



KOMPARASI METODE NAÏVE BAYES DAN SUPPORT VECTOR MACHINE PADA ANALISIS SENTIMEN PENGGUNA APLIKASI TIKTOK

Muhammad Zainul Ramadhan^{1*}, Marlina², Fani Nurona Cahya³

^{1,2} Fakultas Teknik dan Informatika, Universitas Bina Sarana Informatika

E-mail: zainul.ramadhan125@gmail.com

ABSTRACT

The growth of social media has established TikTok as a widely used platform for sharing opinions, experiences, and perceptions through diverse content. The abundance of user opinions generates textual data that can be leveraged to understand sentiment trends regarding the TikTok application. Consequently, a classification method capable of effectively identifying user sentiment is required. This study aims to compare the performance of Naïve Bayes and Support Vector Machine (SVM) methods in classifying TikTok user sentiment into three categories: positive, neutral, and negative. Research data—consisting of 2,013 tweets in Indonesian—was obtained via web scraping from the X/Twitter platform. The research process involved data collection; preprocessing (cleaning, case folding, tokenizing, normalization, filtering, and stemming); data labeling; word weighting using Term Frequency-Inverse Document Frequency (TF-IDF); splitting data into training and testing sets (70:30 ratio); classification using Naïve Bayes and SVM; and evaluation using a Confusion Matrix based on accuracy, precision, recall, and F1-score metrics. The results indicate that Naïve Bayes achieved an accuracy of 75.66%, while SVM achieved 80.46%. These findings demonstrate that SVM delivers superior classification performance compared to Naïve Bayes. However, both methods still face challenges in classifying negative sentiment due to imbalanced data distribution.

Keywords: *Sentiment Analysis, TikTok, Naïve Bayes, Support Vector Machine, TF-IDF, Confusion Matrix*

ABSTRAK

Perkembangan media sosial menjadikan TikTok sebagai salah satu platform yang banyak digunakan masyarakat untuk menyampaikan opini, pengalaman, dan persepsi melalui berbagai konten. Banyaknya opini pengguna menghasilkan data teks yang dapat dimanfaatkan untuk mengetahui kecenderungan sentimen terhadap aplikasi TikTok. Oleh karena itu, diperlukan metode klasifikasi yang mampu mengidentifikasi sentimen pengguna secara efektif. Penelitian ini bertujuan untuk membandingkan kinerja metode Naïve Bayes dan Support Vector Machine (SVM) dalam mengklasifikasikan sentimen pengguna aplikasi TikTok ke dalam tiga kategori, yaitu positif, netral, dan negatif. Data penelitian

Article History

Received: Oktober 2026

Reviewed: Oktober 2026

Published: Oktober 2026

Plagiarism Checker No
237.u873247.fh8374

Copyright : Author

Publish by : Kohesi



This work is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](https://creativecommons.org/licenses/by-nc/4.0/)



diperoleh melalui teknik web scraping dari platform X/Twitter dalam bahasa Indonesia dengan jumlah 2.013 data tweet. Tahapan penelitian meliputi pengumpulan data, preprocessing yang terdiri dari cleaning, case folding, tokenizing, normalisasi, filtering, dan stemming, kemudian pelabelan data, pembobotan kata menggunakan Term Frequency-Inverse Document Frequency (TF-IDF), pembagian data latih dan data uji sebesar 70%:30%, klasifikasi menggunakan Naïve Bayes dan SVM, serta evaluasi menggunakan Confusion Matrix dengan metrik accuracy, precision, recall, dan F1-score. Hasil penelitian menunjukkan bahwa Naïve Bayes memperoleh accuracy sebesar 75,66%, sedangkan SVM memperoleh accuracy sebesar 80,46%. Hasil tersebut menunjukkan bahwa SVM memiliki performa klasifikasi yang lebih tinggi dibandingkan Naïve Bayes. Namun, kedua metode masih mengalami kendala dalam mengklasifikasikan sentimen negatif karena distribusi data yang tidak seimbang.

Kata Kunci: Analisis Sentimen, TikTok, Naïve Bayes, Support Vector Machine, TF-IDF, Confusion Matrix.

