

# Komparasi Algoritma Random Forest, Naïve Bayes, dan SVM pada Sentimen Kebijakan PPN 12%

Dandi Purnomo<sup>1,\*</sup>, Wawan Firgiawan<sup>2</sup>, Nahya Nur<sup>3</sup>

<sup>1,2,3</sup>Fakultas Teknik, Program Studi Informatika, Universitas Sulawesi Barat, Majene  
 Jalan Prof. Dr. Baharuddin Lopa, S.H, Talumung, Kabupaten Majene, Sulawesi Barat 91412, Indonesia  
 Email: <sup>1,\*</sup>dandi.siswa01@email.com, <sup>2</sup>wawanfirgiawan@unsulbar.com, <sup>3</sup>nahya.nur@unsulbar.ac.id

**Abstrak**—Media sosial, salah satunya Twitter, telah berkembang menjadi platform penting untuk orang-orang menyuarakan pendapat mereka, mengkritik, dan mendukung berbagai kebijakan pemerintah. Rencana untuk menaikkan Pajak Pertambahan Nilai (PPN) menjadi 12% di Indonesia pada tahun 2025 adalah salah satu kebijakan yang banyak dibicarakan publik. Dengan menggunakan metode *machine learning*, penelitian ini bertujuan untuk menganalisis persepsi publik terhadap kebijakan tersebut. Dalam penelitian ini, Random Forest, Naive Bayes, dan Support Vector Machine (SVM) digunakan. Data yang digunakan terdiri dari 2747 *tweet* yang didapatkan melalui metode *crawling* menggunakan kata kunci "PPN 12%" dari April 2024 hingga Januari 2025. Setelah melalui tahapan *preprocessing*, data diberi label menggunakan kamus leksikon bahasa Indonesia dengan tiga kelas sentimen yakni positif, netral, dan negatif. Jumlah data positif sebanyak 689, data netral 544, dan data negatif 1460, yang menunjukkan adanya ketidakseimbangan kelas. Meskipun demikian, penelitian ini tidak menggunakan teknik augmentasi seperti *oversampling* atau *undersampling*. Sebagai gantinya, metrik evaluasi yang digunakan tidak hanya akurasi, tetapi juga *precision*, *recall*, dan *f1-score* pada masing-masing kategori dengan pendekatan *weighted average*, guna menghasilkan evaluasi yang lebih adil dan komprehensif. 2154 data latih dan 539 data uji dihasilkan setelah *dataset* dibagi menjadi 80% data latih dan 20% data uji secara acak. Data latih nantinya akan digunakan untuk pembangunan model, sementara data uji digunakan untuk mengevaluasi performanya. Metode *Term Frequency-Inverse Document Frequency* (TF-IDF), yang membantu dalam menentukan seberapa relevan suatu kata yang ada dalam dokumen, digunakan dalam proses ekstraksi fitur penelitian ini. Hasil pengujian menunjukkan bahwa SVM memiliki performa terbaik dengan akurasi 76,25%, diikuti oleh Random Forest 72,35%, dan Naive Bayes 60,66%. SVM juga memiliki nilai *precision*, *recall*, dan *f1-score* tertinggi dibandingkan dua algoritma lainnya, yang menunjukkan kemampuan algoritma untuk memberikan hasil prediksi yang lebih tepat dan seimbang. Hasilnya menunjukkan bahwa SVM paling efektif menangani data opini publik yang cenderung tidak eksplisit dan penuh nuansa di media sosial. Penelitian ini memberikan kontribusi dalam pemanfaatan data Twitter sebagai alat bantu analisis kebijakan publik, serta mendorong pemanfaatan *machine learning* dalam memahami persepsi masyarakat terhadap isu nasional.

**Kata Kunci:** Analisis Sentimen, Pajak Pertambahan Nilai, Random Forest, Naïve Bayes, Support Vector Machine

**Abstract**—Social media, particularly Twitter, has become an important platform for people to express their opinions, criticisms, and support regarding various government policies. One widely discussed policy is the planned increase of the Value Added Tax (VAT) to 12% in Indonesia, set to be implemented in 2025. The purpose of this study is to use machine learning techniques to examine how the public views this policy. The research employs three algorithms: Random Forest, Naïve Bayes, and Support Vector Machine (SVM). The dataset consists of 2747 tweets collected through crawling with the keyword "PPN 12%" from April 2024 to January 2025. After preprocessing, the tweets were labeled using an Indonesian lexicon into three sentiment categories: positive (689), neutral (544), and negative (1460), indicating class imbalance. Although no data augmentation techniques such as oversampling or undersampling were applied, evaluation metrics included accuracy, precision, recall, and F1-score using a weighted average approach to ensure a fair and comprehensive assessment. The dataset was randomly split into 80% training data (2154 tweets) and 20% testing data (539 tweets), with training data used to build the models and testing data to evaluate their performance. Feature extraction was performed using the Term Frequency-Inverse Document Frequency (TF-IDF) method to assess the relevance of words within the tweets. Results show that SVM achieved the best performance with an accuracy of 76.25%, followed by Random Forest at 72.35% and Naïve Bayes at 60.66%. SVM also recorded the highest precision, recall, and f1-score among the three algorithms, demonstrating its superior ability to provide accurate and balanced predictions. These findings suggest that SVM is most effective in handling nuanced and implicit public opinion data on social media. This study contributes to the utilization of Twitter data as a tool for public policy analysis and encourages the application of machine learning to better understand public perception of national issues.

