

## PERBANDINGAN ALGORITMA *NAIVE BAYES* DAN *SUPPORT VECTOR MACHINE* PADA ANALISIS SENTIMEN CHATGPT

Evi Nurmala Pohan Hutagaol<sup>1)</sup>, Niken Agustina Pradana<sup>2)</sup>, Alan Pandu Prihartanto<sup>3)</sup>,  
Glorian Fernando Maker<sup>4)</sup>

<sup>1,2,3,4)</sup> Jurusan Teknik Informatika, Fakultas Teknik, Universitas Papua  
Jl. Gunung Salju, Amban, Manokwari Barat, Kabupaten Manokwari, Papua Barat  
E-mail : <sup>1)</sup> [evipohan06@gmail.com](mailto:evipohan06@gmail.com), <sup>2)</sup> [agustinaniken331@gmail.com](mailto:agustinaniken331@gmail.com), <sup>3)</sup>  
[panduprihartanto@gmail.com](mailto:panduprihartanto@gmail.com), <sup>4)</sup> [glorianfernando381@gmail.com](mailto:glorianfernando381@gmail.com)

### ABSTRAK

Perkembangan penggunaan ChatGPT di berbagai bidang menimbulkan beragam opini publik di media sosial X, sehingga analisis sentimen menjadi penting untuk memahami persepsi pengguna terhadap teknologi kecerdasan buatan berbasis chatbot. Penelitian ini dilakukan untuk menganalisis sentimen publik terhadap penggunaan ChatGPT di media sosial X serta membandingkan performa algoritma Naive Bayes dan *Support Vector Machine* (SVM) dalam klasifikasi sentimen berbasis teks. Dataset yang digunakan diperoleh dari platform Kaggle yang berisi sekitar 500.000 data tweet terkait ChatGPT. Dalam penelitian ini digunakan sebanyak 50.000 data tweet sebagai sampel penelitian. Tahapan *preprocessing* meliputi *cleaning*, *case folding*, *tokenizing*, *stopword removal*, dan normalisasi kata. Selanjutnya, metode TF-IDF digunakan untuk ekstraksi fitur teks. Dataset dibagi menjadi data *training* dan *testing* dengan rasio 80:20 menggunakan *stratified split* untuk menjaga distribusi kelas sentimen. Proses klasifikasi dilakukan menggunakan algoritma Multinomial Naive Bayes dan SVM dengan *linear kernel*. Evaluasi model dilakukan menggunakan *confusion matrix*, *accuracy*, *precision*, *recall*, dan *F1-score*. Hasil penelitian menunjukkan bahwa algoritma SVM memiliki performa lebih baik dengan akurasi sebesar 94,37%, sedangkan Naive Bayes memperoleh akurasi sebesar 77,21%. Performa SVM yang lebih tinggi dipengaruhi oleh kemampuannya dalam menangani data berdimensi tinggi hasil transformasi TF-IDF dan menghasilkan batas pemisah kelas yang lebih optimal. Penelitian ini diharapkan dapat menjadi referensi dalam pemilihan algoritma untuk analisis sentimen berbasis teks.

Kata kunci: analisis sentimen, Naive Bayes, SVM, ChatGPT, TF-IDF

### ABSTRACT

*The increasing use of ChatGPT in various fields has generated diverse public opinions on social media platform X, making sentiment analysis important to understand public perceptions toward AI-based chatbot technology. This study was conducted to analyze public sentiment toward the use of ChatGPT on social media platform X and to compare the performance of the Naive Bayes and Support Vector Machine (SVM) algorithms in text-based sentiment classification. The dataset used in this study was obtained from the Kaggle platform, which contains approximately 500,000 tweets related to ChatGPT. In this study, 50,000 tweets were used as the research sample. The preprocessing stages included cleaning, case folding, tokenizing, stopwords removal, and word normalization. Furthermore, the TF-IDF method was used for text feature extraction. The dataset was divided into training and testing data with an 80:20 ratio using stratified split to maintain the*

*distribution of sentiment classes. The classification process was carried out using the Multinomial Naive Bayes algorithm and SVM with a linear kernel. Model evaluation was performed using a confusion matrix, accuracy, precision, recall, and F1-score. The results showed that the SVM algorithm achieved better performance with an accuracy of 94.37%, while Naive Bayes obtained an accuracy of 77.21%. The higher performance of SVM was influenced by its ability to handle high-dimensional data generated from TF-IDF transformation and to produce a more optimal class separation boundary. This study is expected to serve as a reference in selecting algorithms for text-based sentiment analysis.*

