

# Analisis Sentimen Kinerja Jokowi Menggunakan Metode Naïve Bayes

<sup>1</sup>Mohammad Rizqilla, <sup>2</sup>Dwi Nurani, <sup>3</sup>Supriatin, <sup>4</sup>Alfie Nur Rahmi, <sup>5</sup>Mei Maemunah  
<sup>1,2,3,4,5</sup>Universitas Amikom Yogyakarta  
Yogyakarta

mohammadrizqilla12@students.amikom.ac.id<sup>1</sup>, dwinurani@amikom.ac.id<sup>2</sup>  
supriatin@amikom.ac.id<sup>3</sup>, alfienurrahmi@amikom.ac.id<sup>4</sup>, mei.m@amikom.ac.id

\*Penulis Korespondensi

Diajukan : 25/03/2025

Diterima : 16/04/2025

Dipublikasi : 18/04/2025

## ABSTRAK

Analisis sentimen terhadap opini publik mengenai kinerja Presiden Jokowi bertujuan untuk menerapkan algoritma Naïve Bayes dan hasilnya akan di optimasi dengan Grid Search. Pengambilan data pada sosial media YouTube. Ketidakakuratan analisis sentimen dapat memengaruhi evaluasi opini publik terhadap kebijakan pemerintah. Penelitian ini menggunakan algoritma Naive Bayes dengan optimasi parameter menggunakan Grid Search. Data melalui proses pra-pemrosesan sebelum diterapkan pada model. Hasil menunjukkan bahwa Naive Bayes menunjukkan performa lebih rendah sebelum diluakan grid search. Studi ini berkontribusi dalam analisis sentimen media sosial untuk memahami opini publik terhadap pemerintah.

**Kata Kunci:** Analisis sentimen, Presiden Jokowi, *Support Vector Machine*, kernel RBF, YouTube API v3

## I. PENDAHULUAN

Kinerja Presiden Joko Widodo selama sepuluh tahun terakhir menjadi perhatian utama masyarakat Indonesia. Media sosial menjadi platform utama untuk menyampaikan opini, kritik, dan saran, menggantikan media konvensional yang kurang aksesibel. Opini publik yang meluas ini menciptakan kebutuhan untuk menganalisis sentimen masyarakat secara terstruktur (Nehe et al., 2024).

Analisis sentimen berbasis media sosial efektif dalam memahami opini publik, dengan algoritma *Naive Bayes* sering digunakan karena akurasi yang tinggi (Anreaja et al., 2022). Optimasi algoritma seperti *Grid Search* terbukti dapat meningkatkan akurasi *Naive Bayes* secara signifikan, seperti pada analisis sentimen. Namun, penerapannya pada analisis opini terkait kinerja pemerintahan masih jarang dilakukan, membuka peluang penelitian di bidang ini (K. S. Chong & Shah, 2022).

Penelitian ini bertujuan untuk membandingkan kinerja *Naive Bayes* yang dioptimasi dengan *Grid Search* dalam analisis sentimen terhadap opini masyarakat tentang kinerja Presiden Jokowi. Dengan mengidentifikasi algoritma terbaik, penelitian ini diharapkan memberikan pandangan lebih akurat untuk evaluasi kebijakan yang lebih baik (Luthfanida, 2022).

## II. STUDI LITERATUR

### 1. Penelitian Terdahulu

Penelitian ini mengacu pada referensi yang ada dari tinjauan pustaka yang ada terkait dasar teori sebagai pedoman atau refrensi agar forndasi memahami konteks yang sesuai dengan penelitian

ini. Konfigurasi hyperparameter dalam algoritma analisis sentimen dibahas oleh Chong et al. Penelitian mereka, yang menggunakan Grid Search untuk optimasi parameter, mengungkapkan bahwa SVM mengungguli metode lain pasca-optimasi, dengan akurasi 85,65%, sedangkan Naive Bayes hanya 68,7% (K. S. Chong & Shah, 2022). Temuan ini memiliki relevansi khusus dengan tesis penulis, yang juga menggunakan Grid Search untuk meningkatkan kinerja kedua algoritma dalam analisis sentimen mengenai kinerja Presiden Jokowi.

Meskipun demikian, dalam menganalisis sentimen terkait kendaraan listrik dalam penelitian oleh Ni Wayan Ernawati, I Nyoman Satya Kumara, Widyadi Setiawan. Naive Bayes lebih cepat waktu dalam pelatihan, meskipun SVM sedikit akurat (Ernawati et al., 2023). Hal ini menjadi faktor bagi penulis karena kecepatan model tergantung pada dataset ukuran digunakan untuk analisis sentimen kinerja Presiden Jokowi.

Sebagai kesimpulan yang penulis amati, penelitian yang dilakukan Bilal Jannah et al. Telah menunjukkan bahwa SVM mengungguli Naive Bayes dalam analisis sentimen ulasan produk di pasar, dengan akurasi rata-rata 86,14% dibandingkan Naive Bayes yang hanya 81,13%. Penelitian ini menunjukkan bahwa SVM mampu mengakomodasi variasi data daripada Naive Bayes (Ernawati et al., 2023).

## 2. Kinerja Presiden

Kinerja Presiden merupakan salah satu indikator utama dalam keberhasilan pemerintahan dalam menjalankan mandat konstitusi. Aspek yang sering dievaluasi juga meliputi program-program strategis baik pembangunan infrastruktur, pemberdayaan masyarakat, dan reformasi kebijakan. Dalam era digital pada platform media sosial seperti Twitter (Sekarang X), yang menyediakan data tekstual untuk analisis sentimen (Cabui et al., 2022), (Luthfanida, 2022).

