

SENTIMENT ANALYSIS OF VIDEO EDITING APPLICATIONS USING SUPPORT VECTOR MACHINE ON GOOGLE COLAB

Mugi Raharjo¹⁾, Jordy Lasmana Putra²⁾, Sujiliani Heristian³⁾, Musriatun Napih⁴⁾

¹Informatika, Universitas Nusa Mandiri

Jl. Raya Jatiwaringin No.2, Cipinang Melayu, Kec. Makasar, Daerah Khusus Jakarta

³Teknologi Komputer, Universitas Bina Sarana Informatia

Jl. Kramat Raya No.98, Kwitang, Kec. Senen, Daerah Khusus Jakarta

^{2,4}Informatika, Universitas Bina Sarana Informatia

Jl. Kramat Raya No.98, Kwitang, Kec. Senen, Daerah Khusus Jakarta

Co Responden Email: jordy.jlp@bsi.ac.id

Abstract [Times New Roman 10 Cetak Tebal dan Miring]

Article history

Received 05 Feb 2025

Revised 19 Mar 2025

Accepted 21 Apr 2025

Available online 30 May 2025

Keywords

Sentiment Analysis,
Support Vector Machine,
CapCut,
Google Colab,
Classification.

Sentiment analysis is an important approach in understanding user opinions about an application. This study aims to analyze user reviews of the CapCut application using the Support Vector Machine (SVM) algorithm on the Google Colab platform. The preprocessing stages include data cleaning, word normalization using a dictionary from Kaggle, case folding, tokenization, stopword removal, and stemming. Furthermore, the data is converted into a numerical representation using the TF-IDF vectorization method. The labeling process is carried out using a sentiment lexicon obtained from GitHub. After performing data splitting, the SVM model is applied to classify sentiment into three categories: positive, negative, and neutral. The evaluation results show that the SVM model achieves the best accuracy of 90.12%. Based on the classification report, the model has high precision of 0.94 for positive and negative classes and 0.83 for the neutral class. Additionally, the confusion matrix indicates that the model can classify sentiment quite well, although there are still minor errors in predicting neutral sentiment. The findings of this study demonstrate that the SVM method can be effectively used to analyze user sentiment toward the CapCut application, providing valuable insights for improving user experience in the future.

Abstrak

Riwayat

Diterima 05 Feb 2025

Revisi 19 Mar 2025

Disetujui 21 Apr 2025

Terbit online 30 Mei 2025

Kata Kunci

Analisis Sentimen,
Support Vector Machine,
CapCut,
Google Colab,
Klasifikasi

Analisis sentimen merupakan pendekatan penting dalam memahami opini pengguna terhadap suatu aplikasi. Penelitian ini bertujuan untuk menganalisis ulasan pengguna terhadap aplikasi CapCut menggunakan algoritma Support Vector Machine (SVM) pada platform Google Colab. Tahapan preprocessing yang dilakukan meliputi cleaning data, normalisasi kata menggunakan kamus dari Kaggle, case folding, tokenisasi, penghapusan stopwords, dan stemming. Selanjutnya, data diubah menjadi representasi numerik menggunakan metode TF-IDF vectorization. Proses pelabelan dilakukan dengan memanfaatkan leksikon sentimen yang diperoleh dari GitHub. Setelah dilakukan data splitting, model SVM diterapkan untuk klasifikasi sentimen ke dalam tiga kategori: positif, negatif, dan netral. Hasil evaluasi menunjukkan bahwa model SVM memberikan akurasi terbaik sebesar 90,12%. Berdasarkan classification report, model memiliki presisi tinggi sebesar 0,94 untuk kelas positif dan negatif, serta 0,83 untuk kelas netral. Selain itu, confusion matrix menunjukkan bahwa model mampu mengklasifikasikan sentimen dengan cukup baik meskipun masih terdapat sedikit kesalahan dalam prediksi sentimen netral. Hasil penelitian ini menunjukkan bahwa metode SVM dapat digunakan secara efektif untuk menganalisis sentimen pengguna terhadap aplikasi CapCut, sehingga dapat menjadi acuan dalam meningkatkan pengalaman pengguna di masa mendatang.

PENDAHULUAN

Perkembangan teknologi telah menciptakan berbagai platform media sosial. Media sosial berperan penting dalam mengubah perilaku interaksi sosial para pengguna internet. Dalam era digital yang semakin berkembang (Machmud, Wibisono, & Suryani, 2025), Tanpa disadari Aplikasi sekarang sangat penting bagi kehidupan manusia. Sektor perbankan merupakan salah satu contoh Aplikasi perkembangan yang sejalan dengan tren kemajuan teknologi ini (Dwi et al., 2025). Semakin banyaknya pengguna internet, semakin banyak orang yang dapat mengakses dan memanfaatkan berbagai aplikasi untuk kebutuhan sehari-hari, baik itu untuk komunikasi, hiburan, produktivitas, maupun kreativitas (Khaira, Aryani, & Hardian, 2023). Aplikasi pengeditan video telah menjadi alat penting bagi pengguna media sosial dan kreator konten. Salah satu aplikasi yang populer adalah CapCut, yang menawarkan berbagai fitur untuk menghasilkan video berkualitas tinggi. Namun, ulasan pengguna terhadap aplikasi ini beragam, mulai dari apresiasi terhadap fitur-fitur inovatif hingga kritik terhadap keterbatasan tertentu. Untuk memahami sentimen pengguna secara mendalam, analisis sentimen menjadi pendekatan yang relevan dalam mengkaji opini yang disampaikan melalui ulasan pengguna.