**Keywords:** Sentiment Analysis, Value Added Tax, Random Forest, Naïve Bayes, Support Vector Machine

## 1. PENDAHULUAN

Perkembangan internet telah mengubah cara orang berkomunikasi, dengan banyaknya platform media sosial yang memungkinkan komunikasi daring [1]. Pertumbuhan pesat media sosial dalam beberapa tahun terakhir menjadikan internet sebagai sarana yang efektif untuk menyebarkan pengetahuan. Penggunaan *big data* dari media sosial dapat dimanfaatkan untuk memprediksi kejadian di dunia nyata [2]. Salah satu platform populer di Indonesia adalah Twitter, yang didirikan oleh Jack Dorsey dan digunakan untuk mengirim pesan singkat atau *tweet* [3]. Sebagai platform yang dinamis dengan jumlah pengguna yang cepat meningkat, Twitter menjadi sumber data penting untuk memahami perspektif dan emosi penggunanya [4]. Sementara itu, Pajak Pertambahan Nilai (PPN) adalah pajak yang dikenakan pada transaksi yang melibatkan barang dan jasa, baik domestik maupun internasional [5]. Berdasarkan Undang-Undang Nomor 7 Tahun 2021 tentang Harmonisasi Peraturan Perpajakan,

pemerintah berencana menaikkan tarif PPN dari 11% menjadi 12% mulai 1 Januari 2025. Kenaikan ini diharapkan dapat meningkatkan pendapatan negara guna mendukung pemulihan ekonomi dan pembangunan [6]. Namun, rencana tersebut menuai berbagai reaksi publik, termasuk gelombang opini di media sosial seperti Twitter.

Guna memperoleh pemahaman mendalam tentang respon publik terhadap suatu kebijakan, analisis sentimen melalui platform media sosial menjadi krusial. Analisis sentimen adalah proses yang mengolah, mengekstrak, dan memahami teks yang tidak terstruktur dengan tujuan menemukan informasi tentang perasaan yang ada dalam kalimat opini [7]. Dengan memanfaatkan teknik *Natural Language Processing* (NLP), *data mining*, dan *machine learning*, analisis sentimen memungkinkan evaluasi opini masyarakat secara menyeluruh dan *real-time*, yang memberikan informasi berharga bagi para pembuat kebijakan. *Natural Language Processing* atau Pengolahan Bahasa Alami telah berkembang menjadi ranah yang memikat dalam dunia komputasi dan kecerdasan buatan, yang membolehkan mesin untuk menginterpretasikan dan menganalisis teks manusia dalam beragam bahasa [8]. Sehingga teks dapat dimasukkan ke dalam kategori positif, netral, atau negatif melalui analisis sentimen. Oleh karena itu, penggunaan algoritma klasifikasi yang akurat penting untuk menjamin efektivitas analisis sentimen.

Beberapa penelitian sebelumnya telah membandingkan kinerja berbagai algoritma dalam analisis sentimen. Studi oleh Irawan et al. [9] yang menggunakan algoritma Support Vector Machine (SVM), Random Forest, dan Naïve Bayes dalam menganalisis komentar aplikasi CapCut di Google Play Store. Hasil penelitian menunjukkan bahwa SVM menduduki posisi pertama dengan akurasi sebesar 86%, diikuti oleh Random Forest dengan 83%, dan Naïve Bayes dengan 70%. Penelitian lain oleh Miftahusalam et al. [10] membandingkan algoritma-algoritma seperti Random Forest, SVM, dan Naïve Bayes untuk menganalisis sentimen Twitter yang berkaitan dengan penghapusan tenaga honorer. Random Forest mencapai akurasi 66,67%, sementara SVM dan Naïve Bayes menunjukkan akurasi yang lebih rendah. Studi oleh Fikri et al. [11] melakukan analisis sentimen terhadap *tweet* mengenai Universitas Muhammadiyah Malang menggunakan Naïve Bayes dan SVM. Hasilnya menunjukkan bahwa Naïve Bayes memiliki akurasi tertinggi sebesar 73,65% mengalahkan SVM yang memiliki akurasi 70,20%. Sementara itu, penelitian oleh Kurniawan et al. [12] menganalisis sentimen pengguna Twitter pada aplikasi Sirekap menggunakan Naïve Bayes dan SVM, di mana SVM dengan *kernel* RBF mencapai akurasi 84,15%, sementara Naïve Bayes mencapai 77,64%. Studi lain oleh Kusuma et al. [13] yang juga menggunakan SVM dan Naïve Bayes dalam menganalisis sentimen masyarakat terhadap penerapan PPN pada renovasi dan pembangunan rumah sendiri, temuan menunjukkan bahwa SVM memiliki kinerja yang lebih baik.