## 3. Analisis Sentimen

Analisis sentimen yaitu merupakan suatu metode dalam *Natural Language Processing* (NLP) digunakan untuk mengidentifikasi, mengevaluasi, dan mengklasifikasikan opini atau emosi dari suatu teks. Tujuannya adalah untuk mengetahui terkait opini itu bersifat positif, negatif (Nehe et al., 2024), (Anreaja et al., 2022), (Nalatissifa et al., 2021). Tahapan Analisis sentimen melibatkan beberapa tahapan, yaitu:

1. **Preprocessing:** Yaitu merupakan melakukan pembersihan data seperti penghapusan simbol, stopword, stemming, dan normalisasi teks.

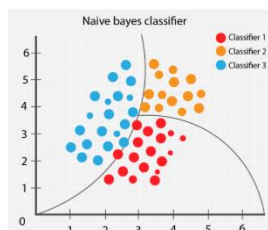
2. **Feature Extraction:** Mengubah teks menjadi fitur numerik dengan menggunakan metode seperti TF-IDF atau Bag of Words (BoW).

3. **Klasifikasi Sentimen:** Menggunakan algoritma machine learning seperti Naive Bayes dan SVM untuk klasifikasi.

4. **Evaluasi Model:** Mengukur performa model menggunakan metric seperti akurasi, presisi, recall, dan F1-Score.

## 4. Naive Bayes

Naive Bayes adalah algoritma klasifikasi dimana berbasis probabilistic classifier dimana yang mengasumsikan setiap fitur bahwa bersifat independen (naif). Keunggulan Naive Bayes efisiensi dalam menangani data besar serta kemampuan dalam menangani data besar serta kemampuan dalam menghasilkan hasil yang konsisten walaupun dengan parameter sederhana (Luthfanida, 2022), (Supian et al., 2024).



Gambar 1. Arsitektur Naive Bayes Pada Gambar 1. menunjukkan visualisasi Naive Bayes Classifier yang membagi ruang fitur menjadi tiga kelas (Classifier 1, 2, dan 3) dengan batas keputusan yang memisahkan setiap kelas berdasarkan probabilitas yang dihitung dari distribusi fitur, di mana setiap titik data dikelompokkan sesuai kelas dengan probabilitas tertinggi.

### 5. Grid Search

Grid Search merupakan teknik optimasi parameter yang digunakan untuk mencari kombinasi parameter terbaik dalam model machine learning. Prinsip dasar Grid Search adalah melakukan pencarian secara sistematis dengan semua kombinasi melalui parameter yang ditentukan (K. Chong & Shah, n.d.), (Iriananda et al., 2024).

Tahapan Grid Search:

1. Menentukan parameter yang akan dioptimalkan.
2. Menentukan rentang nilai parameter..
3. Melakukan pencarian semua kombinasi parameter.
4. Memilih parameter dengan hasil evaluasi terbaik.

### 6. Confusion Matrix

Confusion matrix adalah alat evaluasi yang digunakan untuk mengukur performa algoritma klasifikasi (Ardiyansah & Parjito, 2024), (Naufal et al., 2024), (Carneiro et al., 2018). Matrik ini memberikan gambaran tentang prediksi model dibandingkan dengan nilai aktualnya, dengan membagi hasil prediksi ke dalam empat kategori utama:

Tabel 1. Confusion Matrix

Aktual/Predict	Prediksi Positif	Prediksi Negatif
Aktual Positif	True Positive (TP)	False Negative (FN)
Aktual Negatif	False Positive (FP)	True Negative (TN)

Penjelasan dari confusion matrix diatas:

1. True Positive (TP): Jumlah data yang diprediksi sebagai positif oleh model, dan nilai aktualnya positif.
  2. False Positive (FP): Jumlah data yang diprediksi sebagai positif, tetapi nilai aktualnya negatif.
  3. True Negative (TN): Jumlah data yang diprediksi sebagai negatif, dan nilai aktualnya negative
  4. False Negative (FN): Jumlah data yang diprediksi sebagai negatif, tetapi nilai aktualnya positif.
- Confusion Matrix sangat penting untuk mengevaluasi perfoma model dalam analisis sentimen, baik terutama pada data yang tidak seimbang. Beberapa metrik kinerjanya dapat dihitung oleh confusion Matrix, yaitu:

1. Akurasi: Proporsi prediksi yang benar terhadap total data.

$$Akurasi = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

2. Precision: Proporsi prediksi positif yang benar dalam seluruh prediksi positif.

$$Recall = \frac{TP}{TP + FN} \quad (2)$$

3. Recall: Proporsi data positif yang benar-benar teridentifikasi sebagai positif.

$$ecall = \frac{TP}{TP + FN} \quad (3)$$

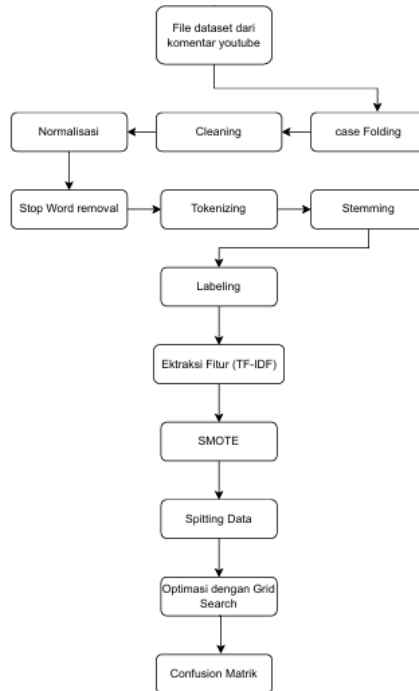
4. F1- Score: F1-Score merupakan rata-rata harmonik antara precision dan recall.

$$F1 = 2 \times \frac{Presisi \times Recall}{Presisi + Recall} \quad (4)$$

## III. METODE

Alur penelitian algoritma Naïve Bayes dalam optimasi menggunakan Grid Search pada analisis sentiment kinerja Presiden Jokowi , adalah sebagai berikut:

1. Scraping data adalah metode yang digunakan untuk mengambil informasi secara otomatis dari sumber-sumber yang tersedia di internet. Pada tahap ini dilakuukan menggunakan platform Google Colab dengan bahasa pemrograman Python. Teknik digunakan yaitu memperoleh data adalah web scraping, yang difokus penulis pada komentar video di media YouTube dengan proses YouTube Data API v3.
2. Preprocessing, digunakan untuk mengubah data teks yang tidak terstruktur akan menjadi bentuk lebih bersih serta dapat diolah oleh model analisis sentimen. Berikut tahapan Preprocessing:



Gambar 2. Alur penelitian

- a. Pada tahap proses ini akan mengkonversi teks ke dalam format huruf kecil (lowercase) dari huruf “A-Z” menjadi “a-z”. Hal ini bertujuan agar memberikan bentuk standar pada teks. Penerapan Case Folding dapat dilihat Tabel 2. berikut.