*Keywords: sentiment analysis, Naive Bayes, SVM, ChatGPT, TF-IDF*

## 1. PENDAHULUAN

Kemajuan teknologi *Artificial Intelligence* (AI) telah mendorong munculnya berbagai sistem otomatis yang mampu membantu aktivitas digital manusia. Salah satu teknologi yang saat ini banyak digunakan adalah ChatGPT, yaitu chatbot berbasis AI yang dapat menghasilkan respons dalam bentuk percakapan teks secara otomatis. Kehadiran ChatGPT dimanfaatkan dalam berbagai bidang, seperti pencarian informasi, pembuatan konten, pemrograman, hingga aktivitas pembelajaran. Tingginya penggunaan teknologi ini memunculkan beragam opini dari masyarakat, baik berupa tanggapan positif maupun negatif. Analisis sentimen publik terhadap ChatGPT penting dilakukan untuk memahami tingkat penerimaan, kepercayaan, serta respons pengguna terhadap teknologi kecerdasan buatan berbasis chatbot. Hasil analisis sentimen juga dapat menjadi sumber informasi dalam mengevaluasi persepsi masyarakat terhadap penggunaan teknologi AI di kehidupan sehari-hari. Opini tersebut banyak ditemukan pada media sosial X dan dapat dimanfaatkan sebagai sumber data dalam proses analisis sentimen [1][2]. Analisis sentimen merupakan salah satu teknik text mining yang digunakan untuk mengidentifikasi kecenderungan opini ke dalam kategori positif, negatif, maupun netral. Dalam proses klasifikasi sentimen, algoritma *machine learning* memiliki peran penting dalam menentukan performa model. Beberapa algoritma yang sering digunakan dalam analisis sentimen adalah Naive Bayes dan *Support*

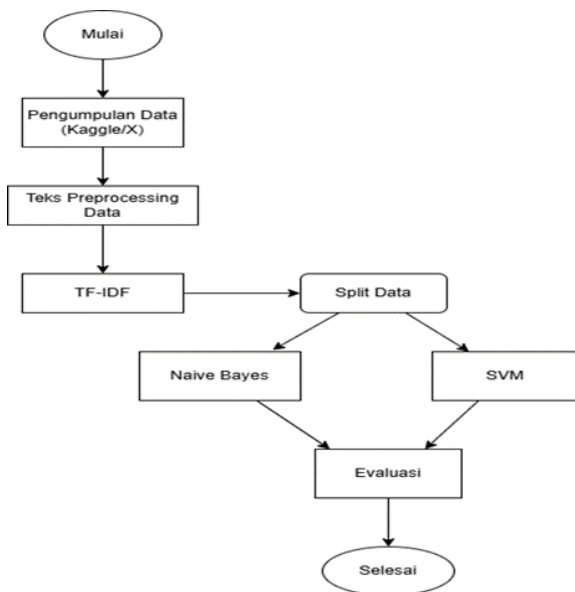
*Vector Machine* (SVM). Naive Bayes dikenal memiliki proses klasifikasi yang sederhana dan efisien pada data teks [3], sedangkan SVM mampu menangani data berdimensi tinggi dan sering menghasilkan tingkat akurasi yang baik pada proses klasifikasi [4]. Penelitian oleh Herlawati dkk. [11] juga menunjukkan bahwa algoritma Naive Bayes dan SVM efektif digunakan dalam klasifikasi sentimen berbasis data teks pada media digital.

Penelitian sebelumnya menunjukkan bahwa algoritma Naive Bayes dan *Support Vector Machine* (SVM) memiliki performa yang berbeda bergantung pada karakteristik dataset yang digunakan. Penelitian oleh Kurniawan dan Apriliani [5] menunjukkan bahwa analisis sentimen berbasis Twitter dapat dilakukan secara efektif menggunakan metode *machine learning*. Selain itu, penelitian Pratama dan Sari [6] membuktikan bahwa algoritma SVM mampu memberikan tingkat akurasi yang baik pada klasifikasi sentimen Twitter. Penelitian terbaru juga menunjukkan bahwa algoritma *Support Vector Machine* (SVM) cenderung menghasilkan performa klasifikasi yang lebih stabil dibandingkan Naive Bayes pada analisis sentimen berbasis media sosial. Penelitian oleh Ningsih dkk. [13] dan Safrudin dkk. [14] menunjukkan bahwa SVM mampu menghasilkan tingkat akurasi yang lebih tinggi pada klasifikasi teks dibandingkan Naive Bayes. Namun, penelitian yang secara khusus membandingkan performa Naive Bayes dan SVM pada analisis sentimen publik terhadap ChatGPT berbasis data media sosial X masih terbatas. Oleh karena itu,

penelitian ini dilakukan untuk membandingkan performa algoritma Naive Bayes dan *Support Vector Machine* dalam mengklasifikasikan sentimen publik terhadap ChatGPT menggunakan data dari media sosial X. Kebaruan penelitian ini terletak pada penggunaan dataset sentimen publik terkait ChatGPT dari media sosial X dengan jumlah data yang cukup besar, yaitu 50.000 tweet. Selain itu, penelitian ini menerapkan ekstraksi fitur menggunakan TF-IDF serta evaluasi model menggunakan *confusion matrix* dan *Stratified K-Fold Cross Validation* untuk membandingkan performa algoritma Naive Bayes dan *Support Vector Machine* secara lebih komprehensif.