Penelitian ini bertujuan untuk menganalisis ulasan pengguna terhadap aplikasi CapCut dengan menerapkan algoritma *Support Vector Machine (SVM)* pada platform *Google Colab*. Metode pengambilan data *scraping* cocok digunakan untuk mengumpulkan data informasi yang berjumlah besar. Teknik *scraping* adalah metode pengambilan dokumen, menganalisis kemudian memproses data dari halaman internet yang digunakan (Yuniar & Kismiantini, 2023). Tahapan *preprocessing* yang dilakukan meliputi pembersihan data, normalisasi kata menggunakan kamus dari *Kaggle*, *case folding*, *tokenation*, penghapusan *stopwords*, dan *stemming*. *Preprocessing* digunakan dalam dokumen mentah hingga siap untuk dilakukan analisis teks yang telah disiapkan (Puji Astuti, Alam, & Jaelani, 2022). Data kemudian dikonversi menjadi representasi numerik menggunakan metode

TF-IDF sebelum proses pelabelan dengan leksikon sentimen dari *GitHub*. Setelah pembagian data menjadi set pelatihan dan pengujian, model *SVM* digunakan untuk mengklasifikasikan sentimen ke dalam tiga kategori: positif, negatif, dan netral.

Berdasarkan penelitian terkait di atas, penelitian ini bertujuan untuk menganalisis sentimen ulasan pengguna terhadap aplikasi CapCut dengan menggunakan metode *Support Vector Machine (SVM)*. Selain itu, penelitian ini akan memanfaatkan platform seperti *Google Colab*, *Kaggle*, dan *GitHub* untuk pengolahan dan analisis data. Diharapkan hasil dari penelitian ini dapat memberikan kontribusi dalam pengembangan aplikasi CapCut serta menjadi referensi bagi penelitian selanjutnya dalam bidang analisis sentimen aplikasi pengeditan video.

Sebagai bidang yang sedang berkembang, analisis sentimen teks memiliki potensi besar dalam penelitian dan aplikasi praktis, membantu menjelaskan bahwa penelitian analisis sentimen teks telah menarik semakin banyak perhatian di dalam dan luar negeri (Han, Chien, Chiu, & Cheng, 2020). Analisis sentimen telah diterima secara luas dalam beberapa tahun terakhir, tidak hanya di kalangan peneliti tetapi juga di kalangan bisnis, pemerintah, dan organisasi (Wankhade, Rao, & Kulkarni, 2022). Analisis sentimen adalah suatu proses memahami, mengekstrak dan mengolah data tekstual secara otomatis untuk memperoleh informasi sentimen yang terdapat pada kalimat opini (Indrayanto, Ratnawati, & Rahayudi, 2023). Analisis sentimen mencakup bidang penambahan teks, pemrograman bahasa alami, dan kecerdasan buatan dan dilakukan untuk secara otomatis mengekstraksi, memahami, dan memproses data teks untuk memperoleh informasi berguna dan pengetahuan baru (Aryanti, Misriati, & Sagiyanto, 2023). Analisis sentimen juga dikenal sebagai penambahan opini, cara menguasai, mengekstrak serta mengerjakan informasi bacaan dengan cara otomatis buat memperoleh data afeksi yang tercantum dalam perkataan terbebas dari apakah opini positif atau negatif (Friska Aditia Indriyani, Ahmad Fauzi, & Sutan Faisal, 2023). Salah satu metode pelabelan yang dapat digunakan adalah metode berbasis leksikon. Berbasis leksikon merupakan metode untuk dapat menentukan sentimen atau polaritas opini melalui beberapa

fungsi kata opini dalam dokumen atau kalimat yang didasarkan pada kamus leksikon (Muhammadi, Laksana, & Arifa, 2022) *Support Vector Machine (SVM)* merupakan salah satu metode klasifikasi dengan menggunakan metode supervised learning yang memprediksi kelas berdasarkan pola dari hasil proses training (Romadoni, Umaidah, & Sari, 2020). Temuan penelitian ini berkontribusi pada pemahaman yang lebih mendalam tentang analisis sentimen dan memberikan wawasan tentang efektivitas pendekatan pembelajaran mesin dalam domain ini (Ahmed Khan, Sadiq, Shahid, Alam, & Mohd Su'ud, 2024).