Tabel 2. Tahap Case Folding

Sebelum <i>Case Folding</i>	Sesudah <i>Case Folding</i>
Udh pak, istirahat santai dan nikmatin hidup sama ibu riana jauh lebih baik.	udh pak istirahat santai dan nikmatin hidup sama ibu riana jauh lebih baik.
Buat aq bapak hebat saat peduli sama rakyat kecil.	buat aq bapak hebat saat peduli sama rakyat kecil.

- b. Cleaning tahap membersihkan data komentar atau ulasan yang mengandung tautan, URL, simbol, dan karakter khusus yang tidak perlukan seperti tanda baca, emoticon, dan karakter khusus lainnya. Penerapan tahap Cleaning dapat dilihat pada Tabel 3. berikut.
- c. Slang Word tahap mengidentifikasi serta mengubah kata-kata informal atau singkatan menjadi bahasa baku bahasa Indonesia. Slang Word atau katau gaul digunakan sering pada ulasan online dan juga mempersulit proses analisis sentimen atau klasifikasi teks. Teknik ini digunakan melibatkan penggunaan kamus slang yang dipersiapkan sebelumnya, di mana nanti setiap kata slang diubah ke bentuk bakunya. Penerapan tahap Slang Word dapat dilihat pada Tabel 4. berikut.
- d. Stop Word Removal proses ini akan terjadi pengurangan jumlah kata yang harus dianalisis, dimana akan hanya kata-kata yang memiliki arti lebih spesifik serta penting untuk analisis sentimen. Dengan menghapus kata-kata umum yang kurang bermakna, seperti "di", "ke", "dari" (Indonesia) atau "the", "is", "in" (Inggris). Proses ini melibatkan tokenisasi, pencocokan dengan daftar stopwords, dan penghapusan kata tidak relevan, sehingga mengurangi dimensi data, mempercepat pemrosesan, dan membantu algoritma fokus pada kata-kata yang lebih informatif. Contohnya, kalimat "Saya pergi ke pasar untuk membeli buah"; setelah penghapusan stopwords menjadi "pergi pasar membeli

buah&quot;. Namun, daftar stopwords dapat disesuaikan sesuai kebutuhan karena tidak semua analisis memerlukan penghapusan ini. Penerapan tahap Slang Word dapat dilihat pada Tabel 5. berikut.

Tabel 5. Tahap Stop Word Removal

Sebelum <i>Stop Word Removal</i>	Sesudah <i>Stop Word Removal</i>
Udh pak, istirahat santai dan nikmatin hidup sama ibu riana jauh lebih baik.	istirahat santai nikmatin hidup
Buat aq bapak hebat saat peduli sama rakyat kecil.	hebat peduli rakyat

- e. Tokenizing, pada tahap ini dilakukan memecah teks menjadi unit-unit yang lebih kecil, yang disebut “token”. Token ini juga dapat berupa kata, frasa, kalimat, atau bahkan paragraf, tergantung pada tingkat tokenisasi yang diinginkan. Dalam bahasa alami (Natural Language Processing atau NLP), tokenisasi merupakan langkah awal penting untuk membantu menguraikan serta mengolah teks agar mudah untuk dianalisis. Penerapan tahap Tokenizing dapat dilihat pada Tabel 6 berikut.

Table 6. Tahap Tokenizing

Sebelum <i>Tokenizing</i>	Sesudah <i>Tokenizing</i>
Udh pak, istirahat santai dan nikmatin hidup sama ibu riana jauh lebih baik.	[['istirahat', 'santai', 'nikmatin', 'hidup']
Buat aq bapak hebat saat peduli sama rakyat kecil.	[['hebat', 'peduli', 'rakyat']

- f. Pada tahap proses ini dilakukan pengolahan bahasa alami dimana bertujuan mengurangi kata-kata ke bentuk dasarnya atau akar kata. Pada proses ini menghilangkan seperti imbuhan, prefiks, sufiks, atau infiks yang terdapat pada kata dapat diolah menjadi lebih lanjut. Dengan stemming juga kata-kata yang ada makna serupa dikelompokkan bersama sehingga memudahkan untuk analisis serta pemrosesan data. Penerapan tahap Tokenizing dapat dilihat pada Tabel 7. berikut.

3. **Labeling**, pada tahap ini, telah dilakukan yaitu pelabelan dimana menggunakan metode VADER yang sudah dimodifikasi dengan lexicon bahasa Indonesia dari SentiStrength\_ID. Modifikasi ini bertujuan baik meningkatkan akurasi untuk analisis sentimen pada teks bahasa Indonesia, terkhusus pada komentar terkait kinerja Presiden Jokowi dari platform YouTube. Analisis sentimen dapat dikatakan lebih relevan dengan konteks bahasa lokal untuk digunakan dalam data. Langkah utama untuk dimulai yaitu memodifikasi lexicon VADER yang berbahasa Inggris menjadi lexicon SentiStrength\_ID untuk mengakomodasi Bahasa Indonesia, lalu diikuti dengan analisis polaritas kalimat menggunakan tiga metric (negatif, positif, dan compound), dimana dengan compound skor ini akan menjadi acuan untuk klasifikasi sentimen ataupun nilai compound lebih dari 0 akan dikategorikan sebagai sentimen positif, dan nilai kurang dari 0 akan disentimenkan negatif. Dari Hasil klasifikasi ini akan disimpan berupa file untuk analisis lebih lanjut.

