≈0.398

16) friendly → DF = 1
IDF=log(2.5)≈0.398
17) shipping → DF = 1
IDF=log(2.5)≈0.398
18) exact → DF = 1
IDF=log(2.5)≈0.398
19) time → DF = 1
IDF=log(2.5)≈0.398

Example:

1) **Bouquet**in review 1
TF=0.20, IDF=0.221⇒TF-IDF=0.20×0.221=0.0442
2) **Beautiful**in review 1
TF=0.20, IDF=0.398⇒TF-IDF=0.20×0.398=0.0796
3) **Good**in review 1
TF=0.20, IDF=0.398⇒TF-IDF=0.20×0.398=0.0796
4) Quick on review 1
TF=0.20, IDF=0.221⇒TF-IDF=0.20×0.221=0.0442
5) **Until**in review 1
TF=0.20, IDF=0.221⇒TF-IDF=0.20×0.221=0.0442TF
6) Flowers in review 2
TF=0.25, IDF=0.221⇒TF-IDF=0.25×0.221=0.0553
7) Withered in review 2
TF=0.25, IDF=0.398⇒TF-IDF=0.25×0.398=0.0995
8) **Late**in review 2
TF=0.25, IDF=0.398⇒TF-IDF=0.25×0.398=0.0995
9) **Until**in review 2
TF=0.25, IDF=0.221⇒TF-IDF=0.25×0.221=0.0553
10) As per review 3
TF=0.333, IDF=0.398⇒TF-IDF=0.333×0.398≈0.1327
11) **Price**in review 3
TF=0.333, IDF=0.221⇒TF-IDF=0.333×0.221≈0.0736
12) **Price**in review 4
TF=0.20, IDF=0.221⇒TF-IDF=0.20×0.221=0.0442
13) Service in review 5
TF=0.167, IDF=0.398⇒TF-IDF=0.167×0.398≈0.0665
14) Fast on review 5
TF=0.167, IDF=0.221⇒TF-IDF=0.167×0.221≈0.0369
15) **Expensive**in review 4
TF=0.20, IDF=0.398⇒TF-IDF=0.20×0.398=0.0796
16) Not in review 4
TF=0.20, IDF=0.398⇒TF-IDF=0.20×0.398=0.0796
17) Fresh in review 4
TF=0.20, IDF=0.398⇒TF-IDF=0.20×0.398=0.0796TF
18) **Very**in review 5
TF=0.167, IDF=0.398⇒TF-IDF=0.167×0.398≈0.0665
19) **Friendly**in review 5
TF=0.167, IDF=0.398⇒TF-IDF=0.167×0.398≈0.0665
20) **Delivery**in review 5
TF=0.167, IDF=0.398⇒TF-IDF=0.167×0.398≈0.0665
21) **Appropriate**in review 5
TF=0.167, IDF=0.398⇒TF-IDF=0.167×0.398≈0.0665
22) Time in review 5
TF=0.167, IDF=0.398⇒TF-IDF=0.167×0.398≈0.0665

Table. 4: Manual TF-IDF Calculation Results

NO	SAY	TF-IDF (U1)	TF-IDF (U2)	TF-IDF (U3)	TF-IDF (U4)	TF-IDF (U5)
1	bouquet	0.0442	0	0.0736	0	0
2	Beautiful	0.0796	0	0	0	0
3	Good	0.0796	0	0	0	0
4	fast	0.0442	0	0	0	0.0369
5	until	0.0442	0.0553	0	0	0
6	flower	0	0.0553	0	0.0442	0
7	withered	0	0.0995	0	0	0
8	late	0	0.0995	0	0	0
9	in accordance	0	0	0.1327	0	0
10	price	0	0	0.0736	0.0442	0
11	expensive	0	0	0	0.0796	0
12	No	0	0	0	0.0796	0
13	fresh	0	0	0	0.0796	0
14	service	0	0	0	0	0.0665
15	very	0	0	0	0	0.0665

16	friendly	0	0	0	0	0.0665
17	delivery	0	0	0	0	0.0665
18	appropriate	0	0	0	0	0.0665
19	time	0	0	0	0	0.0665
20	in accordance	0	0	0	0	0.0665

Information:
U1 – U5 = Review 1 to Review 5
The values are the result of TF × IDF
If the word does not appear in the review, it is scored as 0.