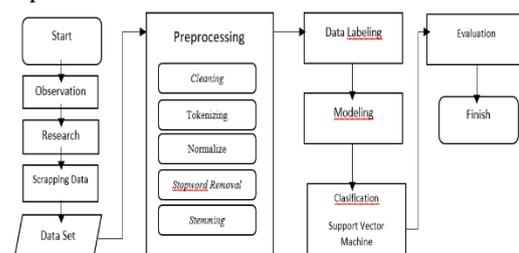
Penelitian sebelumnya yang dilakukan Purnamawati et al mereka melakukan penelitian terhadap aplikasi tiktok dengan algoritma *KNN* pengujian menggunakan 5 kali pengujian rasio fitur kemudian di dapatkan hasil terbaik dari rasio fitur 50% dan $k = 20$ sehingga memperoleh hasil terbaik yaitu nilai *precision* 70,03%, *recall* 67,22%, *accuracy* 83,33% dan *f-measure* 66,26%. Dapat disimpulkan untuk penambahan fitur seleksi dapat membantu meningkatkan hasil *recall*, *f-measure*, *precision*, dan *accuracy* (Purnamawati, Winarto, & Mailasari, 2023). Penelitian yang dilakukan oleh Roland Vincent et al melakukan perbandingan algoritma dengan menggunakan Information Gain, akurasi *Naive Bayes* meningkat dari 59.86% menjadi 72.02% dengan menggunakan Count Vectorizer dan N-gram, sedangkan akurasi *SVM* meningkat dari 61.47% (*TF-IDF*) menjadi 62.38% (*Count Vectorizer dengan N-gram*). Eksperimen juga menunjukkan *Naive Bayes* lebih cocok dengan *Count Vectorizer* dan *SVM* lebih cocok dengan *TF-IDF*. Selain itu, eksperimen juga menunjukkan rasio 80:20 merupakan rasio data latih dan uji terbaik (Vincent, Maulana, & Komarudin, 2024).

METODE PENELITIAN

Tahapan penelitian tersebut menggambarkan tahapan penelitian dalam pemrosesan dan analisis data menggunakan metode *Support Vector Machine (SVM)* untuk klasifikasi. Penelitian dimulai dengan tahap observasi, di mana dilakukan pengamatan awal terhadap permasalahan yang akan diteliti. Setelah itu, masuk ke tahap penelitian (research) guna mengumpulkan informasi lebih lanjut terkait metode dan teknik yang

sesuai untuk menyelesaikan masalah. Data yang dibutuhkan dikumpulkan melalui proses *scraping*, yaitu teknik ekstraksi data dari sumber tertentu, seperti website, media sosial, atau dokumen digital lainnya, hingga membentuk suatu dataset yang akan digunakan dalam analisis. Setelah dataset diperoleh, dilakukan tahap preprocessing untuk membersihkan dan menyiapkan data sebelum dianalisis. Proses ini terdiri dari beberapa langkah, yaitu *cleaning* (pembersihan data dari karakter atau simbol yang tidak diperlukan), *tokenizing* (memecah teks menjadi kata-kata atau unit-unit lebih kecil), *normalize* (menstandarkan format kata, seperti mengubah huruf kapital menjadi huruf kecil), *stopword removal* (menghapus kata-kata umum yang tidak memiliki makna signifikan, misalnya "dan", "atau", "di"), serta *stemming* (mengubah kata ke bentuk dasarnya, misalnya "berlari" menjadi "lari").

Setelah tahap preprocessing selesai, data diberikan label pada tahap data labeling, yang merupakan proses menentukan kategori atau kelas dari setiap data. Selanjutnya, dilakukan modeling, yaitu pembangunan model klasifikasi menggunakan algoritma *Support Vector Machine (SVM)*. Model ini berfungsi untuk mengenali pola dalam data dan mengelompokkan data ke dalam kelas yang telah ditentukan berdasarkan karakteristik yang dianalisis sebelumnya. Setelah model *SVM* diterapkan, dilakukan tahap evaluasi untuk mengukur performa model dalam melakukan klasifikasi. Evaluasi ini bertujuan untuk menilai sejauh mana akurasi dan efektivitas model dalam mengklasifikasikan data dengan benar. Jika hasil evaluasi memuaskan, maka penelitian dianggap selesai dan siap untuk diimplementasikan atau dipublikasikan.



Gambar 1. Tahapan Penelitian

HASIL DAN PEMBAHASAN

1. PENGUMPULAN DATA

Penelitian ini dilakukan dengan bahasa pemrograman python menggunakan platform

google colab dimana kami melakukan data scrapping terhadap aplikasi di google playstore yaitu capcut video editor.

```

3. from google play scraper import reviews, Sort # In
4. import csv
5. from google.colab import files
6. import pandas as pd
7. import os
8.
9. def scrape_reviews(app_id, num_reviews=5000,
10. file_name='capcut_reviews.csv'):
11.     result, _ = reviews(
12.         app_id,
13.         lang='id', # Bahasa Indonesia
14.         country='id', # Negara Indonesia
15.         count=num_reviews,
16.         sort=Sort.NEWEST # Use Sort.NEWEST

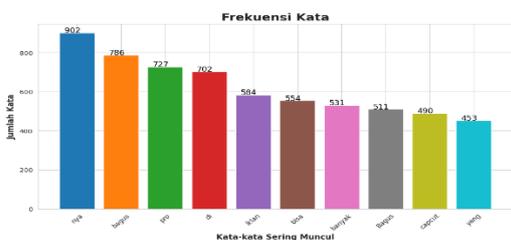
```

Gambar 2. Scrapping Data

Pada tahap pengumpulan data dilakukan scraping data ulasan terakhir sebanyak 5000 data ulasan yang berada pada google playstore. Kode Python pada gambar tersebut digunakan untuk melakukan web scraping terhadap ulasan aplikasi dari Google Play Store menggunakan pustaka google-play-scraper. Pada bagian awal, beberapa pustaka diimpor, termasuk reviews dan Sort dari google_play_scraper, serta pustaka lain seperti csv, files dari google.colab, pandas sebagai pd, dan os. Fungsi scrape_reviews dibuat untuk mengambil ulasan aplikasi berdasarkan app_id, dengan jumlah ulasan maksimum yang dapat ditentukan melalui parameter num_reviews (default 5000). Dalam fungsi ini, ulasan diambil menggunakan reviews() dengan beberapa parameter, termasuk app_id, bahasa (lang) yang diset ke Bahasa Indonesia, dan negara (country) yang juga diset ke Indonesia. Hasil ulasan yang diperoleh nantinya dapat disimpan ke dalam file CSV yang dinamai output_review.csv.