1. PENDAHULUAN

Teknologi *online* terus mengalami perkembangan pesat pada masa sekarang, sejalan dengan peningkatan yang signifikan dalam volume data digital pengelolaan data digital telah menjadi keharusan dalam mengubahnya menjadi pengetahuan yang memiliki nilai. Saat ini, salah satu *platform* terbesar yang berkontribusi besar dalam menciptakan data digital adalah media sosial. Karena melalui media sosial, orang dapat membuat interaksi sosial antar pengguna dari jarak jauh dan berbagi berita, momen, serta pengalaman yang dapat dibagikan di media sosial orang dan organisasi kini menggunakan media sosial sebagai wujud eksistensinya di dunia maya (Pratama et al., 2024)

Di Indonesia, *platform* media sosial yang populer digunakan meliputi *TikTok*, *Facebook*, *Instagram*, dan *YouTube*. Beberapa platform tersebut termasuk dalam kategori platform dengan jumlah pengguna terbanyak di dunia saat ini. Di antara berbagai aplikasi media sosial yang sedang *tren*, *TikTok* menonjol sebagai salah satu yang paling dominan. Beragam kalangan masyarakat memanfaatkan *platform* ini untuk berbagai keperluan (Hilmi, 2024)

TikTok mulai menembus pasar di Indonesia pada 2018-2019. *TikTok* terus berkembang pesat selama pandemi *COVID-19*. Popularitas *TikTok* melonjak dengan cepat, terutama di kalangan generasi muda. Indonesia menjadi salah satu pasar terbesar *TikTok* dengan jutaan pengguna aktif. Jumlah pengguna *TikTok* di Indonesia tembus lebih dari 180 juta pengguna per Maret 2025. Dengan ini, Indonesia menjadi negara dengan pengguna *tiktok* paling banyak di Asia Tenggara *Southeast Asia*(SEA) (Galuh Putri Riyanto, 2025)

Aplikasi *TikTok* pada awalnya merupakan media sosial untuk sarana membagikan video pendek dengan tampilan berbagai *filter* dan *music* yang menarik. Namun seiring berjalannya waktu *TikTok* merilis fitur *social e-commerce* dengan nama *TikTok Shop* pada tanggal 17 april 2021 (Septiani et al., 2024).



Generasi Z dinilai sebagai segmen target yang ideal bagi platform *online shop* karena karakteristik mereka yang adaptif terhadap teknologi. Dalam konteks platform *TikTok Shop*, konsumen dapat mencari berbagai kebutuhan seperti pakaian, alas kaki, perlengkapan rumah tangga, kosmetik, perlengkapan sekolah dan kantor, serta berbagai produk lainnya. Dengan demikian, konsumen dapat mempertimbangkan platform mana yang paling sesuai dengan preferensi mereka. Lebih lanjut, penjual juga dapat melakukan siaran langsung (*live streaming*), yang memungkinkan konsumen berinteraksi secara langsung melalui kolom komentar yang tersedia. Konten promosi, khususnya yang bertema kecantikan, terbukti sangat menarik minat konsumen (Mirandini et al., 2024)

Seiring dengan perkembangan tersebut, data yang dihasilkan dari aktivitas pengguna di media sosial memiliki potensi besar untuk dianalisis, khususnya dalam memahami opini publik. Salah satu pendekatan yang banyak digunakan adalah analisis sentimen, yang bertujuan untuk mengidentifikasi kecenderungan opini pengguna terhadap suatu topik tertentu. Hal ini menjadi semakin relevan karena media sosial tidak hanya digunakan sebagai sarana komunikasi, tetapi juga sebagai sumber informasi yang mencerminkan persepsi masyarakat terhadap berbagai fenomena, termasuk kebijakan atau regulasi yang diterapkan oleh pemerintah maupun platform digital itu sendiri.

Analisis Sentimen Adalah Proses untuk mengidentifikasi ekspresi perasaan dalam teks dengan menganalisis data teks untuk memahami sentiment yang terkandung didalamnya. Teknik ini dapat di terapkan untuk menganalisis opini dalam ulasan pengguna di *Google Play Store* atau *platform X* melalui pemrosesan informasi. Dalam penelitian analisis sentiment ini penulis melaksanakan Analisa afeksi memakai asumsi pengguna tiktok. Analisis sentimen dilakukan untuk mengklasifikasi teks ke dalam kategori *positif*, *negatif*, dan *netral* berdasarkan opini pengguna. Dalam melaksanakan pengelompokan penulis memakai 2 metode yaitu Naïve Bayes dan Support Vector Machine (Indriyani et al., 2023)