#### 4. Ekstrasi Fitur (TF-IDF)

Pada tahap ini dilakukan TF-IDF (Term Frequency-Inverse Document Frequency) adalah metode untuk memberikan bobot pada kata dalam dokumen berdasarkan pentingnya kata tersebut. Pertama, dihitung Term Frequency (TF), yaitu seberapa sering kata muncul dalam dokumen dibandingkan total kata. Kemudian, dihitung Inverse Document Frequency (IDF) untuk memberi bobot rendah pada kata yang sering muncul di banyak dokumen. Kedua nilai ini dikalikan untuk menghasilkan TF-IDF, yang memberikan bobot lebih tinggi pada kata yang sering muncul dalam dokumen tertentu tapi jarang ditemukan di dokumen lain. Hasilnya, dokumen dapat direpresentasikan dalam bentuk vector numerik yang siap digunakan dalam analisis seperti klasifikasi teks atau analisis sentimen.

#### 5. SMOTE

Pada tahap ini akan dilakukan proses menyeimbangkan jumlah data kelas minoritas dengan RandomOverSampler untuk mengatasi ketidakseimbangan kelas dalam dataset, yang akan mengakibatkan performa buruk baik untuk model pembelajaran mesin. Lalu dilakukan oversampling dimana akan menghasilkan data yang seimbang, dan data yang sudah di-resample dimasukan DataFrame baru. Setiap jumlah kelas setelah balancing akan ditampilkan serta proporsi kelas akan digambarkan dalam pie chart. Dengan teknik ini. Model dibangun akan memiliki representasi yang lebih baik dari setiap kelas, dan akan meningkatkan akurasi serta kinerjanya untuk klasifikasikan sentimen.

**6. Spliting Data**

Pada tahap proses ini akan membagi dataset menjadi data latih dan data uji menggunakan fungsi train\_test\_split, yang penting untuk menghindari overfitting. Sebanyak 80% data digunakan untuk melatih model (data latih), sementara 20% sisanya digunakan untuk menguji model (data uji) agar dapat menilai akurasi dan performa model pada data yang tidak dikenalnya. Penggunaan random\_state=42 memastikan pembagian data yang konsisten setiap kali kode dijalankan. Pembagian membantu model belajar dari data yang cukup dan diuji pada data yang baru untuk mengevaluasi kemampuannya dalam memprediksi data nyata.

**7. Optimasi Grid Search**

Pada tahap proses ini, dilakukan optimasi hyperparameter menggunakan GridSearch untuk mencari kombinasi parameter terbaik pada model Naive Bayes. Proses ini bertujuan untuk meningkatkan kinerja model dengan menguji berbagai nilai parameter yang mungkin, seperti parameter var\_smoothing pada Naive Bayes. GridSearch melakukan pencarian secara menyeluruh pada ruang parameter yang telah ditentukan, menggunakan teknik cross-validation untuk mengevaluasi kinerja setiap kombinasi parameter.

Dengan cara ini, GridSearch dapat menemukan kombinasi parameter yang memberikan akurasi terbaik dengan menghindari pengaturan parameter yang kurang optimal. Hasil dari optimasi ini menunjukkan parameter terbaik untuk model, yang kemudian digunakan untuk melatih model dengan kinerja yang lebih optimal. Proses ini memastikan bahwa model memiliki konfigurasi yang tepat untuk memprediksi data dengan lebih akurat dan efektif.

**8. Confusion Matrix**

Pada tahap proses ini akan melakukan yaitu melatih model Naive Bayes untuk analisis publik terkait kinerja Presiden Jokowi, dan selanjutnya dievaluasi dengan confusion matrix untuk mengevaluasi kinerja model baik prediksi kelas, serta model akan cenderung menghasilkan false positive atau false negative. Dengan menghitung nilai yang ada yaitu akurasi, presisi, recall, dan untuk memahami model secara menyeluruh.

**IV. HASIL DAN PEMBAHASAN**

**1. Scraping Data**

Dalam pengumpulan data untuk penelitian terkait kinerja Presiden Jokowi dengan menggunakan platform Google Colab dengan bahasa pemrograman Python. Teknik yang digunakan untuk memperoleh data dengan web scraping. Dengan memanfaatkan YouTube Data API v3 dari komentar video terkait kinerja Presiden Jokowi didapatkan dengan jumlah data 13.222 yang berasal dari video YouTube terkait kinerja Presiden Jokowi.

	publishedAt	authorDisplayName	textDisplay	likeCount
0	2024-10-23T04:30:03Z	@AhmadBukhari-l6z	MO YET BERHENTI MEMANJAT POHON!!!!	2
1	2024-10-10T05:01:57Z	@vellmanmarpaung94	Belum tentu bos sebagai rakyat biasa... bayak ka...	0
2	2024-10-02T12:06:41Z	@Endangiwon-uz4ie	Tdk di terima di solo diusir sama rakyat	1
3	2024-09-19T16:02:08Z	@TrimMargapura	Udah udah, maunya bikin rusuh terus, sebenerny...	0
4	2024-09-11T08:31:49Z	@Motogpboy-n2o	Ngiboel lagie?!	0
...	...	...	...	...
13217	2017-02-01T15:00:42Z	@wantridarwanimanik8985	Meilanda Da ia pak jokowi harus 2 periode	9
13218	2017-02-04T12:31:33Z	@jojokamill1936	bukan dua priode kalo bisa seumur hidup	8
13219	2017-02-04T17:23:28Z	@dianadiana3328	Jojo Kamill ..amiin	3
13220	2017-02-05T00:02:54Z	@alenwibowo7641	Jojo Kamill monga aja alloh menyertai amiin	15
13221	2017-02-05T16:55:59Z	@jhontruly122	semoga sehat selalu buat Pak Jokowi	17

13222 rows x 4 columns



### Gambar 3. Data Komentar YouTube

Pada Gambar 3. penelitian telah melakukan pemilihan variabel pada data komentar yang relevan dimana *textDisplay* berisi komentar publik menjadi acuan variabel untuk memberikan label komentar publik terkait kinerja Presiden Jokowi.