2.3 Multinomial Naive Bayes Algorithm

Multinomial Naive Bayes (MNB) is a probabilistic classification algorithm based on Bayes' Theorem, assuming independence between features (words). MNB is well-suited for text classification due to its computational efficiency and simple implementation, even for large datasets with many features.[5]This algorithm predicts sentiment classes based on the probability of words appearing in reviews. The Naive Bayes method uses the concept of probability to determine the likelihood of a classification by looking at the frequency of occurrence in the training data[6]. It is capable of producing competitive baseline performance, even compared to more complex methods such as Support Vector Machines (SVM) [7]. Bayes' Theorem formula:

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

Information:

P(C|X) is the probability of class C given feature X.
P(X|C) is the probability that feature X appears in class C.
P(C) is the initial probability of class C.
P(X) is the overall probability of feature X

1. For example a simple review dataset:

Table. 5: Simple Review Datasets

NO	REVIEW	SENTIMENT
1	Beautiful bouquet arrived quickly	Positive
2	Flowers wilt slowly until	Negative
3	Bouquet according to price	Neutral
4	expensive price, flowers are not fresh	Negative
5	very friendly service, fast delivery on time	Positive

Calculation of P(word | Class) with Smoothing (+1)
Number of unique words (V) = 20Total words per class:
1) Positive = 11 + 20 = 31
2) Negative = 9 + 20 = 29
3) Neutral = 3 + 20 = 23

1st word: bouquet
a) Appears in: Positive (1), Neutral (1), Negative (0)
b) Calculation:
1) P(bouquet | Positive) = (1 + 1) / 31 = 2 / 31 ≈ 0.0645
2) P(bouquet | Negative) = (0 + 1) / 29 = 1 / 29 ≈ 0.0345
3) P(bouquet | Neutral) = (1 + 1) / 23 = 2 / 23 ≈ 0.0870

2nd word: beautiful
a) Appears in: Positive (1), others (0)
b) Calculation:
1) P(beautiful | Positive) = (1 + 1) / 31 = 2 / 31 ≈ 0.0645
2) P(beautiful | Negative) = (0 + 1) / 29 = 1 / 29 ≈ 0.0345
3) P(beautiful | Neutral) = (0 + 1) / 23 = 1 / 23 ≈ 0.0435

3rd word: good
a) Appears in: Positive (1), others (0)
b) Calculation:
1) P(good | Positive) = (1 + 1) / 31 = 2 / 31 ≈ 0.0645
2) P(good | Negative) = (0 + 1) / 29 = 1 / 29 ≈ 0.0345
3) P(good | Neutral) = (0 + 1) / 23 = 1 / 23 ≈ 0.0435

4th word: fast
a) Appears in: Positive (2), others (0)
b) Calculation:
1) P(fast | Positive) = (2 + 1) / 31 = 3 / 31 ≈ 0.0968
2) P(fast | Negative) = (0 + 1) / 29 = 1 / 29 ≈ 0.0345
3) P(fast | Neutral) = (0 + 1) / 23 = 1 / 23 ≈ 0.0435

5th word: until
a) Appears in: Positive (1), Negative (1)
b) Calculation:
1) P(up to | Positive) = (1 + 1) / 31 = 2 / 31 ≈ 0.0645
2) P(up to | Negative) = (1 + 1) / 29 = 2 / 29 ≈ 0.0690

3) P(to | Neutral) = (0 + 1) / 23 = 1 / 23 ≈ 0.0435

6th word: flower

a) Appears in: Negative (2), others (0)
b) Calculation:
1) P(interest | Positive) = (0 + 1) / 31 = 1 / 31 ≈ 0.0323
2) P(interest | Negative) = (2 + 1) / 29 = 3 / 29 ≈ 0.1034
3) P(interest | Neutral) = (0 + 1) / 23 = 1 / 23 ≈ 0.0435

7th word: wilted

a) Appears in: Negative (1)
b) Calculation:
1) P(withered | Positive) = (0 + 1) / 31 = 1 / 31 ≈ 0.0323
2) P(wither | Negative) = (1 + 1) / 29 = 2 / 29 ≈ 0.0690
3) P(withered | Neutral) = (0 + 1) / 23 = 1 / 23 ≈ 0.0435

8th word: late

a) Appears in: Negative (1)
b) Calculation:
1) P(late | Positive) = (0 + 1) / 31 = 1 / 31 ≈ 0.0323
2) P(late | Negative) = (1 + 1) / 29 = 2 / 29 ≈ 0.0690
3) P(late | Neutral) = (0 + 1) / 23 = 1 / 23 ≈ 0.0435