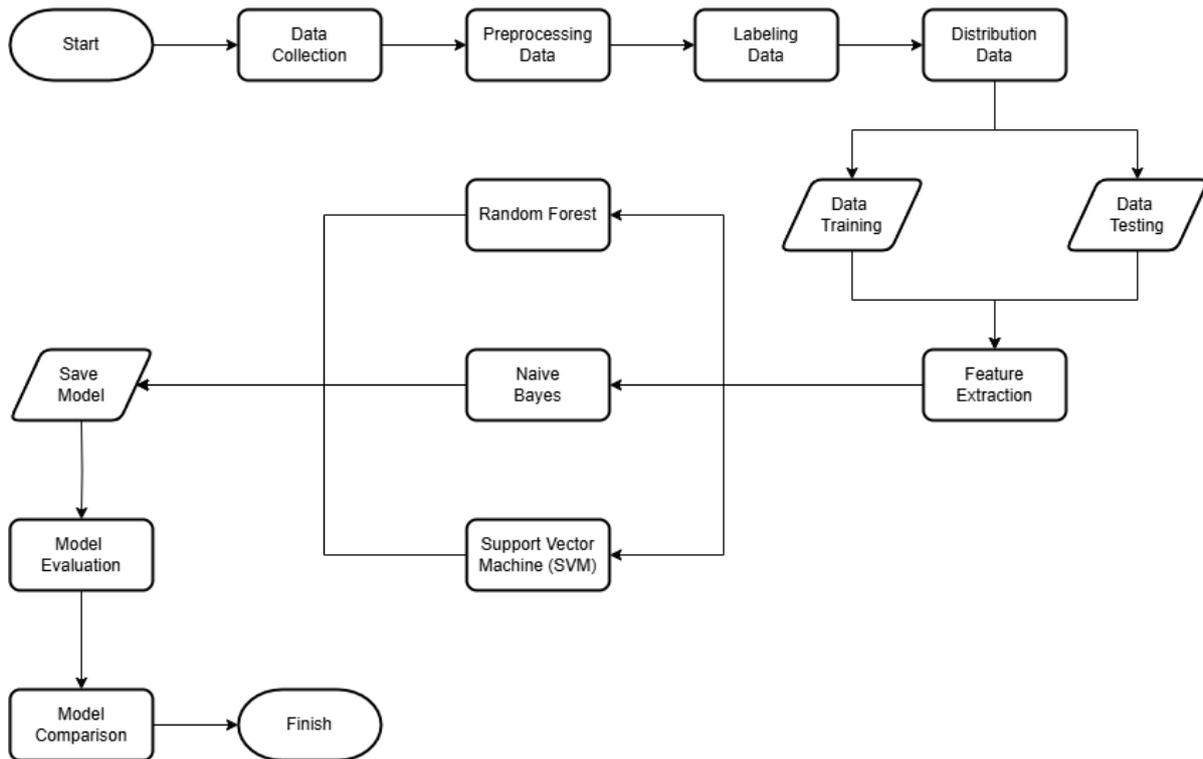
Berdasarkan penelitian-penelitian yang telah dilakukan sebelumnya, performa algoritma *machine learning* dalam analisis sentimen sangat bergantung pada konteks dan karakteristik data yang digunakan. Beberapa studi menunjukkan keunggulan Support Vector Machine (SVM) dalam hal akurasi, sementara studi lain menemukan bahwa Naïve Bayes lebih efektif. Walau bagaimanapun, belum banyak penelitian yang melakukan komparasi kinerja dari tiga algoritma (Random Forest, Naïve Bayes, dan SVM) untuk mengevaluasi reaksi publik terhadap kebijakan kenaikan PPN 12% di Indonesia.

Oleh karena itu, penelitian ini bertujuan untuk mengisi celah tersebut dengan kinerja algoritma-algoritma seperti Random Forest, Naïve Bayes, dan SVM untuk menilai respon publik terhadap kebijakan kenaikan PPN 12%. Diharapkan hasil penelitian ini dapat memberikan rekomendasi algoritma yang memiliki performa terbaik dalam analisis sentimen, khususnya dalam hal akurasi, presisi, dan efisiensi pemrosesan data, sehingga dapat menjadi referensi dalam pemilihan metode klasifikasi yang optimal untuk analisis sentimen. Selain itu, penelitian ini juga diharapkan dapat meningkatkan pemahaman kepada pembuat kebijakan tentang sentimen publik terhadap kebijakan PPN 12%, yang dapat dijadikan sebagai pertimbangan dalam proses pengambilan keputusan.

## 2. METODE PENELITIAN

### 2.1 Tahapan Penelitian

Tahapan penelitian berupa langkah-langkah penelitian untuk menggali sentimen terkait kebijakan PPN 12% pada media sosial Twitter. Metode ini bertujuan untuk melakukan komparasi kinerja antara ketiga algoritma *machine learning* yakni Random Forest, Naïve Bayes, dan Support Vector Machine (SVM). Alur penelitian, sebagaimana diilustrasikan pada Gambar 1, terdiri dari serangkaian tahapan meliputi pengumpulan data, *preprocessing* data, pelabelan data, pembagian data menjadi data latih (*training*) dan data uji (*testing*), ekstraksi fitur, pembangunan model, evaluasi model, dan visualisasi hasil komparasi. Diagram alur yang divisualisasikan dalam Gambar 1 merangkum pendekatan metodologis yang digunakan dalam penelitian ini.