2. METODE PENELITIAN

Penelitian ini menggunakan metode kuantitatif dengan pendekatan eksperimen untuk membandingkan kinerja algoritma Naïve Bayes dan *Support Vector Machine* (SVM) dalam analisis sentimen pengguna aplikasi TikTok. Data penelitian diperoleh melalui teknik *web scraping* dari platform X/Twitter dengan jumlah 2.013 data *tweet* berbahasa Indonesia. Tahapan penelitian meliputi pengumpulan data, *preprocessing* yang terdiri atas *cleaning*, *case folding*, *tokenizing*, *normalization*, *filtering*, dan *stemming*, serta pelabelan sentimen ke dalam kategori positif, netral, dan negatif. Selanjutnya, data diubah menjadi representasi numerik menggunakan pembobotan kata *Term Frequency-Inverse Document Frequency* (TF-IDF) dan dibagi menjadi data latih sebesar 70% serta data uji sebesar 30%. Proses klasifikasi dilakukan menggunakan algoritma Naïve Bayes dan SVM secara terpisah dengan bantuan bahasa pemrograman Python melalui Google Colaboratory. Kinerja kedua algoritma kemudian dievaluasi menggunakan *Confusion Matrix* berdasarkan metrik *accuracy*, *precision*, *recall*, dan *F1-score* untuk menentukan metode yang memiliki performa terbaik dalam mengklasifikasikan sentimen pengguna aplikasi TikTok.

3. HASIL DAN PEMBAHASAN

Pengumpulan Data

Sumber data dalam penelitian ini berasal dari *Tweet* pengguna aplikasi *Tiktok* yang diambil langsung melalui platform *X/Twitter*. Pengumpulan data dilakukan dengan menggunakan teknik *web scrapping* menggunakan *library Python* yang telah disediakan oleh pihak *Google* dan menyimpannya ke dalam file *CSV*. Data ini berisi teks *tweet* serta atribut lainnya rating pengguna dan user id pengguna.

```
from apify_client import ApifyClient

# Initialize the ApifyClient with your API token
client = ApifyClient("<YOUR_API_TOKEN>")
# Prepare the Actor input
run_input = {
    "filter:blue_verified": False,
    "filter:consumer_video": False,
    "filter:has_engagement": False,
    "filter:hashtags": False,
    "filter:images": False,
    "filter:links": False,
    "filter:media": False,
    "filter:mentions": False,
    "filter:native_video": False,
    "filter:nativeretweets": False,
    "filter:news": False,
    "filter:pro_video": False,
    "filter:quote": False,
    "filter:replies": False,
    "filter:safe": False,
    "filter:spaces": False,
    "filter:twimg": False,
    "filter:verified": False,
    "filter:videos": False,
    "filter:vine": False,
    "include:nativeretweets": False,
    "lang": "en",
    "maxItems": 1000,
    "queryType": "latest",
    "twitterContent": "Tiktokshop Indonesia"
}

# Run the Actor and wait for it to finish
run = client.actor("CJdippsian9uRf0oo").call(run_input=run_input)

# Fetch and print Actor results from the run's dataset (if there are any)
for item in client.dataset(run["defaultDatasetId"]).iterate_items():
    print(item)

df = pd.DataFrame(items)
df.to_csv("kuuranglebih.csv", index=False)
```

Gambar 1. Teknik Web Scrapping
Sumber: Hasil Penelitian, 2026

Data yang diambil berupa *tweet x* dalam Bahasa Indonesia beserta user id, tweet dan tanggal tweet. Setelah data berhasil di *scraping*, data disimpan kedalam *CSV* dengan total sebanyak 2013 data, seperti yang terlihat pada table dibawah ini:

	id	url	text	createdat	author,profilePicture	retweetCount	replyCount	likeCount
0	206731899737309396	https://x.com/Kika_259/status/206731899737309396	Gw uth atur algoritma tiktok gw biar ga ada M...	Wed Jun 17 18:50:32 +0000 2026	https://pbs.twimg.com/profile_images/147386491...	0	1	0
1	2067315287087882706	https://x.com/maher/status/2067315287087882706	5.50Can++ Testimoni Sejak 2018w  WA https://...	Wed Jun 17 18:35:47 +0000 2026	https://pbs.twimg.com/profile_images/206718047...	0	0	0
2	2067314558075908196	https://x.com/Kristine_K0f/status/2067314558075908196	Layanan All BerGagasan Fastprocess - 550_	Wed Jun 17 18:32:54 +0000 2026	https://pbs.twimg.com/profile_images/206674754...	0	0	0
3	2067313643679785239	https://x.com/grached/status/2067313643679785239	7 Step Scroll Desain uth paling murah 7th - TC	Wed Jun 17 18:28:16 +0000 2026	https://pbs.twimg.com/profile_images/206722691...	0	0	0
4	2067313366318867236	https://x.com/breacher/status/2067313366318867236	.Oru Jaman Desain Ter murah kuta dg mesai	Wed Jun 17 18:28:10 +0000 2026	https://pbs.twimg.com/profile_images/206722737...	0	0	0
2010	1706281648491544992	https://x.com/PengapaGawand/status/1706281648491544992	Timnas Garuda Indonesia u17 kunjungi stadion II	Mon Sep 25 12:17:40 +0000 2023	https://pbs.twimg.com/profile_images/166235274...	0	0	0
2011	1706281271960510930	https://x.com/veradodd/status/1706281271960510930	ERASPACE IS BACKKK    kami akan m...	Mon Sep 25 12:15:31 +0000 2023	https://pbs.twimg.com/profile_images/178439285...	2	1	12
2012	1706279363728327054	https://x.com/andri/status/1706279363728327054	Waktu Backup di Indonesia pa curhat akan Tika...	Mon Sep 25 12:07:25 +0000 2023	https://pbs.twimg.com/profile_images/200620950...	26	2	262
2013	1706274807720087615	https://x.com/greenmovers/status/1706274807720087615	Gimana Semeru?  Witch tengah kera	Mon Sep 25 11:49:49 +0000 2023	https://pbs.twimg.com/profile_images/131701698...	4	1	21