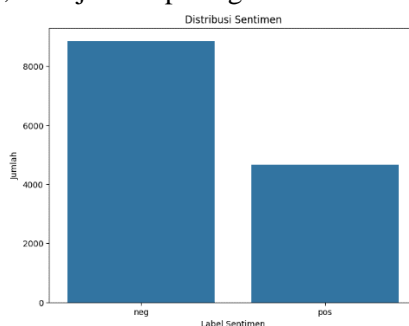
## 2. Preprocessing

Data hasil scraping akan selanjutnya masuk ke tahap preprocessing data dalam analisis sentimen yang dimulai dengan Case Folding, yaitu dengan mengubah semua huruf dalam teks menjadi huruf kecil untuk menghilangkan perbedaan akibat penggunaan huruf besar jadi huruf kecil. Setelah itu dilakukan Cleaning, yakni membersihkan teks dari elemen yang tidak relevan seperti tanda baca, angka, dan simbol khusus. Selanjutnya adalah Slang Word Normalization, yaitu mengubah kata-kata tidak baku atau slang menjadi bentuk standar agar kata-kata dengan arti sama dapat dikenali dengan bahasa Indonesia. Selanjutnya dilakukan Stop Word Removal, yakni menghapus kata-kata umum yang tidak memiliki makna signifikan, seperti “dan”, “di”, atau “yang”. Proses berikutnya adalah Tokenizing, yaitu memisahkan teks menjadi kata-kata atau token individu, dan terakhir dilakukan Stemming, yakni mengubah kata-kata menjadi bentuk dasarnya untuk menyederhanakan variasi kata dengan makna yang sama.

## 3. Labeling Data

Pelabelan data dalam penelitian ini dilakukan menggunakan Sentiment Intensity Analyzer (SIA) yang sudah dimodifikasi dengan kata-kata berbahasa Indonesia, yang disebut *sentiwords\_id*, agar lebih akurat dalam menganalisis sentimen dalam bahasa Indonesia. Pada tahap ini alat menghitung skor sentimen setiap komentar menggunakan fungsi *polarity\_scores()*, yang menghasilkan empat jenis skor: *neg* untuk sentimen negatif, *neu* untuk sentimen netral, *pos* untuk sentimen positif, dan *compound*, yaitu skor gabungan yang berkisar antara -1 hingga +1. Skor *compound* digunakan untuk menentukan sentimen komentar: jika nilainya lebih dari 0, komentar dikategorikan sebagai positif, sedangkan jika nilainya 0 atau kurang, komentar dikategorikan sebagai negatif. Hasil analisis menunjukkan bahwa dari seluruh komentar, terdapat 8.859 komentar dengan sentiment negatif dan 4.660 komentar dengan sentiment positif, yang kemudian divisualisasikan dalam grafik batang untuk memperlihatkan perbandingan jumlahnya.

Berikut hasil pelabelan data dengan *SentiStrength* dengan visualisasinya untuk memberikan gambar distribusi kelas sentimen, ditunjukkan pada gambar 4.



Gambar 4. Hasil Distribusi Sentimen

Pada Gambar 4. menunjukkan dimana stribusi sentimen dalam data yang dianalisis yaitu sentimen negatif (*neg*) memiliki jumlah sekitar 8000, sedangkan sentimen positif (*pos*) sekitar 4000. Hal ini menunjukkan bahwa data didominasi oleh sentimen negatif. Berikut kata-kata yang ada pada *wordcloud* setelah dilakukan labeling pada gambar 5.



Gambar 5. Hasil wordcloud setelah labeling

Pada Gambar 5. menunjukkan sebuah analisis sentimen dari data teks dalam bahasa Indonesia menggunakan *library nltk*, *Sentiment Intensity Analyzer*. Analisis ini ditampilkan dalam bentuk *wordcloud*, yaitu representasi visual dari frekuensi kata-kata yang muncul di data teks. Kata-kata yang paling sering muncul di wordcloud ini adalah "Jokowi" dan "orang", yang menunjukkan bahwa data teks ini berfokus pada tokoh Presiden Joko Widodo dan orang-orang di sekitarnya

#### 4. Ekstraksi Fitur (TF-IDF)

Ekstraksi fitur menggunakan TF-IDF dimulai dengan mendefinisikan corpus yang terdiri dari empat kalimat terkait dokumentasi dan analisis data. Menggunakan *TfidfVectorizer* dari *sklearn*, nilai TF-IDF dihitung untuk setiap kata, mencerminkan relevansinya dalam dokumen dan keseluruhan corpus. Hasilnya berupa matriks TF-IDF, yang dikonversi menjadi *DataFrame* dengan *pandas* untuk memudahkan analisis. Nilai TF-IDF yang lebih tinggi menunjukkan kata lebih penting dalam konteks dokumen tertentu, membantu dalam klasifikasi teks, pengelompokan dokumen, dan pemahaman tema utama dalam corpus. Hasil dari perhitungan ini adalah matriks TF-IDF yang diubah menjadi *DataFrame* dengan menggunakan pustaka *pandas*, sehingga memudahkan untuk analisis dan interpretasi. Setiap elemen dalam *DataFrame* menunjukkan skor TF-IDF untuk kata tertentu dalam dokumen tertentu, di mana nilai yang lebih tinggi menunjukkan relevansi yang lebih besar. Misalnya, kata "dokumentasi" memiliki nilai TF-IDF 0.371565 dalam dokumen pertama, menandakan pentingnya dalam konteks kalimat tersebut, sedangkan kata "algoritma" memiliki nilai 0.000000, menunjukkan bahwa kata tersebut tidak muncul dalam dokumen tersebut. Hasil ini akan memberikan wawasan yang berharga tentang kata-kata kunci yang dapat digunakan untuk analisis lebih lanjut, seperti klasifikasi teks atau pengelompokan dokumen, serta membantu dalam memahami lebih terkait tema utama yang terkandung dalam *corpus* yang dianalisis.