## 2. METODE PENELITIAN

Penelitian ini menggunakan pendekatan kuantitatif dengan metode data mining untuk menganalisis dan membandingkan performa algoritma Naive Bayes dan *Support Vector Machine* (SVM) dalam analisis sentimen terhadap opini publik mengenai ChatGPT. Pendekatan ini digunakan karena penelitian berfokus pada pengolahan data teks menjadi data numerik serta evaluasi kinerja model menggunakan parameter statistik.



**Gambar 1.** Alur Penelitian

## 2.1 Sumber Data

Dataset yang digunakan dalam penelitian ini merupakan data sekunder yang diperoleh dari platform Kaggle dengan nama dataset *500k ChatGPT-related Tweets Jan-Mar 2023*. Dataset tersebut berisi sekitar 500.000 tweet terkait ChatGPT yang berasal dari media sosial X/Twitter periode Januari-Maret 2023 dan menggunakan bahasa Inggris. Dataset dapat diakses melalui tautan: <https://www.kaggle.com/datasets/khalidryder77/7/500k-chatgpt-tweets-jan-mar-2023>.

Dataset awal terdiri dari atribut tanggal, isi, username, jumlah like, dan retweet. Pada penelitian ini digunakan sebanyak 50.000 data teks sebagai sampel penelitian. Sebelum proses klasifikasi dilakukan, data terlebih dahulu melalui proses preprocessing dan pelabelan sentimen ke dalam kategori positif, netral, dan negatif menggunakan library TextBlob pada Google Colab.

**Tabel 1.** Deskripsi Dataset

No	Keterangan	Nilai
1	Sumber Data	Kaggle
2	Platform	Media Sosial X
3	Jumlah Data	50.000
4	Jenis Data	Teks
5	Label	Positif, Netral, Negatif

## 2.2 Teknik Pengumpulan Data

Data dikumpulkan dalam penelitian ini dengan mengunduh dataset dari platform Kaggle. Dataset diunduh pada saat penelitian berlangsung

dan digunakan sebagai bahan analisis tanpa melalui proses pengambilan data secara langsung (*crawling*). Selanjutnya, data dikumpulkan dan disimpan dalam bentuk file CSV untuk diproses pada langkah berikutnya.

### 2.3 Tahapan Penelitian

Penelitian ini membahas proses pengolahan data yang terdiri dari beberapa langkah utama sebagai berikut:

**Tabel 2.** Tahapan Preprocessing

No	Tahapan	Deskripsi
1	Cleaning	Menghapus URL, simbol, dll
2	Case Folding	Mengubah huruf menjadi kecil
3	Tokenizing	Memecah kalimat menjadi kata
4	Stopword Removal	Menghapus kata umum
5	Normalisasi	Mengubah kata tidak baku

#### 1. Data Collection

Tahap awal dalam penelitian ini adalah pengumpulan data (*data collection*), yaitu proses memperoleh data yang digunakan sebagai bahan analisis. Data yang digunakan merupakan data sekunder dari Kaggle yang berisi opini publik terhadap penggunaan ChatGPT yang berasal dari media sosial X.

#### 2. Data Preprocessing

Preprocessing data merupakan tahap penting untuk meningkatkan kualitas data sebelum

proses klasifikasi dilakukan. Pada tahap ini dilakukan beberapa proses pembersihan dan normalisasi data teks agar data lebih terstruktur dan mudah diproses oleh model *machine learning*.

Tahap pertama adalah *cleaning*, yaitu proses membersihkan teks dengan menghapus URL, *hashtag* (#), *mention* (@username), emoji, angka, tanda baca, simbol, dan karakter khusus lainnya yang tidak memiliki pengaruh signifikan terhadap analisis sentimen. Selain itu, dilakukan penghapusan data kosong, retweet, dan teks duplikat untuk mengurangi noise pada dataset. Proses ini dilakukan agar data yang sama tidak muncul pada data training dan data testing, sehingga hasil evaluasi model menjadi lebih valid dan tidak menghasilkan akurasi yang terlalu tinggi akibat kemiripan data.

Tahap berikutnya adalah *case folding* dengan mengubah seluruh huruf menjadi *lowercase* agar penulisan kata menjadi konsisten. Selanjutnya dilakukan *tokenizing* untuk memecah kalimat menjadi token kata berdasarkan spasi atau tanda pemisah tertentu.

Setelah tokenisasi, dilakukan *stopword removal* menggunakan daftar *stopword* bahasa Inggris untuk menghapus kata umum seperti "the", "is", "and", dan kata lain yang tidak memiliki sentimen yang kuat. Kemudian dilakukan normalisasi kata untuk mengubah kata tidak baku, singkatan, atau kata informal menjadi bentuk kata standar agar proses analisis teks menjadi lebih optimal.