Gambar 2. Sampel Data
Sumber: Hasil Penelitian, 2026

Preprocessing Data

Setelah data diperoleh, Langkah berikutnya adalah melakukan *preprocessing*, yang meliputi proses normalisasi teks (*case folding*, pembersihan data seperti penghapusan *URL*, tanda baca, angka dan *symbol* dilanjutkan dengan tokenisasi untuk memisahkan kalimat menjadi kata-kata individual, serta penghapusan kata-kata umum(*stopwords*).

```
Sample processing text:
Original: Gw udh atur algoritma tiktok gw biar ga ada klip tentang demo karena gw ga mau mati muda darah tinggi gara-gara baca komentar orang tolol
Gw setting biar fyp about make up. Lalu ada vlogger Malay yg
Cleaned: sudah atur algoritma tiktok biar klip demo mau mati muda darah tinggi gara gara baca komentar orang tolol saya setting biar for your page about make up lalu ada vlogger mala
```

Gambar 3. Preprocesing Data
 Sumber: Hasil Penelitian, 2026

Labeling Data

Setelah data di bersihkan, setiap *tweet* diberi label *sentiment* berdasarkan kalimat yang terdapat didalam *tweet*. Proses labeling ini dilakukan secara otomatis dengan *library Python pada Google Colab* yang telah disediakan oleh pihak *Google*.

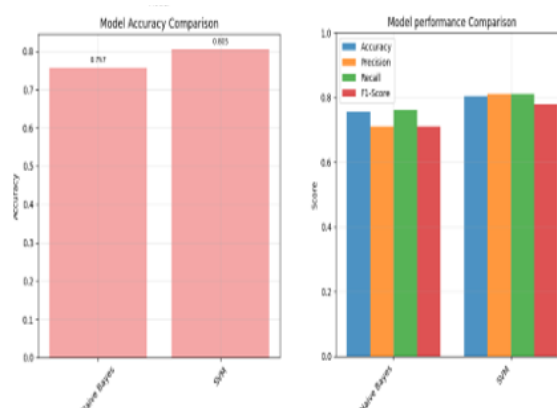
```
words = text.split()
pos_score = sum(1 for word in words if word in positive_words)
neg_score = sum(1 for word in words if word in negative_words)

if pos_score > neg_score:
    return 'positive'
elif pos_score < neg_score:
    return 'negative'
else:
    return 'neutral'
```

Gambar 4. Proses Labeling
 Sumber: Hasil Penelitian, 2026

Implementasi Naïve Bayes dan SVM

Langkah ini bertujuan untuk mempresentasikan komparasi *Naïve Bayes* dan *SVM* menghasilkan performance dan *accuracy* dalam bentuk diagram yang dapat dilihat pada gambar dibawah ini.



Gambar 5. Hasil Implementasi

Adapun penyajian seluruh review *tweet* dalam bentuk *WordCloud* yang dapat dilihat pada gambar berikut ini.



Gambar 6. WorldCloud

Splitting

Tahapan ini bertujuan untuk membagi dataset menjadi 2 bagian, yaitu data latih (*training*) sebesar 70% dan data uji (*testing*) sebesar 30%, guna memungkinkan pelatihan dan pengujian model secara terpisah

```
try:
    X_train, X_test, y_train, y_test = train_test_split(
        X, y,
        test_size=0.3,
        random_state=42,
        stratify=y
    )
    split_method = "Stratified Split"
except ValueError:
    X_train, X_test, y_train, y_test = train_test_split(
        X, y,
        test_size=0.3,
        random_state=42
    )
    split_method = "Regular Split (tanpa stratify)"

total_data = len(df)
jumlah_train = len(X_train)
jumlah_test = len(X_test)
```

Gambar 7. Proses Splitting

Proses di atas menghasilkan 2 subset data, Dimana data latih digunakan untuk membangun model klasifikasi, sedangkan data uji digunakan untuk mengevaluasi dalam gambar berikut.