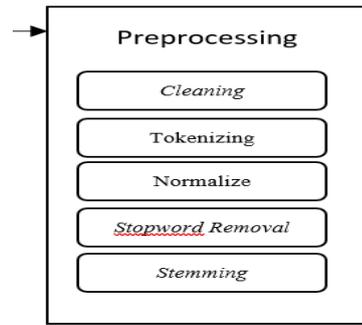


Gambar 3. Wordcloud Kata Sebelum Preprocessing



Gambar 4. Frekuensi Kata Sebelum Preprocessing

2. PREPROCESSING



Gambar 5. Tahapan Preprocessing

Proses preprocessing dalam gambar melibatkan beberapa langkah utama untuk membersihkan dan menyiapkan data teks sebelum dianalisis. Tahap pertama adalah cleaning, yaitu menghapus karakter atau simbol yang tidak diperlukan, seperti tanda baca atau angka yang tidak relevan. Selanjutnya, tokenizing dilakukan untuk memecah teks menjadi unit-unit lebih kecil, seperti kata atau frasa. Kemudian, data dinormalisasi melalui normalize, yang bertujuan untuk menyamakan format teks, misalnya mengonversi huruf besar menjadi huruf kecil atau mengganti ejaan yang berbeda menjadi bentuk standar. Setelah itu, dilakukan stopwords removal, yaitu menghapus kata-kata umum yang tidak memiliki makna signifikan dalam analisis, seperti "dan", "atau", dan "di". Terakhir, stemming diterapkan untuk mengubah kata ke bentuk dasarnya, misalnya "berlari" menjadi "lari", sehingga analisis dapat lebih efektif dengan hanya menggunakan kata dasar. Langkah-langkah ini penting dalam pemrosesan teks agar data yang digunakan lebih bersih, seragam, dan siap untuk digunakan dalam analisis lebih lanjut atau model pembelajaran mesin.

Review Text	cleaning	case_folding	normalisasi	tokenize	stopword removal	stemming_data
Template tersedia seksual keinginan keren banget	[template, tersedia, seksual, keinginan, keren, ...]	[template, tersedia, seksual, keren, banget]	template sedia seksual keren banget			
Ini Aplikasi yang Kreatif dan Kren abis	Ini Aplikasi yang Kreatif dan Kren abis	ini aplikasi yang kreatif dan kren abis	ini aplikasi yang kreatif dan keren habis	[ini, aplikasi, yang, kreatif, dan, keren, habis]	[aplikasi, kreatif, keren, habis]	aplikasi kreatif keren habis

Gambar 6. Hasil Preprocessing

Berikutnya melakukan penampilan wordcloud kembali untuk membedakan antara sebelum dilakukan preprocessing dan sesudah dilakukan tahapan-tahapan preprocessing.

```
import pandas as pd
import numpy as np
from PIL import Image
from wordcloud import WordCloud, STOPWORDS, ImageColorGenerator
import matplotlib.pyplot as plt
```

Gambar 7. Script Membuat Wordcloud



Gambar 8. Wordcloud setelah preprocessing

Gambar di atas merupakan word cloud, yaitu representasi visual dari kata-kata yang sering muncul dalam suatu kumpulan teks. Ukuran kata dalam word cloud menunjukkan frekuensi kemunculannya—kata yang lebih besar berarti lebih sering muncul dibandingkan kata yang lebih kecil. Dalam gambar ini, kata "bagus", "pro", "capcut", dan "iklan" terlihat lebih besar, menunjukkan bahwa kata-kata tersebut sering digunakan dalam data yang dianalisis. Ini mengindikasikan bahwa teks yang digunakan kemungkinan besar berasal dari ulasan aplikasi CapCut, dengan banyak pengguna yang menyebutkan fitur "pro", membahas kualitas aplikasi, serta mengomentari keberadaan "iklan". Word cloud seperti ini sering digunakan dalam analisis teks untuk mendapatkan wawasan cepat tentang tema atau sentimen dalam kumpulan data. Gambar tersebut merupakan diagram batang yang menampilkan kata-kata yang paling sering muncul dalam suatu kumpulan teks. Sumbu horizontal (X) menunjukkan kata-kata yang dianalisis, sementara sumbu vertikal (Y) menunjukkan jumlah kemunculan masing-masing kata. Judul grafik "Kata yang Sering Muncul" menunjukkan bahwa ini adalah hasil analisis frekuensi kata dalam suatu teks, kemungkinan dari ulasan atau komentar pengguna.