Setelah proses *preprocessing* selesai dilakukan, data kemudian melalui tahap pelabelan sentimen secara otomatis menggunakan library TextBlob pada Google Colab. TextBlob digunakan untuk menghitung nilai *polarity* dari setiap teks tweet dengan rentang nilai -1 hingga +1. Nilai *polarity* positif dikategorikan sebagai sentimen positif, nilai *polarity* negatif dikategorikan sebagai sentimen negatif, sedangkan nilai *polarity* sama dengan nol dikategorikan sebagai sentimen netral. Hasil pelabelan tersebut selanjutnya digunakan sebagai target kelas dalam proses klasifikasi.

**Tabel 3.** Hasil Preprocessing

Index	Teks Asli	Final
0	have just published ChatGptNet, a library that allows to easily integrate #ChatGPT in	i have just published chatgptnet a library that allows to easily integrate
1	How to make money with Chat GPT? Follow these exact steps! THREAD https://t.co/DGpRt7OnyH	how to make money with chat gpt follow these exact steps thread
2	@elonmusk With Chat GPT coding for us, it is actually true!	with chat gpt coding for us it is actually true
3	Chat-GPT (#2) https://t.co/nLFWAtzqD5 #ChatGPT #ArtificialIntelligence #writing @Medium	chatgpt
4	#ChatGPT is saving me hours of work and a lot of moneyâ€¦! but it feels weird	is saving me hours of work and a lot of money but it feels weird

### 3. Data Splitting

Setelah tahap *preprocessing* dan pelabelan sentimen selesai dilakukan, dataset dibagi secara acak menjadi dua bagian, yaitu data *training* dan data *testing* dengan perbandingan 80:20. Data *training* digunakan untuk proses pelatihan model, sedangkan data *testing* digunakan untuk menguji kinerja model. Pembagian data dilakukan menggunakan *stratified split* untuk menjaga proporsi distribusi kelas sentimen pada data *training* dan data *testing* agar tetap seimbang.

**Tabel 4.** Split Data

Jenis Data	Persentase	Jumlah
Training	80%	40.000
Testing	20%	10.000

### 4. Data Transformation

Setelah proses pembagian data dilakukan, metode *Term Frequency–Inverse Document Frequency* (TF-IDF) digunakan untuk mengubah data teks yang telah diproses menjadi bentuk data numerik. Proses pembentukan TF-IDF dilakukan pada data *training* (*fit*), kemudian diterapkan pada data *testing* (*transform*) untuk menghindari terjadinya data *leakage* pada proses klasifikasi. Pendekatan ini dilakukan untuk mencegah informasi dari data *testing* mempengaruhi proses pelatihan model.

### 5. Modeling

Tahap *modeling* dilakukan menggunakan algoritma Naive Bayes dan *Support Vector Machine* (SVM) untuk mengklasifikasikan data sentimen ke dalam kategori positif, negatif, dan netral. Penggunaan kedua algoritma ini mengacu pada penelitian sebelumnya yang membandingkan beberapa metode klasifikasi pada analisis sentimen berbasis teks [7]. Proses pelatihan (*training*) dan pengujian (*testing*) dilakukan untuk

mengevaluasi kemampuan masing-masing algoritma dalam mengklasifikasikan data secara akurat, berdasarkan hasil dari tahap ini selanjutnya digunakan untuk membandingkan kinerja kedua algoritma dalam menentukan metode yang paling efektif dalam analisis sentimen terhadap ChatGPT.

## 6. Evaluation

Evaluasi model dilakukan menggunakan *confusion matrix* dengan parameter *accuracy*, *precision*, *recall*, dan *F1-score*. *Accuracy* digunakan untuk melihat tingkat ketepatan model secara keseluruhan, sedangkan *precision* dan *recall* digunakan untuk mengukur kemampuan model dalam mengklasifikasikan setiap kategori sentimen. Selain itu, *F1-score* digunakan untuk melihat keseimbangan antara *precision* dan *recall* pada masing-masing model klasifikasi. *Confusion matrix* digunakan untuk melihat jumlah prediksi benar dan prediksi salah pada setiap kelas sentimen sehingga performa model dapat dianalisis secara lebih detail. Hasil evaluasi digunakan untuk membandingkan performa algoritma Naive Bayes dan *Support Vector Machine* (SVM) dalam klasifikasi sentimen terhadap data teks terkait Chat GPT.

## 2.4 Metode yang Digunakan

### a. Naive Bayes

Naive Bayes merupakan metode klasifikasi berbasis probabilitas yang menggunakan Teorema Bayes dengan asumsi antar fitur bersifat independen. Dalam penelitian ini digunakan metode Multinomial Naive Bayes yang umum diterapkan pada klasifikasi teks karena mampu menangani data berbasis frekuensi kata seperti hasil transformasi TF-IDF. Algoritma ini memiliki proses komputasi yang cepat dan mampu menangani data dalam jumlah besar.