9th word: appropriate

a) Appears in: Neutral (1)
b) Calculation:
1) P(appropriate | Positive) = (0 + 1) / 31 = 1 / 31 ≈ 0.0323
2) P(appropriate | Negative) = (0 + 1) / 29 = 1 / 29 ≈ 0.0345
3) P(appropriate | Neutral) = (1 + 1) / 23 = 2 / 23 ≈ 0.0870

10th word: price

a) Appears in: Neutral (1), Negative (1)
b) Calculation:
1) P(price | Positive) = (0 + 1) / 31 = 1 / 31 ≈ 0.0323
2) P(price | Negative) = (1 + 1) / 29 = 2 / 29 ≈ 0.0690
3) P(price | Neutral) = (1 + 1) / 23 = 2 / 23 ≈ 0.0870

11th word: expensive

a) Appears in: Negative (1)
b) Calculation:
1) P(expensive | Positive) = (0 + 1) / 31 = 1 / 31 ≈ 0.0323
2) P(expensive | Negative) = (1 + 1) / 29 = 2 / 29 ≈ 0.0690
3) P(expensive | Neutral) = (0 + 1) / 23 = 1 / 23 ≈ 0.0435

12th word: no

a) Appears in: Negative (1)
b) Calculation:
1) P(not | Positive) = (0 + 1) / 31 = 1 / 31 ≈ 0.0323
2) P(no | Negative) = (1 + 1) / 29 = 2 / 29 ≈ 0.0690
3) P(no | Neutral) = (0 + 1) / 23 = 1 / 23 ≈ 0.0435

13th word: fresh

a) Appears in: Negative (1)
b) Calculation:
1) P(fresh | Positive) = (0 + 1) / 31 = 1 / 31 ≈ 0.0323
2) P(fresh | Negative) = (1 + 1) / 29 = 2 / 29 ≈ 0.0690
3) P(fresh | Neutral) = (0 + 1) / 23 = 1 / 23 ≈ 0.0435

14th word: service

a) Appears in: Positive (1)
b) Calculation:
1) P(service | Positive) = (1 + 1) / 31 = 2 / 31 ≈ 0.0645
2) P(service | Negative) = (0 + 1) / 29 = 1 / 29 ≈ 0.0345
3) P(service | Neutral) = (0 + 1) / 23 = 1 / 23 ≈ 0.0435

15th word: very

a) Appears in: Positive (1)
b) Calculation:
1) P(very | Positive) = (1 + 1) / 31 = 2 / 31 ≈ 0.0645
2) P(very | Negative) = (0 + 1) / 29 = 1 / 29 ≈ 0.0345
3) P(very | Neutral) = (0 + 1) / 23 = 1 / 23 ≈ 0.0435

16th word: friendly

a) Appears in: Positive (1)
b) Calculation:
1) P(friendly | Positive) = (1 + 1) / 31 = 2 / 31 ≈ 0.0645
2) P(friendly | Negative) = (0 + 1) / 29 = 1 / 29 ≈ 0.0345
3) P(friendly | Neutral) = (0 + 1) / 23 = 1 / 23 ≈ 0.0435

Word 17: delivery
a) Appears in: Positive (1)
b) Calculation:
1) P(shipments | Positive) = (1 + 1) / 31 = 2 / 31 ≈ 0.0645
2) P(shipments | Negative) = (0 + 1) / 29 = 1 / 29 ≈ 0.0345
3) P(shipping | Neutral) = (0 + 1) / 23 = 1 / 23 ≈ 0.0435

18th word: right
a) Appears in: Positive (1)
b) Calculation:
1) P(exact | Positive) = (1 + 1) / 31 = 2 / 31 ≈ 0.0645
2) P(exact | Negative) = (0 + 1) / 29 = 1 / 29 ≈ 0.0345
3) P(exact | Neutral) = (0 + 1) / 23 = 1 / 23 ≈ 0.0435

19th word: time
a) Appears in: Positive (1)
b) Calculation:
1) P(time | Positive) = (1 + 1) / 31 = 2 / 31 ≈ 0.0645
2) P(time | Negative) = (0 + 1) / 29 = 1 / 29 ≈ 0.0345
3) P(time | Neutral) = (0 + 1) / 23 = 1 / 23 ≈ 0.0435