Berdasarkan diagram, kata "bagus" memiliki jumlah kemunculan tertinggi, yaitu 1617 kali, diikuti oleh kata "pro" (1093), "iklan" (831), "aplikasi" (725), dan kata-kata lainnya dengan frekuensi yang lebih rendah. Hal ini menunjukkan bahwa dalam dataset yang dianalisis, pengguna sering menggunakan kata "bagus" untuk menggambarkan pengalaman mereka, serta menyebutkan kata "pro", "iklan", dan "aplikasi", yang mungkin berkaitan dengan fitur premium atau pengalaman pengguna dengan iklan dalam aplikasi. Grafik ini membantu dalam memahami tren kata yang

sering muncul dan dapat digunakan untuk analisis sentimen atau pemahaman lebih lanjut mengenai opini pengguna.



Gambar 9. Frekuensi Kata Setelah Preprocessing

3. PELABELAN DATA

Labeling data sentimen adalah proses memberi label pada data teks berdasarkan sentimen atau emosi yang terkandung di dalamnya. Label ini digunakan dalam analisis sentimen, terutama dalam pelatihan model machine learning atau pemrosesan bahasa alami (NLP). Biasanya, data sentimen dikategorikan ke dalam beberapa kelas, seperti:

1. Sentimen Positif – Menunjukkan kepuasan atau pengalaman baik, misalnya: "Aplikasi ini sangat bagus dan mudah digunakan."
2. Sentimen Negatif – Menunjukkan ketidakpuasan atau pengalaman buruk, misalnya: "Terlalu banyak iklan dan sering error."
3. Sentimen Netral – Tidak mengandung opini yang jelas, misalnya: "Saya baru saja mengunduh aplikasi ini."

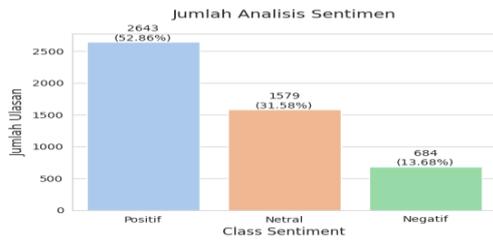
Proses labeling dilakukan menggunakan kamus leksikon yang didapatkan dari github yang dilakukan pemanggilan url untuk masuk ke dalam script python.

```
! Unduh kamus leksikon positif dan negatif dari Github
! https://github.com/mug12900/positive-negative-words
positive_url = "https://raw.githubusercontent.com/mug12900/positive-negative-words/refs/heads/main/positivewords"
negative_url = "https://raw.githubusercontent.com/mug12900/negative-words/refs/heads/main/negative.csv"
positive_lexicon = sat.read_csv(positive_url, sep="t", header=None)
negative_lexicon = sat.read_csv(negative_url, sep="t", header=None)
```

Gambar 10. Labeling Leksikon

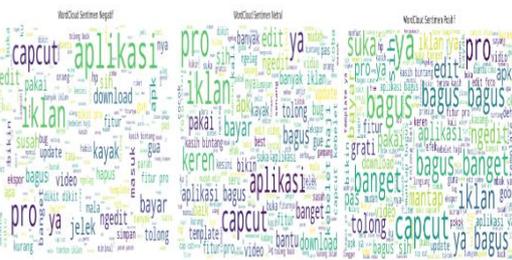
normalisasi	tokenize	stopword removal	stemming_data	Score	Sentiment
template tersedia sesuai keinginan keren banget	['template', 'tersedia', 'sesuai', 'keinginan', 'keren', 'banget']	['template', 'tersedia', 'sesuai', 'keren', 'b...']	template sedia sesuai keren banget	3	Positif
ini aplikasi yang kreatif dan keren habis	['ini', 'aplikasi', 'yang', 'kreatif', 'keren', 'habis', 'dan', '...']	['aplikasi', 'yang', 'kreatif', 'keren', 'habis']	aplikasi kreatif keren habis	0	Netral

Gambar 11. Hasil Labeling



Gambar 12. Plot hasil labeling

Gambar tersebut adalah diagram batang yang menunjukkan hasil analisis sentimen dari sejumlah ulasan pengguna. Sumbu X menunjukkan kelas sentimen, yang terdiri dari tiga kategori: positif, netral, dan negatif, sementara sumbu Y menunjukkan jumlah ulasan dalam masing-masing kategori. Dari grafik, dapat dilihat bahwa mayoritas ulasan memiliki sentimen positif, yaitu sebanyak 2643 ulasan (52.86%), diikuti oleh sentimen netral sebanyak 1579 ulasan (31.58%), dan sentimen negatif sebanyak 684 ulasan (13.68%). Ini menunjukkan bahwa sebagian besar pengguna memberikan ulasan yang positif terhadap produk atau layanan yang dianalisis, sementara hanya sebagian kecil yang memberikan ulasan negatif. Analisis seperti ini berguna untuk memahami persepsi pengguna terhadap suatu aplikasi, layanan, atau produk, sehingga dapat membantu pengembang dalam meningkatkan kualitas dan menangani aspek yang mungkin kurang disukai oleh pengguna.



Gambar 13. Perbandingan Tiga Wordcloud Sentimen

4. TRAINING DAN TEST

Split data dalam penelitian analisis sentimen sangat penting untuk membangun dan mengevaluasi model pembelajaran mesin dengan baik. Proses ini membagi dataset menjadi beberapa bagian, biasanya training set dan testing set, dengan tujuan utama sebagai berikut:

- Melatih Model (Training Set) – Bagian terbesar dari data digunakan untuk melatih model agar dapat mengenali pola dalam

teks dan memahami karakteristik sentimen yang berbeda (positif, negatif, netral).