Penerapan algoritma dimulai dari pembagian dataset menjadi data *training* dan *testing*. Selanjutnya, data teks ditransformasikan menggunakan metode TF-IDF untuk mengubah teks menjadi data numerik. Setelah itu, data *training* digunakan untuk menghitung probabilitas kelas sentimen

positif, netral, dan negatif. Pada penelitian ini digunakan parameter *alpha* sebesar 0.5 untuk mengurangi probabilitas nol pada proses klasifikasi. Kelas dengan probabilitas tertinggi dipilih sebagai hasil prediksi sentimen. Selanjutnya, hasil klasifikasi dievaluasi menggunakan *confusion matrix*, *accuracy*, *precision*, *recall*, dan *F1-score*.

### b. Support Vector Machine (SVM)

*Support Vector Machine* (SVM) merupakan algoritma klasifikasi yang bekerja dengan mencari *hyperplane* terbaik untuk memisahkan data berdasarkan margin maksimum. Dalam penelitian ini digunakan kernel *linear* yang sesuai untuk data teks berdimensi tinggi hasil transformasi TF-IDF. Pada penelitian ini digunakan parameter  $C = 1.0$  dengan pengaturan *class weight default* tanpa penerapan *hyperparameter tuning* karena penelitian difokuskan pada perbandingan performa dasar kedua algoritma klasifikasi.

Penerapan algoritma SVM dimulai dari pembagian dataset menjadi data *training* dan *testing*. Selanjutnya, data teks ditransformasikan menggunakan TF-IDF untuk mengubah teks menjadi data numerik. Algoritma SVM dengan *linear kernel* kemudian membentuk *hyperplane* untuk memisahkan kelas sentimen positif, netral, dan negatif. Pada penelitian ini digunakan parameter  $C$  sebesar 1.0 untuk mengontrol toleransi kesalahan klasifikasi. Selanjutnya, hasil klasifikasi dievaluasi menggunakan *confusion matrix*, *accuracy*, *precision*, *recall*, dan *F1-score*.

## 3. HASIL DAN DISKUSI

Bagian ini menguraikan hasil eksperimen analisis sentimen terhadap ulasan ChatGPT menggunakan algoritma *Naive Bayes* dan *Support Vector Machine* (SVM).

### 3.1 Kinerja Model Klasifikasi

Pengujian dilakukan dengan membandingkan dua algoritma populer, yaitu

*Naive Bayes* dan *Support Vector Machine* (SVM) dalam mengklasifikasikan sentimen publik terhadap ChatGPT. Hasil evaluasi kinerja kedua model disajikan dalam bentuk classification report pada Tabel 5 dan Tabel 6.

**Tabel 5.** Classification Naive Bayes

	<b>Prec ision</b>	<b>Rec all</b>	<b>F1- scor e</b>	<b>suppor t</b>
Negative	94,9 8%	26,4 9%	41,4 3%	1359
Neutral	90,1 4%	72,1 4%	80,9 6%	3966
Positive	69,7 3%	96,5 1%	80,9 6%	4613
Accuracy	77,2 1%	77,2 1%	77,2 1%	9938
Macro avg	84,9 5%	65,0 5%	67,5 1%	9938
Weighted avg	81,3 3%	77,2 1%	75,2 3%	9938

Berdasarkan hasil evaluasi yang disajikan pada Tabel 5, algoritma Naive Bayes menghasilkan tingkat akurasi sebesar 77,21%. Model ini menunjukkan performa yang cukup baik dalam mengklasifikasikan sentimen positif dengan nilai recall yang tinggi. Namun, performa model pada kelas negatif masih relatif rendah dibandingkan kelas lainnya. Hal ini menunjukkan bahwa Naive Bayes cenderung mengalami bias terhadap kelas mayoritas sehingga kurang optimal dalam mengenali sentimen negatif.

Rendahnya performa Naive Bayes pada kelas negatif dipengaruhi oleh distribusi dataset yang tidak seimbang sehingga model cenderung mengikuti kelas mayoritas. Selain itu, asumsi independensi antar fitur pada Naive Bayes membuat model kurang optimal dalam menangkap hubungan antar kata pada teks hasil transformasi TF-IDF. Hasil ini sejalan dengan penelitian Iskandar dan Nataliani [7] yang menunjukkan bahwa performa Naive Bayes cenderung lebih rendah dibandingkan SVM pada klasifikasi sentimen berbasis teks.

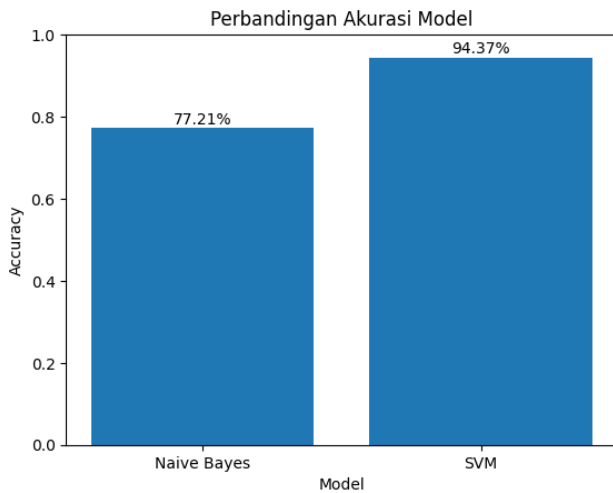
**Tabel 6.** Classification SVM

	<b>Precis ion</b>	<b>Recall</b>	<b>F1- score</b>	<b>suppo rt</b>
Negative	91,25 %	78,29%	84,28 %	1359
Neutral	93,86 %	98,21%	95,98 %	3966
Positive	95,61 %	95,79%	95,70 %	4613
Accurac y	94,37 %	94,37%	94,37 %	9938
Macro avg	93,51 %	90,77%	91,99 %	9938
Weighte d avg	94,31 %	94,37%	94,25 %	9938