Berikut hasil Ekstraksi fitur atau TF-IDF pada gambar 6.

	adalah	algoritma	analisis	berbasis	dalam	data	dicari
0	0.292946	0.000000	0.000000	0.000000	0.292946	0.000000	0.000000
1	0.214687	0.000000	0.000000	0.000000	0.214687	0.000000	0.000000
2	0.000000	0.000000	0.348299	0.000000	0.000000	0.348299	0.000000
3	0.000000	0.300043	0.000000	0.300043	0.000000	0.000000	0.300043
	dokumen	dokumentasi	harus	...	perangkat	pola	relevan
0	0.000000	0.371565	0.000000	...	0.371565	0.000000	0.000000
1	0.214687	0.000000	0.000000	...	0.000000	0.000000	0.000000
2	0.000000	0.000000	0.000000	...	0.000000	0.348299	0.000000
3	0.473114	0.000000	0.300043	...	0.000000	0.000000	0.300043
	sebuah	sistem	statistik	teks	tf	untuk	yang
0	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
1	0.544606	0.000000	0.272303	0.000000	0.272303	0.214687	0.000000
2	0.000000	0.348299	0.000000	0.274603	0.000000	0.274603	0.000000
3	0.000000	0.000000	0.000000	0.236557	0.000000	0.000000	0.300043

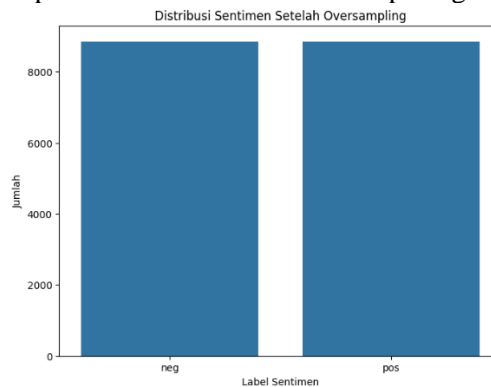
[4 rows x 34 columns]

Gambar 6. Hasil Ekstraksi fitur (TF-IDF)

Pada Gambar 6. adalah tabel yang menunjukkan seberapa penting setiap kata dalam kumpulan teks yang dianalisis. Nilai-nilai dalam tabel menunjukkan bobot sebuah kata, dihitung berdasarkan seberapa sering kata itu muncul dalam satu dokumen dibandingkan dengan semua dokumen lain. Semakin tinggi nilainya, semakin penting kata tersebut dalam dokumen tersebut.

## 5. SMOTE

Proses SMOTE (Synthetic Minority Over-sampling Technique) digunakan untuk menangani ketidakseimbangan kelas dalam dataset dengan cara menghasilkan data sintesis pada kelas minoritas. Dalam dataset ini, sebelum dilakukan SMOTE, jumlah data pada kelas negatif (neg) dan positif (pos) tidak seimbang, yang dapat menyebabkan model machine learning bias terhadap kelas mayoritas. SMOTE bekerja dengan membuat sampel baru berdasarkan interpolasi antara data minoritas dan tetangga terdekatnya, sehingga menambah jumlah sampel pada kelas minoritas tanpa menduplikasi data. Setelah proses ini, jumlah data pada kedua kelas menjadi seimbang, yaitu masing-masing 8.859 sampel, seperti terlihat pada grafik distribusi setelah oversampling. Penyeimbangan ini memungkinkan model untuk mempelajari pola dari kedua kelas secara adil, meningkatkan kemampuan model dalam mengenali kelas minoritas, dan menghasilkan prediksi yang lebih akurat serta andal. Proses ini merupakan langkah penting dalam preprocessing untuk analisis sentimen yang lebih optimal. Berikut hasil SMOTE pada gambar 7.



Gambar 7. Hasil SMOTE

Pada Gambar 7. adalah merupakan distribusi sentimen setelah penerapan teknik *oversampling* menggunakan SMOTE, di mana jumlah data pada kedua kelas, yaitu negatif (neg) dan positif (pos), telah menjadi seimbang dengan masing-masing sebanyak 8.859 data."

## 6. Splitting Data

Splitting data adalah proses membagi dataset menjadi dua bagian utama, yaitu data latih (training data) dan data uji (testing data), dengan tujuan agar model machine learning dapat dilatih dan diuji secara terpisah.

```
Ukuran data latih (X_train): 14174
Ukuran data uji (X_test): 3544
Ukuran label latih (y_train): 14174
Ukuran label uji (y_test): 3544
```

Gambar 8. Hasil *Splitting Data*

Hasil Splitting Data menunjukkan bahwa dataset telah berhasil dibagi menjadi dua bagian, yaitu data latih (X\_train dan y\_train) sebanyak 14.174 sampel, yang digunakan untuk melatih model, serta data uji (X\_test dan y\_test) sebanyak 3.544 sampel, yang digunakan untuk menguji performa model dalam melakukan prediksi pada data baru.