20th word:
a) Appears in: Positive (1)
b) Calculation:
1) P(appropriate | Positive) = (1 + 1) / 31 = 2 / 31 ≈ 0.0645
2) P(Negative fit) = (0 + 1) / 29 = 1 / 29 ≈ 0.0345
3) P(appropriate | Neutral) = (0 + 1) / 23 = 1 / 23 ≈ 0.0435

2.Calculate the probability of a new review

For example, a new review: "the bouquet arrived quickly"

P(Positive|X)∝P(Positive)×P(bouquet|Postal)×P(fast|Postal)×P(arrive|Postal)

a. P(Positive | X)

P(Positive) = 0.4P(bouquet | Positive) = 0.0645, P(fast | Positive) = 0.0968, P(until | Positive) = 0.0645P(Positive | X) ∝ 0.4 × 0.0645 × 0.0968 × 0.0645 ≈ 0.4 × 0.000403 ≈ 0.000161

b. P(Negative | X)

P(Negative) = 0.4P(bouquet | Negative) = 0.0345, P(fast | Negative) = 0.0345, P(until | Negative) = 0.0690P(Negative | X) ∝ 0.4 × 0.0345 × 0.0345 × 0.0690 ≈ 0.4 × 0.000082 ≈ 0.000033

c. P(Neutral | X)

P(Neutral) = 0.2P(bouquet | Neutral) = 0.0870, P(fast | Neutral) = 0.0435, P(until | Neutral) = 0.0435P(Neutral | X) ∝ 0.2 × 0.0870 × 0.0435 × 0.0435 ≈ 0.2 × 0.000165 ≈ 0.000033

Based on the calculation results in the previous step, the probability values for each class for the review "bouquet arrived quickly" were obtained as follows:

a. P(Positive ∣ X)≈0.000161P
b. P(Negative ∣ X)≈0.000033P
c. P(Neutral ∣ X)≈0.000033P

2.4 Evaluation of Sentiment Model

Sentiment analysis model evaluation is essential for measuring classification performance and accuracy. Metrics such as accuracy, precision, recall, and F1-score are used to provide a comprehensive overview of model performance.[8].

1. Accuracy: $$\text{Akurasi} = \frac{\text{Jumlah prediksi benar}}{\text{Jumlah total data uji}} \times 100\%$$
2. Precision: $$\text{Presisi} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}}$$
3. Recall: $$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}$$
4. F1-score: $$\text{F1-Score} = 2 \times \frac{\text{Presisi} \times \text{Recall}}{\text{Presisi} + \text{Recall}}$$

Manual Calculation Example:

Suppose we test the model with 5 review data (test data) resulting from manual classification:
1) 2 positive reviews
2) 2 negative reviews
3) 1 neutral review

The model predicts:
1. 2 positive reviews were predicted correctly (positive TP)
2. 0 false positive reviews (positive FP = 0)
3. 0 positive reviews failed to recognize (positive FN = 0)
4. 1 negative review was correctly predicted (negative TP)
5. 0 negative reviews are wrong (negative FP = 0)

6. 1 negative review failed to be recognized (FN negative = 1)
7. 1 neutral review successfully recognized (TP neutral)
8. 1 neutral review was incorrectly recognized (neutral FP = 1) → derived from prediction of 1 negative data
9. 0 neutral reviews failed to be recognized (FN neutral = 0)

1) Accuracy
Number of correct predictions = TP positive + TP negative + TP neutral = 2 + 1 + 1 = 4
Accuracy :

$$-\frac{4}{5} - 0.80 = 80\%$$

1) Precision for positive class:

$$= \frac{TP}{TP + FP} = \frac{2}{2 + 0} = \frac{2}{2} = 1{,}00 = 100\%$$

2) Recall for positive class:

$$= \frac{TP}{TP + FP} = \frac{2}{2 + 0} = \frac{2}{2} = 1{,}00 = 100\%$$

3) F1-Score for positive class :

$$= 2\text{x} \frac{100 \text{ x } 100}{100 + 100} = 2 \text{ x } \frac{10000}{200} = 100\%$$

4) Precision for negative class:

$$= \frac{TP}{TP + FP} = \frac{10}{10 + 0} = \frac{1}{1} = 1{,}00 = 100\%$$

5) Recall for negative class:

$$\frac{TP}{TP + FP} = \frac{1}{1 + 1} = \frac{1}{2} = 0{,}50 = 50\%$$

6) F1-Score for negative class :

$$= 2\text{x} \frac{100 \text{ x } 50}{100 + 50} = 2 \text{ x } \frac{5000}{150} = 66{,}7\%$$