Total data	: 2013
Jumlah data latih (70%)	: 1409
Jumlah data uji (30%)	: 604

Gambar 8. Hasil Splitting
Sumber: Hasil Penelitian, 2026

Evaluasi Data

Evaluasi data merupakan tahap akhir dari analisis sentiment, yang bertujuan untuk mengukur kinerja model dalam mengklasifikasikan *tweet* ke dalam kategori sentiment akursi, presisi, recal dan f1-score, untuk memastikan bahwa model yang dihasilkan memiliki perfoma yang optimal sebelum diimplentasikan lebih lanjut. Proses evaluasi dapat dilihat pada gambar dibawah.

```
# ===== 2. TRAINING MODEL =====
# --- Naive Bayes ---
nb_model = MultinomialNB(alpha=1.0) # Using MultinomialNB for text classification, alpha=1.0 for Laplace smoothing
nb_model.fit(X_train_tfidf, y_train)
y_pred_nb = nb_model.predict(X_test_tfidf)

# --- SVM (tugas scaling, dengan kernel linear) ---
# Aligning with previous successful configuration
svm_model = SVC(kernel='linear', C=1.0, random_state=42) # Changed kernel to linear, removed StandardScaler
svm_model.fit(X_train_tfidf, y_train)
y_pred_svm = svm_model.predict(X_test_tfidf)

# ===== 3. CONFUSION MATRIX =====
# Confusion Matrix Naive Bayes
cm_nb = confusion_matrix(y_test, y_pred_nb)
# Confusion Matrix SVM
cm_svm = confusion_matrix(y_test, y_pred_svm)

# ===== 4. VISUALISASI =====
fig, axes = plt.subplots(1, 2, figsize=(14, 6))

# Get unique labels from y_test for consistent xticklabels/yticklabels
unique_labels = np.unique(y_test)

# --- Plot Confusion Matrix Naive Bayes ---
sns.heatmap(cm_nb, annot=True, fmt='d', cmap='Blues',
            xticklabels=unique_labels, yticklabels=unique_labels,
            ax=axes[0])
axes[0].set_title('Naive Bayes Confusion Matrix Accuracy: (accuracy_score(y_test, y_pred_nb):.2%)',
                fontsize=14, fontweight='bold')
axes[0].set_xlabel('Predicted', fontsize=12)
axes[0].set_ylabel('Actual', fontsize=12)

# --- Plot Confusion Matrix SVM ---
sns.heatmap(cm_svm, annot=True, fmt='d', cmap='Blues',
            xticklabels=unique_labels, yticklabels=unique_labels,
            ax=axes[1])
axes[1].set_title('SVM Confusion Matrix Accuracy: (accuracy_score(y_test, y_pred_svm):.2%)',
                fontsize=14, fontweight='bold')
axes[1].set_xlabel('Predicted', fontsize=12)
axes[1].set_ylabel('Actual', fontsize=12)

plt.tight_layout()
plt.show()

# ===== 5. METRIK EVALUASI LENGAP =====
print("\n📊 Accuracy")
print("\n📊 Naive Bayes: (accuracy_score(y_test, y_pred_nb):.4f) ((accuracy_score(y_test, y_pred_nb):.2%))")
print("\n📊 SVM : (accuracy_score(y_test, y_pred_svm):.4f) ((accuracy_score(y_test, y_pred_svm):.2%))")

# Classification Report
print("\n📊 Classification Report - Naive Bayes:")
print(classification_report(y_test, y_pred_nb, zero_division=0))

print("\n📊 Classification Report - SVM:")
print(classification_report(y_test, y_pred_svm, zero_division=0))
```

Gambar 9. Proses Evaluasi
 Sumber: Hasil Penelitian, 2026

Proses di atas menghasilkan nilai evaluasi yang mencerminkan seberapa baik model dalam mengenali dan mengklasifikasi sentiment dari *tweet* pengguna, sebagaimana ditampilkan pada gambar dibawah ini.