## 7. Naïve Bayes

Dalam penelitian ini, algoritma Naive Bayes juga digunakan untuk melakukan analisis sentimen publik sebagai salah satu pembandingan performa model. Naïve Bayes merupakan algoritma yang berbasis probabilitas dan asumsi independensi antar fitur, sehingga sering digunakan dalam klasifikasi teks. Evaluasi model dilakukan dengan menghitung metrik precision, recall, f1-score,

dan akurasi keseluruhan menggunakan data uji. Berikut hasil algoritma Naive Bayes pada gambar 9

	precision	recall	f1-score	support
neg	0.91	0.52	0.67	1830
pos	0.65	0.95	0.77	1714
accuracy			0.73	3544
macro avg	0.78	0.74	0.72	3544
weighted avg	0.79	0.73	0.72	3544

Akurasi model Naive Bayes: 0.7291196388261851

Gambar 9. Hasil algoritma *Naive Bayes*

Pada Gambar 9. menunjukkan bahwa model Naive Bayes memiliki akurasi sebesar 72.91%. Untuk kelas negatif, precision tercatat sebesar 91%, tetapi recall hanya 52%, yang menunjukkan bahwa model kurang optimal dalam mengidentifikasi semua data negatif. Sebaliknya, untuk kelas positif, precision mencapai 65%, sementara recall lebih tinggi, yaitu 95%.

### 8. Optimasi Grid Search

Penelitian melakukan optimasi Grid Search untuk penerapan model Naive Bayes sebelum dan setelah dilakukan Grid Search dengan menunjukkan peningkatan performa yang signifikan. Sebelum Grid Search, model memiliki akurasi keseluruhan sebesar 72,91%, dengan kelemahan pada kemampuan mengenali data negatif, ditunjukkan oleh recall kelas negatif yang rendah (52%). Setelah dilakukan Grid Search dengan parameter terbaik ( $\alpha = 0.1$ ), akurasi meningkat menjadi 75,22%, disertai peningkatan recall kelas negatif menjadi 86%, yang berarti model lebih baik dalam mengenali data negatif tanpa banyak mengorbankan performa pada kelas positif. Berikut hasil dari optimasi Grid Search pada gambar 10.

	precision	recall	f1-score	support
neg	0.91	0.52	0.67	1830
pos	0.65	0.95	0.77	1714
accuracy			0.73	3544
macro avg	0.78	0.74	0.72	3544
weighted avg	0.79	0.73	0.72	3544

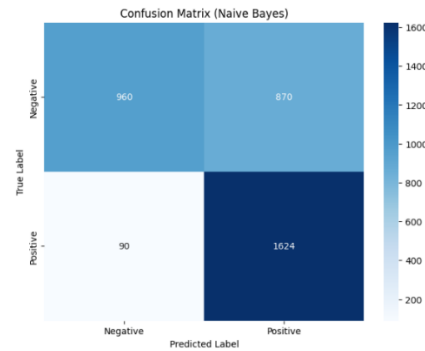
Akurasi model Naive Bayes: 0.7291196388261851

Gambar 10. Hasil setelah dilakukan optimasi *Grid Search*

Pada Gambar 10. menunjukkan Setelah dilakukan optimasi dengan Grid Search, model Naive Bayes mengalami peningkatan akurasi dari 72,91% menjadi 75,22%, dengan recall kelas negatif meningkat signifikan dari 52% menjadi 86%, serta performa yang lebih seimbang antara precision dan recall pada kedua kelas, menunjukkan model yang lebih andal dan optimal dalam klasifikasi data.

### 9. Confusion Matrix

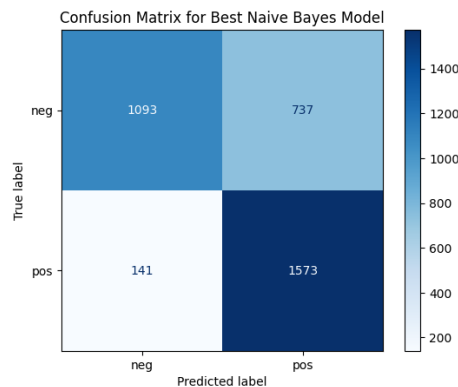
Setelah penerapan algoritma Naive Bayes dan SVM, lalu melakukan optimasi dengan metode Grid Search. Berikut dapat dilihat Visualisasi algoritma Naive Bayes.



Gambar 11. Hasil *Confusion Matrix* algoritma *Naive Bayes*

Pada Gambar 11. *Confusion Matrix* menunjukkan hasil prediksi di mana model *Naive Bayes*. Matriks ini terdiri dari empat komponen utama yaitu *True Negative* (960), yaitu jumlah data negatif yang diprediksi dengan benar; *False Positive* (870), yaitu jumlah data negatif yang salah diprediksi sebagai positif; *True Positive* (1624), yaitu jumlah data positif yang diprediksi dengan benar; dan *False Negative* (90), yaitu jumlah data positif yang salah diprediksi sebagai negatif.

Hasil ini menunjukkan bahwa model memiliki kelemahan pada prediksi data negatif, terlihat dari tingginya nilai *False Positive* (870), yang berarti model cenderung salah memprediksi data negatif sebagai positif. Selain itu, meskipun jumlah *True Positive* (1624) relatif tinggi, nilai *False Negative* (90) menunjukkan bahwa ada sebagian kecil data positif yang salah diklasifikasikan sebagai negatif. Dengan demikian, performa model ini masih dapat ditingkatkan, terutama dalam menyeimbangkan prediksi antara kelas negatif dan positif, yang akan dilakukan melalui proses optimasi parameter menggunakan *Grid Search*. Berikut adalah visualisasi setelah dilakukan optimasi *Grid Search* pada gambar 12.



Gambar 12. Hasil *Confusion Matrix* Optimasi *Grid Search* pada *Naive Bayes*

Pada Gambar 12. menunjukkan *Confusion Matrix* dari hasil optimasi algoritma *Naive Bayes* menggunakan *Grid Search*, di mana performa model meningkat dengan jumlah *True Negative* (1093) dan *True Positive* (1573) yang lebih tinggi dibandingkan sebelumnya, serta penurunan jumlah kesalahan pada *False Positive* (737) dan *False Negative* (141), sehingga model menjadi lebih akurat dan seimbang dalam mengklasifikasikan data negatif dan positif.