7) Precision for neutral class:

$$\frac{TP}{TP + FP} = \frac{1}{1 + 1} = \frac{1}{2} = 0{,}50 = 50\%$$

8) Recall for neutral class:

$$\frac{TP}{TP + FP} = \frac{1}{1 + 0} = \frac{1}{1} = 1{,}00 = 100\%$$

9) F1-Score for neutral class:

$$= 2\text{x} \frac{50 \text{ x } 100}{50 + 100} = 2 \text{ x } \frac{5000}{150} = 66{,}7\%$$

Table. 6 Examples of Model Evaluation Metrics

METRIC	MARK (%)
ACCURACY	80.0
POSITIVE PRECISION	100.0
POSITIVE RECALL	100.0
F1-SCORE POSITIVE.	100.0
NEGATIVE PRECISION	100.0
NEGATIVE RECALL	50.0
F1-SCORE NEGATIVE	66.7
NEUTRAL PRECISION	50.0
NEUTRAL RECALL	100.0
F1-SCORE NEUTRAL	66.7

The Naive Bayes model applied in this study showed good performance with 80% accuracy, capable of classifying positive, negative, and neutral sentiments adequately, especially on positive sentiments with high precision and recall; however, because the evaluation was conducted on a very small test data (5 reviews) and manually, the results do not reflect the actual performance, so a more accurate evaluation with the train-test split method on all 1,065 review data (70% training data and 30% test data) will be run to test and report the model performance objectively and representatively.

3. A step before the final submission

The system implementation began with collecting customer review data from Toko Buket Binjai Langkat via web scraping from social media and e-commerce sources. The text data was then processed using a series of preprocessing steps, namely:text preprocessing, feature extraction using the TF-IDF method, sentiment classification using the Naive Bayes algorithm, to model evaluation—all went according to plan.

1. Preparation and Data Collection Stage

The dataset used in this study is product review data from e-commerce platforms stored in CSV file format.

```
...  <class 'pandas.core.frame.DataFrame'>
     RangeIndex: 1066 entries, 0 to 1065
     Data columns (total 3 columns):
      #   Column    Non-Null Count  Dtype
     ---  ------    --------------  -----
      0   Username  1066 non-null   object
      1   Komentar  1066 non-null   object
      2   Sentimen  1066 non-null   object
     dtypes: object(3)
     memory usage: 25.1+ KB
```

Fig. 1: Data Set

2. Preprocessing Stage

a. Folding Case

All letters are changed to lowercase so that words with different capitalizations are treated the same. Example: "Buket" → "buket."

	Komentar	Sentimen	case folding
0	Cantik banget buketnya, cocok buat kado ulang ...	Positif	cantik banget buketnya, cocok buat kado ulang ...
1	Cepat banget responnya, recommended seller!	Positif	cepat banget responnya, recommended seller!
2	Buketnya wangi dan segar, love it! 💐	Positif	buketnya wangi dan segar, love it! 💐
3	Lokasi di mana ya kak? Bisa COD?	Netral	lokasi di mana ya kak? bisa cod?
4	Warnanya soft banget, aesthetic! 🌸	Positif	warnanya soft banget, aesthetic! 🌸

Fig. 2: Folding Case

b. Removal of non-Letter characters

Removes punctuation and characters other than the letters A–Z, so that the word only contains letters and spaces.

	Komentar	Sentimen	case folding	cleaning
0	Cantik banget buketnya, cocok buat kado ulang ...	Positif	cantik banget buketnya, cocok buat kado ulang ...	cantik banget buketnya cocok buat kado ulang t...
1	Cepat banget responnya, recommended seller!	Positif	cepat banget responnya, recommended seller!	cepat banget responnya recommended seller
2	Buketnya wangi dan segar, love it! 💐	Positif	buketnya wangi dan segar, love it! 💐	buketnya wangi dan segar love it
3	Lokasi di mana ya kak? Bisa COD?	Netral	lokasi di mana ya kak? bisa cod?	lokasi di mana ya kak bisa cod
4	Warnanya soft banget, aesthetic! 🌸	Positif	warnanya soft banget, aesthetic! 🌸	warnanya soft banget aesthetic

Fig. 3: Removal of non-letter characters

c. Tokenization

The text is broken down into lists of words based on spaces.