Gambar 1. Tahapan Penelitian

## 2.2 Pengumpulan Data

Data dikumpulkan melalui proses *crawling*, yaitu kegiatan mengunduh data *tweet* dari Twitter menggunakan *Application Programming Interface (API)* Twitter. Proses ini melibatkan pengambilan *tweet* dan data pengguna dengan menggunakan kata kunci dalam periode waktu tertentu, seperti nama pengguna dan *hashtag* [14]. Untuk penelitian ini, *crawling* dilakukan dengan memasukkan kata kunci “PPN 12%” dari April 2024 hingga Januari 2025, menghasilkan sebanyak 2747 *tweet*.

## 2.3 Preprocessing Data

*Preprocessing* data, yaitu langkah-langkah yang dilakukan untuk menyiapkan, mengubah, dan membersihkan data mentah sehingga data siap untuk melanjutkan proses ke tahap berikutnya [15]. Dalam melakukan analisis sentimen, sebelumnya perlu melalui tahap *preprocessing* data. *Preprocessing* data ini penting untuk mengoptimalkan hasil analisis sentimen [16]. Pada tahap ini, beberapa langkah dilakukan dalam memastikan bahwa data yang akan digunakan untuk pemodelan memiliki kualitas tinggi, bebas dari inkonsistensi, dan berada dalam format yang mudah diproses oleh algoritma [17]. *Preprocessing* data yang dilakukan adalah penghapusan duplikat *tweet*, pembersihan teks, *case folding*, normalisasi teks, pemisahan dua kata yang menyatu, dan penghapusan *stopword*.

## 2.4 Pelabelan Data

Data akan dilabelkan sebelum dibagi menjadi data latih dan data uji setelah melewati tahap *preprocessing*. Untuk melakukan analisis sentimen, ulasan-ulasan tersebut harus diklasifikasikan sebagai positif, netral, atau negatif [18]. Pelabelan pada data teks dalam penelitian ini dilakukan menggunakan metode yang dikenal sebagai *lexicon-based*, di mana pelabelan dilakukan dengan memberikan bobot pada teks untuk menentukan apakah teks tersebut termasuk dalam kategori kategori positif, netral, atau negatif [15].

## 2.5 Pembagian Data

Setelah proses pelabelan, data dipisahkan menjadi dua diantaranya adalah data latih dan data uji. Data latih akan membantu algoritma menemukan model terbaik, sementara data uji akan berperan dalam mengevaluasi dan menguji kinerja model yang dihasilkan pada tahap pengujian [19]. Penelitian ini memanfaatkan 80% *dataset* sebagai data latih dan menggunakan 20% sisanya untuk pengujian.

## 2.6 Ekstraksi Fitur

Metode yang digunakan untuk ekstraksi fitur pada penelitian ini adalah *Term Frequency – Inverse Document Frequency* (TF-IDF). Proses pembobotan TF-IDF adalah metode yang diterapkan dalam pengelompokan dan analisis teks untuk mengukur seberapa signifikan suatu kata dalam konteks keseluruhan koleksi dokumen [20]. Metode ini membantu *machine learning* untuk memahami kata-kata pada *dataset*. TF-IDF merupakan metodologi pembobotan kata yang terbagi ke dalam dua komponen utama. Komponen pertama adalah *Term Frequency* (TF), yang menghitung frekuensi kemunculan suatu istilah dalam satu dokumen relatif terhadap total kata dalam dokumen tersebut. Komponen kedua, *Inverse Document Frequency* (IDF), menghitung tingkat kelangkaan istilah dalam dokumen secara keseluruhan. Nilai akhir TF-IDF didapatkan dari perkalian TF dan IDF, sehingga istilah yang kerap muncul dalam dokumen spesifik tetapi jarang ditemukan di dokumen lain akan memperoleh bobot IDF yang signifikan. Sebaliknya, istilah yang umum muncul di banyak dokumen akan memiliki nilai IDF yang minimal. Secara matematis, nilai TF-IDF dihitung dengan rumus yang dapat dilihat dalam persamaan 1, 2, dan 3.

$$TF(t, d) = \frac{f_{t,d}}{\sum_{t' \in d} f_{t',d}} \quad (1)$$

$$IDF(t) = \log\left(\frac{N}{1 + df(t)}\right) \quad (2)$$

$$TF-IDF(t, d) = TF(t, d) \times IDF(t) \quad (3)$$

Penjelasan :

- $f_{t,d}$  : Jumlah kemunculan term  $t$  dalam dokumen  $d$
- $\sum_{t' \in d} f_{t',d}$  : Total kata dalam dokumen  $d$
- $N$  : Total jumlah dokumen
- $df(t)$  : Jumlah dokumen yang mengandung term  $t$

## 2.7 Pembangunan Model

### 2.7.1 Random Forest

Random Forest adalah algoritma pembelajaran mesin berbasis kelompok yang meningkatkan akurasi prediksi dengan menggabungkan beberapa pohon keputusan. Selama fase pelatihan, algoritma ini membuat banyak pohon keputusan (*decision tree*) dan memberikan hasil prediksi berdasarkan suara mayoritas untuk klasifikasi atau rata-rata untuk regresi [21]. Rumus Random Forest dijelaskan dalam persamaan 4.

$$f(x) = \text{Average}\left(f_1(x), f_2(x), \dots, f_n(x)\right) \quad (4)[7]$$

Penjelasan :