Classification Report - Naive Bayes:				
	precision	recall	f1-score	support
negative	0.00	0.00	0.00	40
neutral	0.76	0.96	0.85	399
positive	0.75	0.45	0.56	165
accuracy			0.76	604
macro avg	0.50	0.47	0.47	604
weighted avg	0.71	0.76	0.71	604

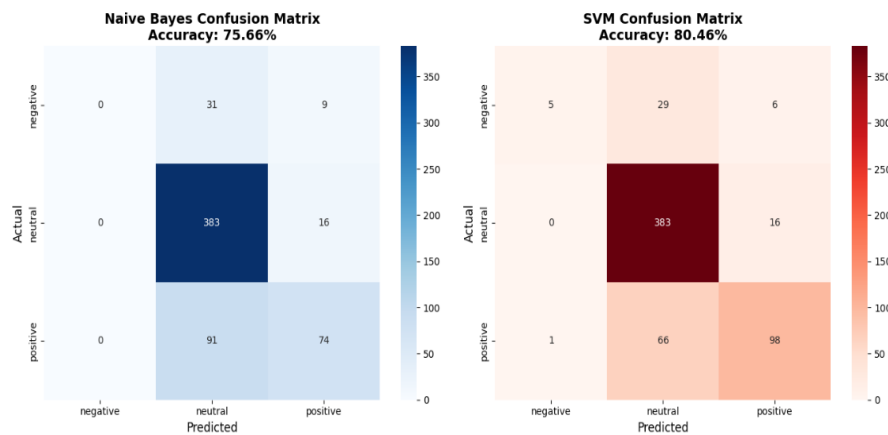
Classification Report - SVM:				
	precision	recall	f1-score	support
negative	0.83	0.12	0.22	40
neutral	0.80	0.96	0.87	399
positive	0.82	0.59	0.69	165
accuracy			0.80	604
macro avg	0.82	0.56	0.59	604
weighted avg	0.81	0.80	0.78	604

Gambar 10. Hasil Evaluasi

Sumber: Hasil Penelitian, 2026

Hasil Penelitian

Hasil Penelitian dianalisis Menggunakan *Confusion Matrix* perbandingan performance *Naïve Bayes* dan *SVM*, yang bertujuan untuk mengevaluasi hasil secara lebih mendalam kinerja model dalam mengklasifikasi kelas *positif*, *negative* dan *netral* pada dataset *tweet* pengguna aplikasi *tiktok*



Gambar 11. *Confusion Matrix*

Berdasarkan hasil pengujian *Confusion Matrix*, *Naïve Bayes* berhasil mengklasifikasikan 74 data sentimen positif dan 383 data sentimen netral secara benar, tetapi tidak mampu mengenali sentimen negatif. Sementara itu, *SVM* berhasil mengklasifikasikan 98 data positif, 383 data netral, dan 29 data negatif secara benar. Hasil evaluasi menunjukkan bahwa *Naïve Bayes* memperoleh akurasi sebesar 76%, sedangkan *SVM* mencapai 80%. Nilai *F1-score* pada kelas positif, netral, dan negatif masing-masing sebesar 0,56; 0,85; dan 0 pada *Naïve Bayes*, serta 0,69; 0,87; dan 0,22 pada *SVM*. Dengan demikian, *SVM* menunjukkan kinerja yang lebih baik secara keseluruhan, meskipun kedua model masih memiliki keterbatasan dalam mengklasifikasikan sentimen negatif.

Hasil Pengujian

Hasil Pengujian dilakukan terhadap 604 data uji yang mencakup nilai *precision*, *recall*, dan *f1-score* untuk masing-masing kelas sentiment. Model klasifikasi yang digunakan dalam penelitian ini adalah *Naïve Bayes* dan *Support Vector Machine*. Setelah proses pelatihan model selesai, dilakukan pengujian terhadap data uji guna mengevaluasi perfoma model dalam mengklasifikasi ulasan menjadi 3 kategori *sentiment*, yaitu *positif*, *netral* dan *negative*. Berikut adalah hasil perhitungan hasil pengujian setiap kelas.

Naïve Bayes

a. Accuracy

$$= \frac{0 + 383 + 74}{604} = \frac{457}{604} = 0,7566 = 75.66\%$$

b. Kelas Positif



True Positive (TP) = 74
False Positive (FP) = 25
False Negative (FN) = 91
Precision:

$$= \frac{TP}{TP + FP} = \frac{74}{74 + 25} = \frac{74}{99} = 0,7475 = 74.75\%$$

Recall:

$$= \frac{TP}{TP + FN} = \frac{74}{74 + 91} = \frac{74}{165} = 0.4485 = 44.85\%$$

F1-Score:

$$= 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} = 2 \times \frac{0.7475 \times 0.4485}{0.7475 + 0.4485} = 0,5606$$

c. Kelas Netral

True Positif (TP) = 383
False Positif (FP) = 122
False Negative (FN) = 16
Precision:

$$= \frac{TP}{TP + FP} = \frac{383}{383 + 122} = \frac{383}{505} = 0.7584 = 75.84\%$$

Recall:

$$= \frac{TP}{TP + FN} = \frac{383}{383 + 16} = \frac{383}{399} = 0,9599 = 95.99\%$$