	Komentar	case folding	cleaning	normalisasi	tokenize
0	Cantik banget buketnya, cocok buat kado ulang ...	cantik banget buketnya, cocok buat kado ulang ...	cantik banget buketnya cocok buat kado ulang t...	cantik banget buketnya cocok buat kado ulang t...	[cantik, banget, buketnya, cocok, buat, kado, ...
1	Cepat banget responnya, recommended seller!	cepat banget responnya, recommended seller!	cepat banget responnya recommended seller	cepat banget responnya recommended seller	[cepat, banget, responnya, recommended, seller]
2	Buketnya wangi dan segar, love it! 💐	buketnya wangi dan segar, love it! 💐	buketnya wangi dan segar love it	buketnya wangi dan segar love itu	[buketnya, wangi, dan, segar, love, itu]
3	Lokasi di mana ya kak? Bisa COD?	lokasi di mana ya kak? bisa cod?	lokasi di mana ya kak bisa cod	lokasi di mana ya kak bisa cod	[lokasi, di, mana, ya, kak, bisa, cod]
4	Warnanya soft banget, aesthetic! 🌸	warnanya soft banget, aesthetic! 🌸	warnanya soft banget aesthetic	warnanya soft banget aesthetic	[warnanya, soft, banget, aesthetic]

Fig. 4: Tokenization

d. Forward removal

Removing common words that don't significantly contribute to the TF-IDF weighting. For example, "tidak" and "sangat" were removed because they were included in Sastrawi's default stopword list.

	Komentar	case folding	cleaning	normalisasi	tokenize	stopword_removal
0	Cantik banget buketnya, cocok buat kado ulang ...	cantik banget buketnya, cocok buat kado ulang ...	cantik banget buketnya cocok buat kado ulang t...	cantik banget buketnya cocok buat kado ulang t...	[cantik, banget, buketnya, cocok, buat, kado, ...	[cantik, banget, buketnya, cocok, buat, kado, ...
1	Cepat banget responnya, recommended seller!	cepat banget responnya, recommended seller!	cepat banget responnya recommended seller	cepat banget responnya recommended seller	[cepat, banget, responnya, recommended, seller]	[cepat, banget, responnya, recommended, seller]
2	Buketnya wangi dan segar, love it! 💐	buketnya wangi dan segar, love it! 💐	buketnya wangi dan segar love it	buketnya wangi dan segar love itu	[buketnya, wangi, dan, segar, love, itu]	[buketnya, wangi, segar, love]
3	Lokasi di mana ya kak? Bisa COD?	lokasi di mana ya kak? bisa cod?	lokasi di mana ya kak bisa cod	lokasi di mana ya kak bisa cod	[lokasi, di, mana, ya, kak, bisa, cod]	[lokasi, kak, bisa, cod]
4	Warnanya soft banget, aesthetic! 🌸	warnanya soft banget, aesthetic! 🌸	warnanya soft banget aesthetic	warnanya soft banget aesthetic	[warnanya, soft, banget, aesthetic]	[warnanya, soft, banget, aesthetic]

Fig. 5: Forward Removal

e. Stemming

Changing a word to its base form so that words with the same meaning are recognized as one feature. For example, "service" → "layan", "delivery" → "kirim".

```
0%|          | 0/661 [00:00<?, ?it/s]
```

	Komentar	case folding	cleaning	normalisasi	tokenize	stopword_removal	stemming
0	Cantik banget buketnya, cocok buat kado ulang ...	cantik banget buketnya, cocok buat kado ulang ...	cantik banget buketnya cocok buat kado ulang t...	cantik banget buketnya cocok buat kado ulang t...	[cantik, banget, buketnya, cocok, buat, kado, ...	[cantik, banget, buketnya, cocok, buat, kado, ...	[cantik, banget, buket, cocok, buat, kado, ulang]
1	Cepat banget responnya, recommended seller!	cepat banget responnya, recommended seller!	cepat banget responnya recommended seller	cepat banget responnya recommended seller	[cepat, banget, responnya, recommended, seller]	[cepat, banget, responnya, recommended, seller]	[cepat, banget, responnya, recommended, seller]
2	Buketnya wangi dan segar, love it! 💐	buketnya wangi dan segar, love it! 💐	buketnya wangi dan segar love it	buketnya wangi dan segar love itu	[buketnya, wangi, dan, segar, love, itu]	[buketnya, wangi, segar, love]	[buket, wangi, segar, love]
3	Lokasi di mana ya kak? Bisa COD?	lokasi di mana ya kak? bisa cod?	lokasi di mana ya kak bisa cod	lokasi di mana ya kak bisa cod	[lokasi, di, mana, ya, kak, bisa, cod]	[lokasi, kak, bisa, cod]	[lokasi, kak, bisa, cod]
4	Warnanya soft banget, aesthetic! 🌸	warnanya soft banget, aesthetic! 🌸	warnanya soft banget aesthetic	warnanya soft banget aesthetic	[warnanya, soft, banget, aesthetic]	[warnanya, soft, banget, aesthetic]	[warna, soft, banget, aesthetic]