Sementara itu, berdasarkan Tabel 6, algoritma Support Vector Machine (SVM) menunjukkan performa yang lebih baik dengan tingkat akurasi mencapai 94,37%. Nilai *precision*, *recall*, dan *F1-score* lebih seimbang dibandingkan Naive Bayes. Hasil penelitian ini sejalan dengan penelitian yang dilakukan oleh Imbiri dkk. [8] yang menunjukkan bahwa algoritma SVM memiliki kemampuan generalisasi yang baik pada klasifikasi sentimen berbasis Twitter. Selain itu, penelitian Emiliana dkk. [9] juga menunjukkan bahwa SVM mampu menghasilkan performa klasifikasi yang lebih stabil dibandingkan metode lainnya pada analisis sentimen media sosial. Penelitian terbaru oleh Ningsih dkk. [13] dan Hidayati dan Rochim [15] juga menunjukkan bahwa algoritma SVM memiliki performa klasifikasi yang lebih baik dibandingkan Naive Bayes pada analisis sentimen berbasis Twitter/X.

Hasil akurasi SVM pada penelitian ini menunjukkan performa yang lebih tinggi dibandingkan beberapa penelitian analisis sentimen sebelumnya pada data media sosial Twitter/X. Penelitian oleh Pratama dan Sari [6] menunjukkan bahwa algoritma SVM mampu menghasilkan performa klasifikasi yang baik pada analisis sentimen Twitter. Tingginya akurasi pada penelitian ini dipengaruhi oleh penggunaan *preprocessing*, transformasi fitur TF-IDF, serta

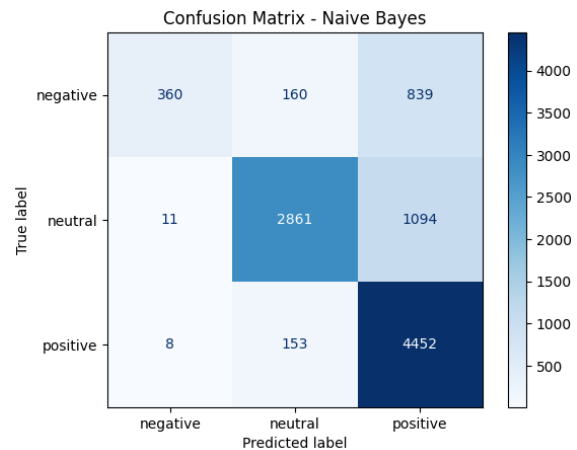
penggunaan *linear kernel* yang sesuai untuk data teks berdimensi tinggi.



**Gambar 2.** Perbandingan Akurasi Model Naïve Bayes dan SVM

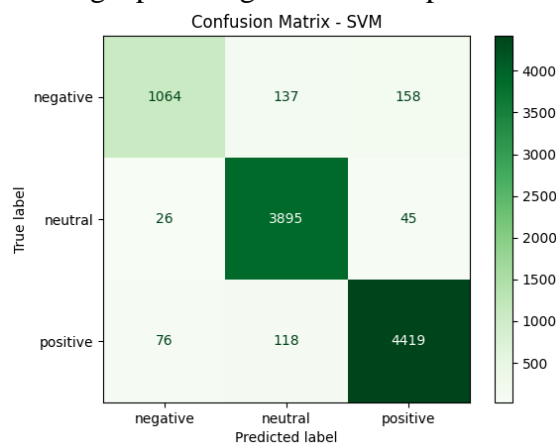
Perbandingan kinerja kedua algoritma ditunjukkan pada Gambar 2. Berdasarkan hasil pengujian, algoritma *Support Vector Machine* (SVM) memperoleh tingkat akurasi sebesar 94,37%, sedangkan Naive Bayes memperoleh akurasi sebesar 77,21%. Hasil tersebut menunjukkan bahwa SVM secara signifikan mengungguli Naive Bayes dalam proses klasifikasi sentimen terhadap ChatGPT. Hasil penelitian ini juga sejalan dengan penelitian Rahayu dkk. [12] yang menunjukkan bahwa algoritma SVM cenderung menghasilkan performa klasifikasi yang lebih baik dibandingkan Naive Bayes pada analisis sentimen berbasis teks.