F1-Score:

$$2 \times \frac{0.7484 \times 0.9599}{0.7484 + 0.9599} = 0,8474 = 84.74\%$$

d. Kelas Negative

True Positive (TP) = 0
False Positive (FP) = 0
False Negative (FN) = 40
Precision:

$$= \frac{TP}{TP + FP} = \frac{0}{0 + 0} = 0$$

Recall:

$$= \frac{TP}{TP + FN} = \frac{0}{0 + 40} = 0$$

F1-Score:

$$= 0 \text{ (karena Precision dan Recall} = 0$$

e. Marco Average

$$\text{Marco Avg} = \frac{1}{n} \sum_{i=1}^n (\text{Precision}_i, \text{Recall}_i, F1_i)$$

$$\text{Marco Avg} = \frac{\text{Nilai Kelas Negative} + \text{Neutral} + \text{Positive}}{3}$$

Dengan nilai:

$$\text{Precision} = \frac{0.00 + 0.7584 + 0.7475}{3} = \frac{1.5059}{3} = 0.5020$$

$$\text{Recall} = \frac{0.000 + 0.9599 + 0.4485}{3} = \frac{1.4084}{3} = 0.4695$$



$$F1 = \frac{0.000 + 0.8474 + 0.5606}{3} = \frac{1.4080}{3} = 0.4693$$

f. Weighted Average

Weighted Average merupakan rata - rata pertimbangan berdasarkan jumlah support:

$$Weight\ Avg = N = \sum_{i=1}^n Support_i$$

$$Weight\ Avg = \frac{Nilai\ Kelas\ Ne + Neu + Pos \times Support}{Total\ Data}$$

Dengan Nilai:

$$Precision = \frac{(0.000 \times 40) + (0.7584 \times 399) + (0.7475 \times 165)}{604} = \frac{0 + 302.6016 + 123.3375}{604}$$

$$= \frac{425.9391}{604} = 0.7052$$

$$Recall = \frac{(0.000 \times 40) + (0.9599 \times 399) + (0.4485 \times 165)}{604} = \frac{0 + 383.0001 + 74.0025}{604} = \frac{457.0026}{604}$$

$$= 0.7566$$

$$F1 - Score = \frac{(0.000 \times 40) + (0.7584 \times 399) + (0.7475 \times 165)}{604} = \frac{0 + 338.7126 + 92.4990}{604}$$

$$= \frac{431.2116}{604} = 0.7139$$

Support Vector Machine

a. Accuracy

$$Accuracy = \frac{TP^{(pos)} + TP^{(neu)} + TP^{(neg)}}{Total\ Data}$$

$$= \frac{5 + 383 + 98}{604} = \frac{486}{604} = 0,8046 = 80.46\%$$

b. Kelas Positive

True Positve (TP) = 98

False Positif (FP) = 22

False Negative (FN) = 67

Precision:

$$\frac{TP}{TP + FP} = \frac{98}{98 + 22} = \frac{98}{120} = 0.8167$$

Recall:

$$\frac{98}{98 + 67} = \frac{98}{165} = 0,5939$$

F1-Score:

$$2 \times \frac{0.8167 \times 0.5939}{0.8167 + 0.5939} = 0,6877$$

c. Kelas Netral

True Positive(TP) = 383

False Positive(FP) = 95

False Negative(FN) = 16

Precision:



$$\frac{TP}{TP + FP} = \frac{383}{383 + 95} = \frac{383}{478} = 0.8013$$

Recall:

$$\frac{383}{383 + 16} = \frac{383}{399} = 0.9599$$

F1-Score:

$$2 \times \frac{0.8013 \times 0.9599}{0.8013 + 0.9599} = 0,8735$$

d. Kelas Negative

True Positive (TP) =5

False Positive (FP) =1

False Negative (FN) = 35

Precision:

$$\frac{TP}{TP + FP} = \frac{5}{5 + 1} = \frac{5}{6} = 0.8333$$

Recall:

$$\frac{5}{5 + 35} = \frac{5}{40} = 0.125$$

F1-Score:

$$2 \times \frac{0.8333 \times 0.125}{0.8333 + 0.125} = 0,2174$$

e. Marco Average

$$Marco\ Avg = \frac{1}{n} = \sum_{i=1}^n (Precision_i, Recall_i, F1_i)$$

$$Marco\ Avg = \frac{Nilai\ Kelas\ Negative + Neutral + Positive}{3}$$

Dengan Nilai:

$$Precision = \frac{0.8333 + 0.8013 + 0.8013}{3} = \frac{2.4513}{3} = 0.8171$$

$$Recall = \frac{0.1250 + 0.9599 + 0.5939}{3} = \frac{1.68788}{3} = 0.5596$$

$$F1 - Score = \frac{0.2174 + 0.8735 + 0.6877}{3} = \frac{1.7786}{3} = 0.5929$$

f. Weighted Average

Weighted Average merupakan rata - rata pertimbangan berdasarkan jumlah support

$$Weight\ Avg = N = \sum_{i=1}^n Support_i$$

$$Weight\ Avg = \frac{Nilai\ Kelas\ Ne + Neu + Pos \times Support}{Total\ Data}$$