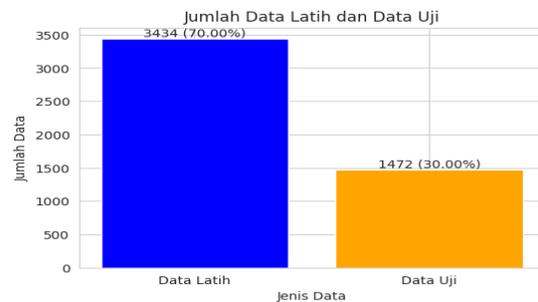
- Mengukur Performa Model (Testing Set) – Sebagian data yang tidak digunakan dalam pelatihan dipakai untuk menguji performa model guna memastikan bahwa model dapat melakukan prediksi dengan baik terhadap data baru yang belum pernah dilihat sebelumnya.

Pembagian data dilakukan dengan rasio 80:20 DAN 70:30 (training:testing). Dengan split data, penelitian analisis sentimen menjadi lebih akurat, karena model dapat diuji secara objektif sebelum diterapkan dalam dunia nyata.

```
from sklearn.model_selection import train_test_split

# Bagi data menjadi data pelatihan dan pengujian
X_train, X_test, y_train, y_test =
train_test_split(df[['stemming_data']], df[['Sentiment']],
test_size=0.3, random_state=42)
```

Gambar 14. Split Data 70:30



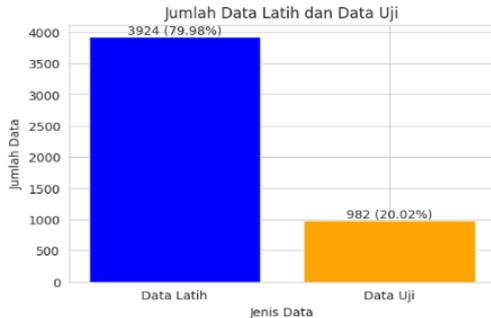
Gambar 15. Hasil Split Data 70:30

Gambar ini adalah diagram batang yang menunjukkan pembagian data dalam proses training (pelatihan) dan testing (pengujian) untuk analisis sentimen atau model machine learning lainnya.

- Sumbu X menunjukkan jenis data, yaitu Data Latih (Training Data) dan Data Uji (Testing Data).
- Sumbu Y menunjukkan jumlah data dalam masing-masing kategori.
- Data Latih berjumlah 3434 sampel (70.00%), yang digunakan untuk melatih model agar dapat mengenali pola dan memahami karakteristik sentimen dalam teks.
- Data Uji berjumlah 1472 sampel (30.00%), yang digunakan untuk mengukur performa model setelah proses pelatihan, memastikan bahwa model dapat bekerja dengan baik terhadap data baru yang belum pernah dilihat sebelumnya.

Pembagian 70:30 ini merupakan metode

umum dalam split data untuk memastikan keseimbangan antara pelatihan yang optimal dan evaluasi yang cukup representatif. Dengan pembagian ini, model dapat belajar dari data yang cukup banyak, sekaligus diuji secara akurat untuk menghindari *overfitting* atau hasil yang bias.

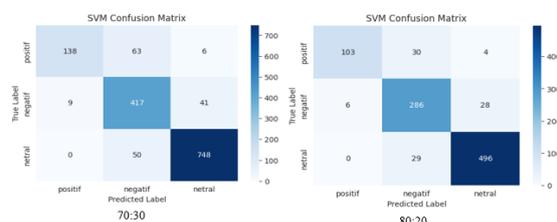


Gambar 16. Hasil Split Data 80:20

Gambar ini adalah diagram batang yang menunjukkan pembagian dataset menjadi data latih (training data) dan data uji (testing data) dalam suatu eksperimen atau penelitian, kemungkinan untuk analisis sentimen atau model machine learning lainnya.

- Sumbu X menunjukkan jenis data: Data Latih dan Data Uji.
- Sumbu Y menunjukkan jumlah sampel dalam masing-masing kategori.
- Data Latih memiliki 3924 sampel (79.98%), yang digunakan untuk melatih model agar dapat memahami pola dan karakteristik data.
- Data Uji memiliki 982 sampel (20.02%), yang digunakan untuk mengevaluasi performa model terhadap data baru yang belum pernah dipelajari sebelumnya.

Pembagian ini menggunakan rasio 80:20, yang merupakan pendekatan umum dalam pembelajaran mesin untuk memastikan model memiliki cukup data untuk belajar, sekaligus dapat diuji dengan data yang cukup representatif. Rasio ini bertujuan untuk menghindari *overfitting*, di mana model terlalu menyesuaikan diri dengan data pelatihan dan kurang mampu menangani data baru secara umum.



Gambar 17. SVM Confusion Matrix

Gambar menampilkan *Confusion Matrix* dari model *Support Vector Machine (SVM)* untuk dua skenario pembagian data: 70:30 (kiri) dan 80:20 (kanan).

Confusion Matrix digunakan untuk mengevaluasi performa model klasifikasi dengan menunjukkan jumlah prediksi benar dan salah untuk setiap kategori.