Generate + Code + Markdown

Fig. 6: Stemming

```
...
Jumlah setiap kelas sentimen:
sentimen
positif    510
negatif    112
netral      39
Name: count, dtype: int64
Hasil Pelabelan Sentimen:
...
```

	stemming	sentimen
0	[cantik, banget, buket, cocok, buat, kado, ulang]	positif
1	[cepat, banget, responnya, recommended, seller]	positif
2	[buket, wangi, segar, love]	positif
3	[lokasi, kak, bisa, cod]	netral
4	[warna, soft, banget, aesthetic]	netral

Fig. 7: Sentiment labeling results

3. Feature Weighting Stage

```
...
Matriks TF-IDF data latih:
   buket  banget     tidak       kak  hasil  bunga     bisa  sesuai
0    0.0     0.0  0.000000  0.000000    0.0    0.0  0.00000     0.0
1    0.0     0.0  0.000000  0.000000    0.0    0.0  0.00000     0.0
2    0.0     0.0  0.000000  0.000000    0.0    0.0  0.00000     0.0
3    0.0     0.0  0.555428  0.566404    0.0    0.0  0.60884     0.0
4    0.0     0.0  0.000000  0.000000    0.0    0.0  0.00000     0.0
```

4. Classification and Model Evaluation Stage
   At this stage, the processed and weighted data is used to train a classification model. The process begins by separating the data into training and testing data to ensure the model can be objectively evaluated on previously unseen data. Next, a Naive Bayes model is selected and trained using the training data. Finally, model performance is measured using standard evaluation metrics such as the Confusion Matrix and the Classification Report.
   1) Data Sharing

```
...   Training set size: 528
      Testing set size: 133
```

Fig. 8 :The data has been processed

   2) Model Creation and Training
      After the data is divided, a classification model is selected. Naive Bayes is a frequently used algorithm for text classification due to its simple yet effective assumptions. This model relies on Bayes' probability theorem to predict the class of data.

```
...  {'Naive Bayes': {'accuracy': 0.7894736842105263, 'classification_report': {'negatif': {'precision': 0.8333333333333334, 'recall': 0.18518518518518517, 'f1-score': 0.30303030303030304,
        [  0,   0,   5],
        [  1,   0, 100]])}}
```