Performa SVM yang lebih tinggi dipengaruhi oleh kemampuannya dalam menangani data teks berdimensi tinggi hasil transformasi TF-IDF serta kemampuannya dalam membentuk *hyperplane* yang optimal untuk memisahkan setiap kelas sentimen. Sementara itu, Naive Bayes cenderung lebih sensitif terhadap distribusi data yang tidak seimbang sehingga performanya pada kelas minoritas masih kurang optimal.



**Gambar 3.** Confusion Matrix - Naive Bayes

Berdasarkan *confusion matrix* pada algoritma Naive Bayes, model mampu mengklasifikasikan sentimen positif dengan cukup baik, terlihat dari jumlah prediksi benar pada kelas positif sebanyak 4.452 data. Namun, performa model pada kelas negatif masih relatif rendah, dimana hanya 360 data negatif yang berhasil diklasifikasikan dengan benar, sedangkan sebagian besar lainnya diprediksi sebagai sentimen positif. Hal ini menunjukkan bahwa Naive Bayes cenderung mengalami bias terhadap kelas mayoritas akibat distribusi data yang tidak seimbang. Selain itu, asumsi independensi antar fitur pada Naive Bayes menyebabkan model kurang optimal dalam menangkap hubungan antar kata pada data teks.



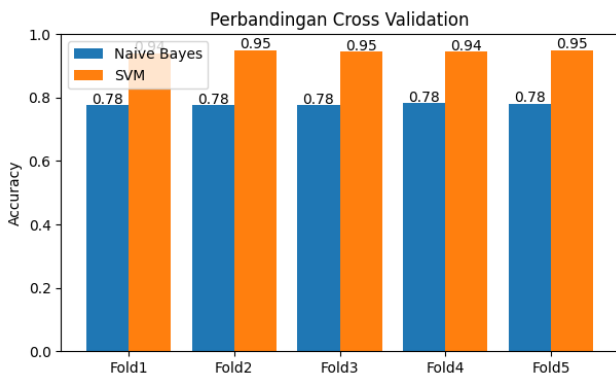
**Gambar 4.** Confusion Matrix SVM

Berbeda dengan Naive Bayes, *confusion matrix* pada algoritma *Support Vector Machine* (SVM) menunjukkan distribusi prediksi yang lebih seimbang pada setiap kelas sentimen.





memperoleh nilai akurasi yang lebih tinggi pada seluruh *fold* pengujian. Selain itu, perbedaan nilai akurasi antar *fold* pada SVM terlihat lebih stabil dibandingkan Naive Bayes. Hal ini menunjukkan bahwa SVM memiliki kemampuan generalisasi yang lebih baik dalam menangani data teks berbasis TF-IDF.



**Gambar 12.** Perbandingan Cross Validation

Visualisasi perbandingan *cross validation* pada Gambar 12 memperlihatkan perbedaan performa kedua algoritma secara lebih jelas. Algoritma SVM memperoleh nilai akurasi mendekati 95% pada seluruh *fold*, sedangkan Naive Bayes berada pada kisaran 78%. Hasil tersebut memperkuat bahwa SVM lebih efektif dan lebih stabil dalam melakukan klasifikasi sentimen dibandingkan Naive Bayes pada dataset yang digunakan dalam penelitian ini.

## 4. KESIMPULAN DAN SARAN

### 4.1 Kesimpulan

Hasil pengujian menunjukkan bahwa algoritma *Support Vector Machine* (SVM) memiliki performa yang lebih baik dibandingkan Naive Bayes dalam klasifikasi sentimen publik terhadap ChatGPT. Model SVM mampu mencapai tingkat akurasi sebesar 94,37%, sedangkan Naive Bayes memperoleh akurasi sebesar 77,21%. Selain itu, SVM menghasilkan nilai *precision*, *recall*, dan *F1-score* yang lebih seimbang pada setiap kategori sentimen sehingga mampu melakukan klasifikasi data dengan lebih stabil dan akurat.

Berdasarkan hasil *confusion matrix*, SVM juga menunjukkan tingkat kesalahan klasifikasi yang lebih rendah dibandingkan Naive Bayes pada setiap kelas sentimen. Sementara itu,

algoritma Naive Bayes cenderung mengalami bias terhadap kelas sentimen mayoritas akibat distribusi data yang tidak seimbang sehingga performanya pada sentimen negatif masih kurang optimal.

Hasil validasi menggunakan *Stratified K-Fold Cross Validation* menunjukkan bahwa SVM memiliki tingkat konsistensi dan kemampuan generalisasi yang lebih baik dibandingkan Naive Bayes dalam proses klasifikasi data teks. Dengan demikian, algoritma *Support Vector Machine* (SVM) dapat direkomendasikan sebagai metode yang lebih efektif untuk diterapkan pada analisis sentimen berbasis teks, khususnya terhadap opini publik mengenai ChatGPT di media sosial X.