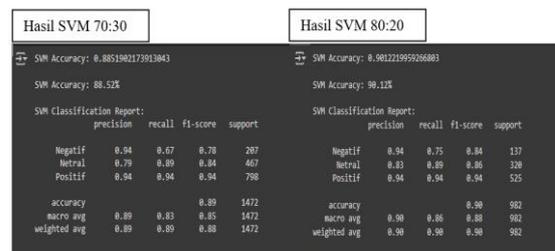
a. *Confusion Matrix* 70:30:

Model memprediksi 138 ulasan positif dengan benar, tetapi salah mengklasifikasikan 63 sebagai negatif dan 6 sebagai netral. Untuk kelas negatif, 417 prediksi benar, tetapi 9 salah diklasifikasikan sebagai positif dan 41 sebagai netral. Untuk kelas netral, model memprediksi 748 dengan benar, namun 50 salah sebagai negatif.

b. *Confusion Matrix* 80:20:

Model memprediksi 103 ulasan positif dengan benar, tetapi salah mengklasifikasikan 30 sebagai negatif dan 4 sebagai netral. Untuk kelas negatif, 286 prediksi benar, tetapi 6 salah diklasifikasikan sebagai positif dan 28 sebagai netral. Untuk kelas netral, model memprediksi 496 dengan benar, namun 29 salah sebagai negatif.

Dari perbandingan ini, terlihat bahwa perubahan rasio training:testing (dari 70:30 menjadi 80:20) berdampak pada jumlah kesalahan prediksi model. Dengan lebih banyak data latih (80:20), model tampaknya tetap mempertahankan performa yang baik, tetapi jumlah prediksi salah pada beberapa kelas masih terjadi. Evaluasi lebih lanjut diperlukan untuk melihat metrik seperti akurasi, presisi, recall, dan F1-score guna menentukan pembagian data terbaik untuk model ini.



Gambar 18. Hasil Tingkat Akurasi

Gambar menampilkan hasil evaluasi model *Support Vector Machine (SVM)* pada dua

skenario pembagian data: 70:30 (kiri) dan 80:20 (kanan). Perbandingan utama antara kedua hasil ini adalah sebagai berikut:

1. Akurasi Model
 - a) Pada 70:30, akurasi model adalah 88.52%.
 - b) Pada 80:20, akurasi meningkat menjadi 90.12%.
 - c) Peningkatan akurasi menunjukkan bahwa dengan lebih banyak data latih (80% dibandingkan 70%), model mampu belajar lebih baik dan menghasilkan prediksi yang lebih akurat.
2. Evaluasi Metrik (*Precision*, *Recall*, dan *F1-Score*)
 - a) Negatif: *Precision* tetap di 0.94, tetapi *recall* meningkat dari 0.67 (70:30) menjadi 0.75 (80:20), yang berarti model lebih baik dalam mengenali ulasan negatif.
 - b) Netral: *Precision* dan *recall* mengalami sedikit peningkatan, dari 0.79/0.89 (70:30) menjadi 0.83/0.89 (80:20).
 - c) Positif: Performa tetap konsisten dengan *precision* dan *recall* di angka 0.94 pada kedua skenario.
 - d) *F1-score* rata-rata juga meningkat dari 0.88 (70:30) menjadi 0.90 (80:20), menunjukkan keseimbangan yang lebih baik antara *precision* dan *recall*.

Hasilnya adalah dengan menggunakan lebih banyak data latih (80:20), model menjadi lebih akurat dan lebih mampu mengenali ulasan negatif dengan lebih baik dibandingkan dengan pembagian 70:30. Namun, perbedaannya tidak terlalu besar, sehingga perlu dipertimbangkan faktor lain seperti overfitting, performa pada data baru, dan efisiensi waktu pelatihan model.

KESIMPULAN

Berdasarkan hasil analisis sentimen menggunakan *Support Vector Machine (SVM)*, mayoritas ulasan pengguna terhadap aplikasi *CapCut* bersentimen positif, dengan persentase sebesar 52,86% dari total ulasan yang dianalisis. Sentimen netral berada di posisi kedua dengan 31,58%, sedangkan sentimen negatif hanya 13,68%. Hasil ini menunjukkan bahwa sebagian besar pengguna memiliki pengalaman yang baik dengan aplikasi *CapCut*, yang tercermin dari banyaknya kata-kata positif yang muncul dalam analisis, seperti "bagus," "pro," dan "aplikasi". Namun, masih terdapat sebagian

kecil ulasan negatif yang kemungkinan besar berkaitan dengan kendala teknis, fitur yang terbatas, atau kebijakan pembayaran dalam aplikasi.

Dengan akurasi model *SVM* yang mencapai 90,12%, penelitian ini membuktikan bahwa metode ini efektif dalam mengklasifikasikan opini pengguna terhadap aplikasi *CapCut*. Dari hasil analisis, dapat disimpulkan bahwa aplikasi ini secara umum diterima dengan baik oleh pengguna, tetapi tetap ada aspek-aspek yang bisa diperbaiki untuk meningkatkan pengalaman pengguna. Hasil penelitian ini dapat digunakan oleh pengembang *CapCut* untuk memahami kebutuhan pengguna dan mengoptimalkan fitur aplikasi berdasarkan umpan balik yang diberikan. Selain itu, metode yang digunakan dalam penelitian ini juga dapat diterapkan dalam berbagai analisis sentimen lain, seperti ulasan produk, layanan pelanggan, dan pemantauan opini di media sosial.