Dengan Nilai:

$$Precision = \frac{(0.8333 \times 40) + (0.8013 \times 399) + (0.8167 \times 165)}{604} = \frac{33.332 + 319.7187 + 134.755}{604}$$

$$= \frac{487.8062}{604} = 0.8076$$

$$Recall = \frac{(0.1250 \times 40) + (0.9599 \times 399) + (0.5939 \times 165)}{604} = \frac{5 + 383.0001 + 98.0}{604} = \frac{486.0001}{604} = 0.8046$$



$$F1 - Score = \frac{(0.2174 \times 40) + (0.8735 \times 399) + (0.6877 \times 165)}{604} = \frac{8.696 + 348.8265 + 113.4705}{604}$$

$$= \frac{470.993}{604} = 0.7798$$

4. KESIMPULAN

Berdasarkan hasil penelitian, metode *Naïve Bayes* dan *Support Vector Machine* (SVM) mampu mengklasifikasikan sentimen pengguna TikTok ke dalam kategori positif, netral, dan negatif. *Naïve Bayes* memperoleh akurasi sebesar 75,66%, sedangkan SVM mencapai 80,46%. SVM juga menunjukkan performa lebih baik pada sebagian besar metrik evaluasi, terutama pada kelas positif dan netral. Namun, kedua metode masih mengalami kesulitan dalam mengklasifikasikan sentimen negatif, sehingga diperlukan peningkatan kualitas dan keseimbangan data untuk memperoleh hasil klasifikasi yang lebih optimal.

DAFTAR PUSTAKA

- Galuh Putri Riyanto. (2025). *Pengguna TikTok di Indonesia Tembus 160 Juta, Terbanyak di Asia Tenggara*. <https://tekno.kompas.com/read/2025/07/30/14164347/pengguna-tiktok-di-indonesia-tembus-160-juta-terbanyak-di-asia-tenggara?page=all>
- Indra Budi Trisno, M. A. R. (2023). *WEBINAR ARTIFICIAL INTELLIGENCE DAN MACHINE LEARNING*. 2(11), 2307-2315. <https://doi.org/10.53625/jpm.v2i11.6873>
- Mirandini, N. A., Kuswari, N. I., Revinta, S. N., & Yulia, S. F. (2024). *Opini Publik Terhadap Kebijakan Penutupan Tik Tok Shop (Studi Literatur Dan Analisis Sentimen)*. 10(16), 556-569. <https://www.jurnal.peneliti.net/index.php/JIWP/article/view/7860/6633>
- Pratama, A. P., Purnamasari, A. I., & Ali, I. (2024). *ANALISIS SENTIMEN TANGGAPAN MASYARAKAT DI MEDIA SOSIAL TWITTER MENGENAI PENUTUPAN FITUR TIKTOK SHOP MENGGUNAKAN ALGORITMA KLASIFIKASI NAÏVE BAYES*. 8(1), 630-637. <https://www.ejournal.itn.ac.id/jati/article/view/8367/5031>
- Qamal, M., & Fuadi, W. (2021). *ANALISIS SENTIMEN TOKO ONLINE MENGGUNAKAN ALGORITMA NAIVE BAYES CLASSIFIER*. 2(1), 10. <https://ojs.unimal.ac.id/tts/article/view/6771>
- Septiani, A., Voutama, A., Hendriadi, A. A., & Heryana, N. (2024). *ANALISIS SENTIMEN MENGGUNAKAN ALGORITMA NAÏVE BAYES TERHADAP REGULASI TIKTOK SHOP PADA MEDIA SOSIAL X (TWITTER)*. 8(4), 5729-5735. <https://ejournal.itn.ac.id/jati/article/view/10040/5781>
- Sidupa, B. C., Dewi, C., Informasi, F. T., Kristen, U., & Wacana, S. (2025). *SENTIMEN ANALISIS TERHADAP APLIKASI TIKTOK MENGGUNAKAN SUPPORT VECTOR CLASSIFICATION*. 8(1), 1-7. <https://www.ejournal.itn.ac.id/index.php/mnemonic/article/view/12635>