- $f(x)$  : Prediksi akhir
- $f_{1-n}(x)$  : Prediksi dari masing-masing *decision tree* ke- $n$
- $(x)$  : Pengisian data

### 2.7.2 Naïve Bayes

Metode pengklasifikasian Naive Bayes adalah pendekatan probabilistik untuk pengklasifikasian yang sederhana. Berdasarkan Teorema Bayes, metode ini memungkinkan klasifikasi yang efisien melalui set pelatihan yang terdiri dari sejumlah data [22]. Selama proses klasifikasi, nilai probabilitas tertinggi digunakan [23]. Rumus Naïve Bayes dijelaskan dalam persamaan 5.

$$P(H|X) = \frac{P(X|H) \cdot P(H)}{P(X)} \quad (5)[7]$$

Penjelasan :

- $X$  : Data dengan kelas yang tidak diketahui
- $H$  : Hipotesis data termasuk dalam kategori tertentu
- $P(H|X)$  : Kemungkinan hipotesis H berdasarkan kondisi X (posteriori probabilitas)
- $P(H)$  : Kemungkinan hipotesis H (prior probabilitas)
- $P(X|H)$  : Kemungkinan X berdasarkan kondisi pada hipotesis H
- $P(X)$  : Kemungkinan X

### 2.7.3 Support Vector Machine (SVM)

Salah satu algoritma pembelajaran mesin yang digunakan untuk tugas klasifikasi dan regresi adalah Support Vector Machine (SVM). SVM bekerja dengan menemukan *hyperplane* optimal untuk memisahkan data ke dalam dua kelas yang berbeda. Algoritma ini dikenal karena kemampuan untuk menangani data berukuran besar [24]. Rumus SVM dijelaskan dalam persamaan 6 dan 7.

$$(w \cdot x_i + b) \leq 1, \quad y_i = -1 \quad (6)$$

$$(w \cdot x_i + b) \geq 1, \quad y_i = 1 \quad (7)[7]$$

Penjelasan :

- $x_i$  : Data ke-i
- $w \cdot x_i$  : Nilai berat kelas data ke-i
- $b$  : Nilai yang bias
- $y_i$  : Kelas data ke-i

### 2.8 Evaluasi Model

*Confusion matrix* digunakan untuk menilai kinerja model penelitian ini. Tabel *confusion matrix* menunjukkan berapa banyak data uji yang diklasifikasikan dengan benar atau salah, memudahkan evaluasi akurasi sistem klasifikasi [25]. Selain itu, untuk melengkapi evaluasi performa model secara lebih menyeluruh, penelitian ini juga menggunakan metrik *precision*, *recall*, dan *f1-score*. Hal ini dilakukan karena metrik ini mampu memberikan informasi yang lebih detail tentang kinerja klasifikasi masing-masing kelas, terutama dalam situasi di mana distribusi data tidak seimbang. Nilai *precision*, *recall*, dan *f1-score* diambil dari *weighted average* untuk mempertimbangkan proporsi tiap kelas dalam perhitungan keseluruhan.

## 3. HASIL DAN PEMBAHASAN

### 3.1 Pengumpulan Data

Pengumpulan data melalui proses *crawling* dilakukan dengan memasukkan kata kunci “PPN 12%” dimulai dari bulan April 2024 hingga Januari 2025 dan mendapatkan sebanyak 2747 data *tweet*. Data dikumpulkan menggunakan Google Colab dan Python. Selanjutnya, data yang dikumpulkan disimpan dalam bentuk CSV untuk digunakan dalam analisis sentimen. Tabel 1 menunjukkan sampel *dataset* awal yang digunakan untuk pemodelan.

**Tabel 1.** Contoh sampel *dataset* awal

Sampel <i>dataset</i> awal
Menurut gw PPN 12% ini udh keterlalu aja, harga barang2 jadi makin mahal ☹️ spt biasa, yg kena dampak palingbesar ya pasti rakyat kecil lagi. @KemenkeuRI tolong dong evaluasi kebijakan ini! #PPN12% #EkonomiRakyat

### 3.2 Preprocessing Data

#### 3.2.1 Penghapusan Duplikat Tweet

Penghapusan duplikat *tweet* adalah proses menghilangkan *tweet* yang memiliki isi teks yang sama untuk mencegah data redundan. Tahap ini dilakukan untuk membuat *dataset* yang digunakan menjadi lebih bersih, meningkatkan akurasi data dan meningkatkan performa saat pemodelan dilakukan. Setelah melakukan proses penghapusan duplikat *tweet*, *dataset* yang awalnya memiliki 2747 data sekarang menjadi 2693 data *tweet*. Setelah proses penghapusan duplikat selesai maka *dataset* siap untuk menuju tahap berikutnya.

#### 3.2.2 Pembersihan Teks

Tahap selanjutnya adalah pembersihan teks, pada tahap ini karakter yang kurang penting untuk analisis sentimen akan dihapus. Karakter yang dimaksud seperti non-ascii, *url/link*, *mentions*, *hashtags*, simbol dan karakter khusus (@, \$, %, &, !, ?), angka, spasi ganda, spasi di awal dan akhir kalimat. Contoh hasil dari pembersihan teks dapat dilihat pada Tabel 2.