### 4.2 Saran

Penelitian selanjutnya disarankan menggunakan jumlah dataset yang lebih besar dan lebih beragam agar hasil klasifikasi menjadi lebih representatif. Selain itu, penelitian berikutnya dapat membandingkan algoritma lain seperti *Random Forest*, *Logistic Regression*, maupun metode *deep learning* untuk memperoleh performa klasifikasi yang lebih optimal. Pengembangan pada tahap *preprocessing* dan penanganan data tidak seimbang juga dapat dilakukan untuk meningkatkan kualitas hasil analisis sentimen.

## 5. DAFTAR PUSTAKA

- [1] S. J. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 4th ed. Pearson, 2021. [Online]. Available: <http://aima.cs.berkeley.edu/>
- [2] B. Liu, "Sentiment Analysis and Opinion Mining," *Synth. Lect. Hum. Lang. Technol.*, vol. 5, no. 1, pp. 1–167, 2012, doi: 10.2200/S00416ED1V01Y201204HLT016.
- [3] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. Cambridge University Press, 2008. [Online]. Available:

<https://nlp.stanford.edu/IR-book/>

8, no. 1, pp. 45–53, 2023.

- [4] C. Cortes and V. Vapnik, “Support-vector networks,” *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995, doi: 10.1007/BF00994018.
- [5] R. Kurniawan and A. Apriliani, “Analisis Sentimen Masyarakat Terhadap Virus Corona Berdasarkan Opini Dari Twitter Berbasis Web Scraper,” *J. INSTEK (Informatika Sains dan Teknologi)*, vol. 5, no. 1, pp. 67–75, 2020, doi: 10.24252/instek.v5i1.13686.
- [6] M. A. Z. Pratama and W. K. Sari, “Analisis Sentimen Pengguna Twitter Terhadap Kebijakan Pemerintah Menggunakan Algoritme Support Vector Machine,” *J. Ilmu Komput. dan Sist. Inf.*, vol. 9, no. 2, pp. 145–152, 2021.
- [7] J. W. Iskandar and Y. Nataliani, “Perbandingan Naive Bayes, SVM, dan k-NN untuk Analisis Sentimen Gadget Berbasis Aspek,” *J. RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 5, no. 6, pp. 1120–1126, 2021, doi: 10.29207/resti.v5i6.3588.
- [8] M. A. Imbiri, L. Y. Baisa, and J. J. A. Limbong, “Sentiment Classification and Influential Actor Detection on Twitter (Case Study: The Raja Ampat Mining Conflict),” *Indonesian Journal of Data and Science*, vol. 7, no. 1, pp. 15–31, Mar. 2026.
- [9] M. R. Emiliana, M. R. I. Fata, and M. G. Adam, “Perbandingan Metode Naive Bayes dan Support Vector Machine untuk Analisis Sentimen pada Ulasan Pengguna Aplikasi X,” *J. Teknol. Inf. dan Ilmu Komput.*, vol. 10, no. 2, pp. 155–163, 2023.
- [10] W. Ningsih, B. Alfianda, Rahmaddeni, and D. Wulandari, “Perbandingan Algoritma SVM dan Naive Bayes dalam Analisis Sentimen Twitter pada Penggunaan Mobil Listrik di Indonesia,” *J. Tek. Inform.*, vol. 8, no. 1, pp. 45–53, 2023.
- [11] H. Herlawati, R. T. Handayanto, P. D. Atika, F. N. Khasanah, A. Y. P. Yusuf, and D. Y. Septia, “Analisis Sentimen Pada Situs Google Review dengan Naïve Bayes dan Support Vector Machine,” *J. Komtika (Komputasi dan Informatika)*, vol. 5, no. 2, pp. 153–163, 2021, doi: 10.31603/komtika.v5i2.6280.
- [12] A. S. Rahayu, A. Fauzi, and R. Rahmat, “Komparasi Algoritma Naïve Bayes dan Support Vector Machine (SVM) Pada Analisis Sentimen Spotify,” *J. Sistem Komputer dan Informatika*, vol. 4, no. 2, pp. 349–356, 2022, doi: 10.30865/json.v4i2.5398.
- [13] W. Ningsih, B. Alfianda, R. Rahmaddeni, and D. Wulandari, “Perbandingan Algoritma SVM dan Naïve Bayes dalam Analisis Sentimen Twitter pada Penggunaan Mobil Listrik di Indonesia,” *MALCOM Indonesian Journal of Machine Learning and Computer Science*, vol. 4, no. 2, pp. 556–562, 2024, doi: 10.57152/malcom.v4i2.1253.
- [14] M. Safrudin, M. Martanto, and U. Hayati, “Perbandingan Kinerja Naïve Bayes dan Support Vector Machine Untuk Klasifikasi Sentimen Ulasan Game Genshin Impact,” *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 8, no. 3, pp. 3182–3188, 2024, doi: 10.36040/jati.v8i3.8415.
- [15] N. Hidayati and A. F. Rochim, “Analisis Sentimen Pengguna Twitter/X terhadap Artificial Intelligence Menggunakan Support Vector Machine dan Naive Bayes,” *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 9, no. 1, pp. 45–53, 2025.