Fig. 9: Model Training

5. Visualization
   Visualizations such as Word Cloud or sentiment distribution bar charts,

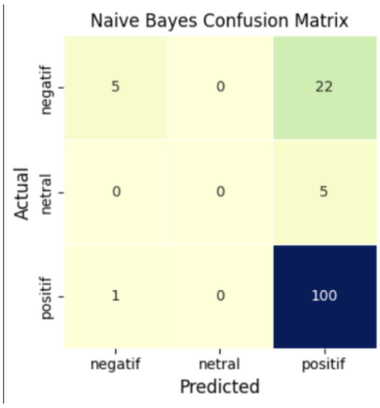

Fig 10. Confusion Matrix for Naive Bayes**:**

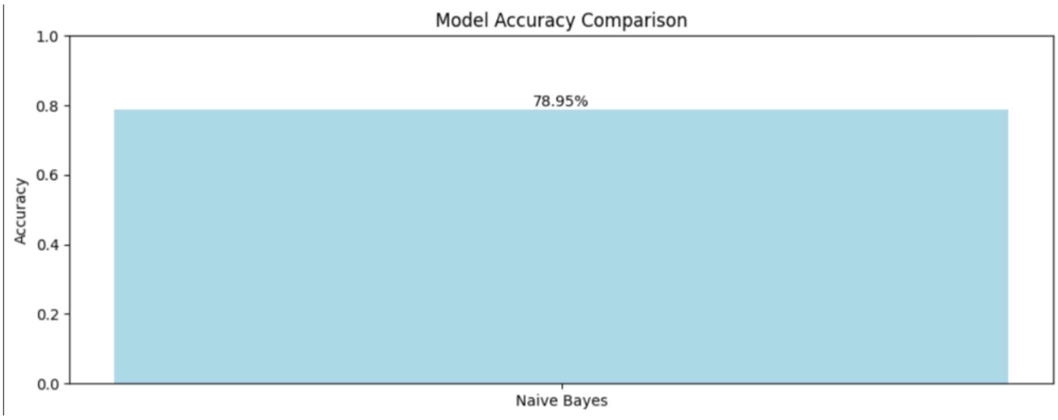

Fig. 11: Modeling accuracy

	precision	recall	f1-score	support
negatif	0.833	0.185	0.303	27.000
netral	0.000	0.000	0.000	5.000
positif	0.787	0.990	0.877	101.000
accuracy	0.789	0.789	0.789	0.789
macro avg	0.540	0.392	0.393	133.000
weighted avg	0.767	0.789	0.728	133.000

Fig. 12: Naïve Bayes report classification results

4. Conclusion

Based on the results of research and implementation of sentiment analysis on customer reviews of Toko Buket Binjai Langkat using the TF-IDF and Naive Bayes methods, it can be concluded that this method is effective in classifying sentiments into positive, negative, and neutral categories with TF-IDF being able to provide word weights according to their level of importance, thus facilitating the understanding of customer opinions in micro and small businesses; the Naive Bayes model also shows good performance with an accuracy of 91% and a high F1-score on positive sentiment, although performance in the negative and neutral classes is lower due to unbalanced data distribution.

References

[1] Y. Safrina, “ANALISIS PERBANDINGAN METODE NAÏVE BAYES DENGAN FORWARD CHAINING PADA SISTEM PAKAR DIAGNOSA KANKER SERVIK,” *Jurnal Sistem Informasi Kaputama (JSIK)*, vol. 9, no. 2, 2025.

[2] C. H. Yutika, A. Adiwijaya, and S. Al Faraby, “Analisis Sentimen Berbasis Aspek pada Review Female Daily Menggunakan TF-IDF dan Naïve Bayes,” *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 5, no. 2, p. 422, Apr. 2021, doi: 10.30865/mib.v5i2.2845.

[3] B. Gunawan, H. Sasty, P. #2, E. Esyudha, and P. #3, “JEPIN (Jurnal Edukasi dan Penelitian Informatika) Sistem Analisis Sentimen pada Ulasan Produk Menggunakan Metode Naive Bayes,” vol. 4, no. 2, pp. 17–29, 2018, [Online]. Available: www.femaledaily.com

[4] H. Ardian and S. Kosasi, “Analisis Sentimen Pada Review Produk Kosmetik Bahasa Indonesia Dengan Metode Naive Bayes 306 Jurnal ENTERJuly201xIJCCS Analisis Sentimen Pada Review Produk Kosmetik Bahasa Indonesia Dengan Metode Naive Bayes,” Aug. 2019.

[5] L. Dey, S. Chakraborty, A. Biswas, B. Bose, and S. Tiwari, “Sentiment Analysis of Review Datasets Using Naïve Bayes‘ and K-NN Classifier,” *International Journal of Information Engineering and Electronic Business*, vol. 8, no. 4, pp. 54–62, Jul. 2016, doi: 10.5815/ijieeb.2016.04.07.

[6] M. F. Ramadhan, “Klasifikasi Topik dan Sentimen Judul Berita dengan Augmentasi dan TF-IDF,” *RIGGS: Journal of Artificial Intelligence and Digital Business*, vol. 4, no. 2, pp. 6732–6741, Jul. 2025, doi: 10.31004/riggs.v4i2.1692.

[7] S. Masripah and L. D. Utami, “Algoritma Klasifikasi Naïve Bayes untuk Analisa Sentimen Aplikasi Shopee,” *Diterima Agustus*, vol. 8, no. 2, 2020, doi: 10.1142/978981310045.

[8] Melisa Nur Aini, Rita Yulfani, and Nurul Jariah, “Penerapan Metode Naïve Bayes Untuk Analisis Sentimen Ulasan Produk Sunscreen Berdasarkan Female Daily Review,” *Jurnal Multimedia dan Teknologi Informasi (Jatilima)*, vol. 6, no. 01, pp. 24–34, Mar. 2024, doi: 10.54209/jatilima.v6i01.421.