**Tabel 2.** Contoh sebelum dan setelah pembersihan

Sebelum pembersihan	Setelah pembersihan
Menurut gw PPN 12% ini udh keterlalu aja, harga barang2 jadi makin mahal ☹️ spt biasa, yg kena dampak palingbesar ya pasti rakyat kecil lagi.	Menurut gw PPN ini udh keterlalu aja harga barang jadi makin mahal spt biasa yg kena dampak

@KemenkeuRI tolong dong evaluasi kebijakan ini!  
#PPN12% #EkonomiRakyat

palingbesar ya pasti rakyat kecil lagi tolong dong  
evaluasi kebijakan ini

### 3.2.3 Case Folding

*Case folding* dilakukan dengan tujuan meningkatkan akurasi dengan menyeragamkan format teks dengan mengubah teks menjadi *lowercase* atau huruf kecil. Contoh hasil dari *case folding* dapat dilihat pada Tabel 3.

**Tabel 3.** Contoh sebelum dan setelah *case folding*

Sebelum <i>case folding</i>	Setelah <i>case folding</i>
Menurut gw PPN ini udh keterlalu an aja harga barang jadi makin mahal spt biasa yg kena dampak palingbesar ya pasti rakyat kecil lagi tolong dong evaluasi kebijakan ini	menurut gw ppn ini udh keterlalu an aja harga barang jadi makin mahal spt biasa yg kena dampak palingbesar ya pasti rakyat kecil lagi tolong dong evaluasi kebijakan ini

### 3.2.4 Normalisasi Teks

Dalam tahap normalisasi ini, kata-kata yang tidak baku, singkatan atau slang, akan diubah menjadi kata baku untuk mempermudah proses pelabelan dan meningkatkan akurasi pemodelan. Contoh hasil dari normalisasi teks dapat dilihat pada Tabel 4.

**Tabel 4** Contoh sebelum dan setelah normalisasi

Sebelum normalisasi	Setelah normalisasi
menurut gw ppn ini udh keterlalu an aja harga barang jadi makin mahal spt biasa yg kena dampak palingbesar ya pasti rakyat kecil lagi tolong dong evaluasi kebijakan ini	menurut gue ppn ini sudah keterlalu an aja harga barang jadi makin mahal seperti biasa yang kena dampak palingbesar ya pasti rakyat kecil lagi tolong dong evaluasi kebijakan ini

### 3.2.5 Pemisahan Dua Kata yang Menyatu

Tahap selanjutnya adalah pemisahan dua kata yang menyatu dengan bantuan dari daftar kata kata dasar bahasa Indonesia, bertujuan untuk memisahkan kata jika memang di dalam *dataset* ada dua kata yang menyatu. Contoh hasil dari pemisahan dua kata yang menyatu dapat dilihat pada Tabel 5.

**Tabel 5.** Contoh sebelum dan setelah pemisahan

Sebelum pemisahan	Setelah pemisahan
menurut gue ppn ini sudah keterlalu an aja harga barang jadi makin mahal seperti biasa yang kena dampak palingbesar ya pasti rakyat kecil lagi tolong dong evaluasi kebijakan ini	menurut gue ppn ini sudah keterlalu an aja harga barang jadi makin mahal seperti biasa yang kena dampak paling besar ya pasti rakyat kecil lagi tolong dong evaluasi kebijakan ini

### 3.2.6 Penghapusan Stopword

Tahap terakhir yaitu penghapusan *stopword*, tujuannya adalah untuk meningkatkan efektifitas pemodelan. Proses ini mencakup membuat daftar *stopword* dan kemudian menghapus kata-kata yang ada di dalamnya, sehingga analisis dapat berkonsentrasi pada kata-kata yang lebih relevan dan penting. Daftar *stopword* disusun secara manual dengan mempertimbangkan kata-kata yang tidak memiliki nilai emosional atau tidak memengaruhi arah sentimen terhadap kebijakan PPN 12%. Kata-kata ini umumnya berupa kata sambung, kata tunjuk, atau istilah administratif yang bersifat netral. Penyusunan dilakukan dengan meninjau frekuensi kata dalam data dan mengevaluasi relevansinya terhadap analisis sentimen. Contoh hasil dari penghapusan *stopword* dapat dilihat pada Tabel 6.

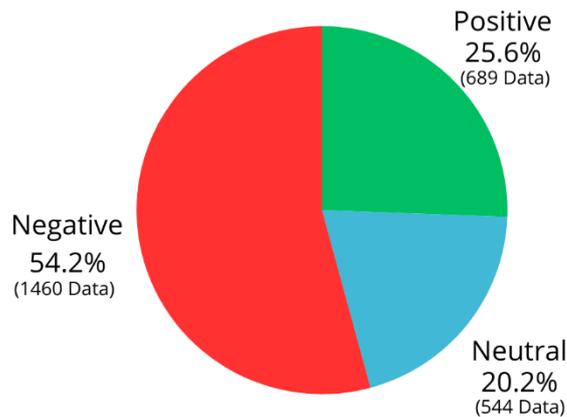
**Tabel 6.** Contoh sebelum dan setelah penghapusan