## V. KESIMPULAN

Algoritma *Naive Bayes* sebelum optimasi memiliki akurasi sebesar 72,91%, dengan *recall* kelas negatif yang rendah (52%) namun *recall* kelas positif yang tinggi (95%). Setelah optimasi menggunakan *Grid Search*, akurasi *Naive Bayes* meningkat menjadi 75,22%, dengan *recall* kelas negatif meningkat signifikan menjadi 86%.

## VI. REFERENSI

- Anreaja, L. J., Harefa, N. N., Galih, J., Negara, P., Nathan, V., Priyantara, H., & Prasetyo, A. B. (2022). *JISA (Jurnal Informatika dan Sains) Naive Bayes and Support Vector Machine Algorithm for Sentiment Analysis Opensea Mobile Application Users in Indonesia*.
- Ardiyansah, & Parjito. (2024). KLIK: Kajian Ilmiah Informatika dan Komputer Perbandingan Metode Naive Bayes dan Support Vector Machine Dalam Analisis Sentimen Terhadap Tokoh Publik. *Media Online*, 4(6), 2813–2821. <https://doi.org/10.30865/klik.v4i6.1928>
- Cabui, C. E., Ilmu, D., & Universitas, P. (2022). *REFLEKSI KINERJA PEMERINTAHAN PADA PERIODE KEDUA PEMERINTAHAN*. 1(April), 221–225.
- Carneiro, T., Da Nobrega, R. V. M., Nepomuceno, T., Bian, G. Bin, De Albuquerque, V. H. C., & Filho, P. P. R. (2018). Performance Analysis of Google Colaboratory as a Tool for Accelerating Deep Learning Applications. *IEEE Access*, 6, 61677–61685. <https://doi.org/10.1109/ACCESS.2018.2874767>
- Chong, K. S., & Shah, N. (2022). Comparison of Naive Bayes and SVM Classification in Grid-Search Hyperparameter Tuned and Non-Hyperparameter Tuned Healthcare Stock Market Sentiment Analysis. *International Journal of Advanced Computer Science and Applications*, 13(12), 90–94. <https://doi.org/10.14569/IJACSA.2022.0131213>
- Chong, K., & Shah, N. (n.d.). Comparison of Naive Bayes and SVM Classification in Grid-Search Hyperparameter Tuned and Non-Hyperparameter Tuned Healthcare Stock Market Sentiment Analysis. In *IJACSA International Journal of Advanced Computer Science and Applications* (Vol. 13, Issue 12). [www.ijacsa.thesai.org](http://www.ijacsa.thesai.org)
- Ernawati, N. W., Satya Kumara, I. N., & Setiawan, W. (2023). Perbandingan Metode Klasifikasi Support Vector Machine Dan Naive Bayes Pada Analisis Sentimen Kendaraan Listrik. *Jurnal SPEKTRUM*, 10(3), 106. <https://doi.org/10.24843/spektrum.2023.v10.i03.p12>
- Iriananda, S. W., Budiawan, R. W., Rahman, A. Y., & Istiadi, I. (2024). Optimasi Klasifikasi Sentimen Komentar Pengguna Game Bergerak Menggunakan Svm, Grid Search Dan Kombinasi N-Gram. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 11(4), 743–752. <https://doi.org/10.25126/jtiik.1148244>
- Karnando, J., Muskhir, M., & Luthfi, A. (2024). Exploring Python Programming: A Project Based Learning-Centric Learning Experience. *Journal of Education Technology*, 8(2), 306–314. <https://doi.org/10.23887/jet.v8i2.68694>
- Luthfanida, L. (2022). Analisis Sentimen Data Twitter Menggunakan Metode Naive Bayes Dan Support Vector Machine (Svm) Tentang Presiden Jokowi 3 Periode. *Djtechno: Jurnal Teknologi Informasi*, 3(1), 5–11. <https://doi.org/10.46576/djtechno.v3i1.2143>
- Mahawardana, P. P. O., Imawati, I. A. P. F., & Dika, I. W. (2022). Analisis Sentimen Berdasarkan Opini dari Media Sosial Twitter terhadap “Figure Pemimpin” Menggunakan Python. *Jurnal Manajemen Dan Teknologi Informasi*, 12(2), 50–56. <https://ojs.mahadewa.ac.id/index.php/jmti/article/view/2111>
- Nalatissifa, H., Gata, W., Diantika, S., & Nisa, K. (2021). *Perbandingan Kinerja Algoritma Klasifikasi Naive Bayes, Support Vector Machine (SVM), dan Random Forest untuk Prediksi Ketidakhadiran di Tempat Kerja*. 5(4), 578–584.
- Naufal, M., Abrari, M., Abdulloh, F. F., & Amikom, U. (2024). *COMPARISON OF MACHINE LEARNING ALGORITHMS FOR SENTIMENT*. 20(2), 146–154. <https://doi.org/10.33480/pilar.v20i2.5736>
- Nazar, R. (2024). Implementasi Pemrograman Python Menggunakan Google Colab. *Jurnal Informatika Dan Komputer (JIK)*, 15(1), 50–56.
- Nehe, P. H., Berutu, S. S., & Budiati, H. (2024). Analisis Sentimen Opini Masyarakat Terhadap Presiden Jokowi Sebelum Dan Sesudah Pilpres 2024 Menggunakan Metode Naive Bayes Classification. *Jutisi : Jurnal Ilmiah Teknik Informatika Dan Sistem Informasi*, 13(1), 451. <https://doi.org/10.35889/jutisi.v13i1.1841>
- Supian, A., Tri Revaldo, B., Marhadi, N., Efrizoni, L., & Rahmaddeni, R. (2024). Perbandingan Kinerja Naive Bayes Dan Svm Pada Analisis Sentimen Twitter Ibukota Nusantara. *Jurnal Ilmiah Informatika*, 12(01), 15–21. <https://doi.org/10.33884/jif.v12i01.8721>