REFERENSI

- Ahmed Khan, T., Sadiq, R., Shahid, Z., Alam, M. M., & Mohd Su'ud, M. (2024). Sentiment Analysis using Support Vector Machine and Random Forest. *Journal of Informatics and Web Engineering*, 3(1), 67–75.
<https://doi.org/10.33093/jiwe.2024.3.1.5>
- Aryanti, R., Misriati, T., & Sagiyo, A. (2023). Analisis Sentimen Aplikasi Primaku Menggunakan Algoritma Random Forest dan SMOTE untuk Mengatasi Ketidakseimbangan Data. *Journal of Computer System and Informatics (JoSYC)*, 5(1), 218–227.
<https://doi.org/10.47065/josyc.v5i1.4562>
- Dwi, E., Wardani, K., Yo, F. F., Meylugita, W. N., Katolik, U., & Charitas, M. (2025). IMPLEMENTASI ALGORITMA NAÏVE BAYES UNTUK ANALISIS ULASAN IMPLEMENTATION OF THE NAIVE BAYES ALGORITHM FOR USER REVIEW, 4(1), 13–24.
- Friska Aditia Indriyani, Ahmad Fauzi, & Sutan Faisal. (2023). Analisis sentimen aplikasi tiktok menggunakan algoritma naïve bayes dan support vector machine. *TEKNOSAINS: Jurnal Sains, Teknologi dan Informatika*, 10(2), 176–184.
<https://doi.org/10.37373/tekno.v10i2.41>

- Han, K. X., Chien, W., Chiu, C. C., & Cheng, Y. T. (2020). Application of support vector machine (SVM) in the sentiment analysis of twitter dataset. *Applied Sciences (Switzerland)*, 10(3). <https://doi.org/10.3390/app10031125>
- Indrayanto, C. G., Ratnawati, D. E., & Rahayudi, B. (2023). Analisis Sentimen Data Ulasan Pengguna Aplikasi MyPertamina di Indonesia pada Google Play Store menggunakan Metode Random Forest. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 7(3), 1131–1139. Diambil dari <http://j-ptiik.ub.ac.id>
- Khaira, U., Aryani, R., & Hardian, R. W. (2023). Komparasi Algoritma Naïve Bayes Dan Support Vector Machine (SVM) Pada Analisis Sentimen Kebijakan Kemdikbudristek Mengenai Kuota Internet Selama Covid-19. *Jurnal PROCESSOR*, 18(2), 272–285. <https://doi.org/10.33998/processor.2023.18.2.897>
- Machmud, A., Wibisono, B., & Suryani, N. (2025). Analisis Sentimen Cyberbullying Pada Komentar X Menggunakan Metode Naïve Bayes, 5(1).
- Muhammadi, R. H., Laksana, T. G., & Arifa, A. B. (2022). Combination of Support Vector Machine and Lexicon-Based Algorithm in Twitter Sentiment Analysis. *Khazanah Informatika : Jurnal Ilmu Komputer dan Informatika*, 8(1), 59–71. <https://doi.org/10.23917/khif.v8i1.15213>
- Puji Astuti, A., Alam, S., & Jaelani, I. (2022). Komparasi Algoritma Support Vector Machine dengan Naive Bayes Untuk Analisis Sentimen Pada Aplikasi BRImo. *Jurnal Bangkit Indonesia*, 11(2), 1–6. <https://doi.org/10.52771/bangkitindonesia.v11i2.196>
- Purnamawati, A., Winarto, M. N., & Mailasari, M. (2023). Analisis Sentimen Aplikasi TikTok menggunakan Metode BM25 dan Improved K-NN Fitur Chi-Square. *Jurnal Komtika (Komputasi dan Informatika)*, 7(1), 97–105. <https://doi.org/10.31603/komtika.v7i1.8938>
- Romadoni, F., Umaidah, Y., & Sari, B. N. (2020). Text Mining Untuk Analisis Sentimen Pelanggan Terhadap Layanan Uang Elektronik Menggunakan Algoritma Support Vector Machine. *Jurnal Sisfokom (Sistem Informasi dan Komputer)*, 9(2), 247–253. <https://doi.org/10.32736/sisfokom.v9i2.903>
- Vincent, R., Maulana, I., & Komarudin, O. (2024). Perbandingan Klasifikasi Naive Bayes Dan Support Vector Machine Dalam Analisis Sentimen Dengan Multiclass Di Twitter. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 7(4), 2496–2505. <https://doi.org/10.36040/jati.v7i4.7152>
- Wankhade, M., Rao, A. C. S., & Kulkarni, C. (2022). *A survey on sentiment analysis methods, applications, and challenges. Artificial Intelligence Review* (Vol. 55). Springer Netherlands. <https://doi.org/10.1007/s10462-022-10144-1>
- Yuniar, P., & Kismiantini. (2023). Analisis Sentimen Ulasan pada Gojek Menggunakan Metode Naive Bayes. *Statistika*, 23(2), 164–175. <https://doi.org/10.29313/statistika.v23i2.2353>