Sebelum penghapusan <i>stopword</i>	Setelah penghapusan <i>stopword</i>
menurut gue ppn ini sudah keterlalu an aja harga barang jadi makin mahal seperti biasa yang kena dampak paling besar ya pasti rakyat kecil lagi tolong dong evaluasi kebijakan ini	gue sudah keterlalu an aja harga jadi makin mahal seperti biasa yang kena dampak paling besar ya pasti rakyat kecil lagi tolong dong

## 3.3 Pelabelan Data

*Dataset* yang sudah melalui *preprocessing* data maka selanjutnya akan diberikan label. Semua data dilabeli positif, netral, dan negatif menggunakan kamus leksikon bahasa Indonesia. 689 data termasuk dalam kategori positif, 544 data termasuk dalam kategori netral, dan 1460 data termasuk dalam kategori negatif. Meskipun distribusi kelas

pada *dataset* menunjukkan ketidakseimbangan dengan jumlah data sentimen negatif (1460 data) lebih dominan dibanding positif (689 data) dan netral (544 data), penelitian ini tidak menggunakan teknik augmentasi seperti *oversampling* atau *undersampling*. Dalam kondisi ini, metrik akurasi tetap digunakan sebagai salah satu ukuran performa karena dapat memberikan gambaran umum mengenai proporsi prediksi yang benar secara keseluruhan. Namun demikian, penelitian ini melibatkan metrik tambahan yakni *precision*, *recall*, dan *f1-score* untuk masing-masing kelas untuk mencegah bias yang disebabkan oleh ketidakseimbangan kelas. Oleh karena itu, analisis yang lebih adil dan menyeluruh dapat dilakukan mengenai kinerja model terhadap kelas minoritas tetap. Gambar 2 menunjukkan persentase hasil pelabelan.



Gambar 2. Hasil Pelabelan Data

### 3.4 Pembagian Data

*Dataset* akan dibagi ke dalam dua kelompok yaitu data latih (*training*) dan data uji (*testing*). Pembagian *dataset* menjadi 80% sebagai data latih dan 20% sebagai data uji adalah pendekatan yang umum dan dianggap ideal dalam pengembangan model pembelajaran mesin. Dengan 80% data latih, model memiliki ruang yang cukup untuk mempelajari pola dan karakteristik data, sementara 20% data uji digunakan untuk menguji performa algoritma untuk memprediksi data baru, memberikan gambaran tentang bagaimana model bekerja di dunia nyata. Agar distribusi data tetap representatif dan tidak bias terhadap jenis perasaan tertentu, pembagian ini dilakukan secara acak. Fungsi *train\_test\_split* dari *library* Scikit-learn digunakan untuk pembagian data dalam penelitian ini, yang membantu menjaga keseimbangan distribusi label dan menghasilkan evaluasi model yang lebih objektif. Dalam proses pelatihan dan pengujian model, beberapa penelitian sebelumnya juga membagi *dataset* menjadi 80% data latih dan 20% data uji. Ini termasuk studi yang dilakukan oleh Elisa et al. "Comparison Of Random Forest, Support Vector Machine And Naive Bayes Algorithms To Analyze Sentiment Towards Mental Health Stigma" [7] dan studi yang dilakukan oleh Miftahusalam et al. "Perbandingan Algoritma Random Forest, Naïve Bayes, dan Support Vector Machine Pada Analisis Sentimen Twitter Mengenai Opini Masyarakat Terhadap Penghapusan Tenaga Honorar" [10]. Data latih berjumlah 2154 data dan data uji berjumlah 539 data yang akan digunakan untuk melatih algoritma untuk menghasilkan model terbaik.

Dalam penelitian ini, data tidak dibagi secara eksplisit ke dalam *subset* validasi karena model yang digunakan tidak melalui proses *hyperparameter tuning*. Penyesuaian parameter hanya dilakukan pada algoritma SVM, yakni dengan mengubah *kernel* menjadi *linear* lalu mengaktifkan opsi *probability=True* dan *break\_ties=True*, tanpa memerlukan proses validasi berulang. Oleh karena itu, dua *subset* data (latih dan uji) dinilai cukup untuk mengevaluasi performa dasar dari masing-masing model yang digunakan dalam analisis.

### 3.5 Ekstraksi Fitur

TF-IDF digunakan untuk ekstraksi fitur pada tahap ini. Tujuan dari teknik ini adalah untuk memberi bobot pada kata yang ada untuk menentukan seberapa penting kata tersebut dalam *dataset*, sehingga model dapat memahami konteks dan karakter kalimat dalam *dataset*. Dalam penelitian ini tahap TF-IDF menggunakan satu parameter lagi yaitu (*min\_df=2*) yang bertujuan mengabaikan kata yang muncul kurang dari dua kali dalam *dataset* sehingga dapat mengurangi *noise* dan dapat meningkatkan kualitas analisis. Tabel 7 menunjukkan contoh hasil ekstraksi fitur menggunakan TF-IDF.

Tabel 7. Contoh dari ekstraksi fitur menggunakan TF-IDF

Kata	Bobot TF-IDF
gue	0.12

sudah	0.08
keterlaluan	0.15
aja	0.05
harga	0.18
jadi	0.10
makin	0.09
mahal	0.20
seperti	0.07
biasa	0.06
yang	0.04
kena	0.14
dampak	0.17
paling	0.09
besar	0.13
ya	0.05
pasti	0.08
rakyat	0.19
kecil	0.18
lagi	0.07
tolong	0.16
dong	0.06

### 3.6 Pembangunan Model

Dalam tahap pembangunan model atau pemodelan, klasifikasi sentimen dilakukan dengan menggunakan tiga algoritma *machine learning*: Random Forest, Naïve Bayes, dan SVM. Algoritma-algoritma ini digunakan dengan menggunakan bahasa pemrograman Python. Proses pelatihan algoritma atau model Random Forest, Naïve Bayes, dan SVM ini menggunakan data latih yang telah melalui proses ekstraksi fitur. Setelah dilatih, model digunakan untuk memprediksi sentimen data uji untuk mengevaluasi kinerja model.

### 3.7 Evaluasi Model

Setelah pembangunan model selesai, maka selanjutnya model akan di evaluasi. Penjelasan mengenai evaluasi ketiga algoritma tersebut akan disajikan pada bagian berikut.

#### 3.7.1 Random Forest

Random Forest menunjukkan hasil evaluasi bahwa kelas positif dan negatif memiliki nilai *precision* yang sama (77%), *recall* tertinggi (90%), dan *f1-score* tertinggi (83%) ditemukan di kelas negatif. Random Forest menggunakan 100 pohon keputusan. Akurasi model Random Forest adalah sebesar 72,35%. Hasil evaluasi Random Forest dapat dilihat pada Tabel 8.

**Tabel 8.** Hasil evaluasi model Random Forest

	<i>precision</i>	<i>recall</i>	<i>f1-score</i>	<i>support</i>
<b>Negatif</b>	0.77	0.90	0.83	301
<b>Netral</b>	0.44	0.36	0.40	94
<b>Positif</b>	0.77	0.59	0.67	144
<i>accuracy</i>			0.72	539
<i>macro avg</i>	0.66	0.62	0.63	539
<i>weighted avg</i>	0.71	0.72	0.71	539

### 3.7.2 Naïve Bayes

Naïve Bayes menunjukkan hasil evaluasi bahwa kelas positif memiliki nilai *precision* tertinggi (90%), *recall* tertinggi (99%), dan *f1-score* tertinggi (74%) ditemukan di kelas negatif. Akurasi model Naïve Bayes adalah sebesar 60,66%. Hasil evaluasi Naïve Bayes dapat dilihat pada Tabel 9.

**Tabel 9.** Hasil evaluasi model Naïve Bayes

	<i>precision</i>	<i>recall</i>	<i>f1-score</i>	<i>support</i>
<b>Negatif</b>	0.59	0.99	0.74	301
<b>Netral</b>	0.25	0.02	0.04	94
<b>Positif</b>	0.90	0.19	0.32	144
<i>accuracy</i>			0.61	539
<i>macro avg</i>	0.58	0.40	0.37	539
<i>weighted avg</i>	0.62	0.61	0.51	539

### 3.7.3 Support Vector Machine (SVM)

SVM menunjukkan hasil evaluasi bahwa nilai *precision* tertinggi (82%), *recall* tertinggi (92%), dan *f1-score* tertinggi (87%) ditemukan di kelas negatif. SVM menggunakan *kernel linear* dengan parameter *probability true* dan *break\_ties true* yang membuat model akan memilih kelas dengan skor keputusan (*decision function*) tertinggi, sehingga meningkatkan akurasi prediksi. Akurasi model SVM adalah sebesar 76,25%. Hasil evaluasi SVM dapat dilihat pada Tabel 10.

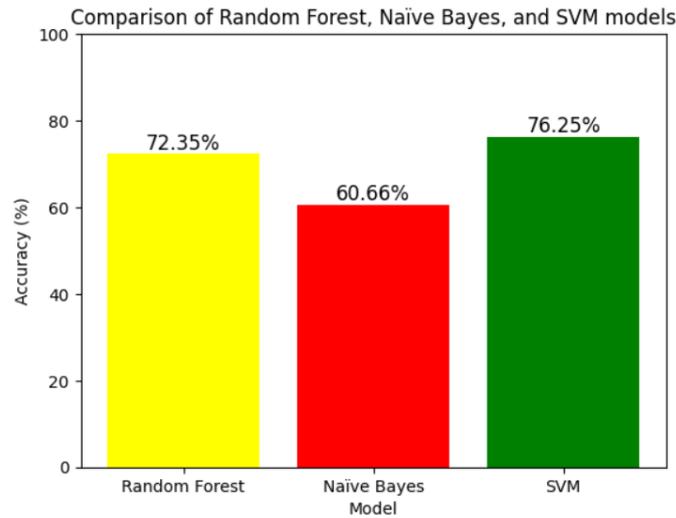
**Tabel 10.** Hasil evaluasi model SVM

	<i>precision</i>	<i>recall</i>	<i>f1-score</i>	<i>support</i>
<b>Negatif</b>	0.82	0.92	0.87	301
<b>Netral</b>	0.37	0.27	0.31	94
<b>Positif</b>	0.80	0.76	0.78	144
<i>accuracy</i>			0.76	539
<i>macro avg</i>	0.67	0.65	0.65	539
<i>weighted avg</i>	0.74	0.76	0.75	539

### 3.8 Komparasi Model

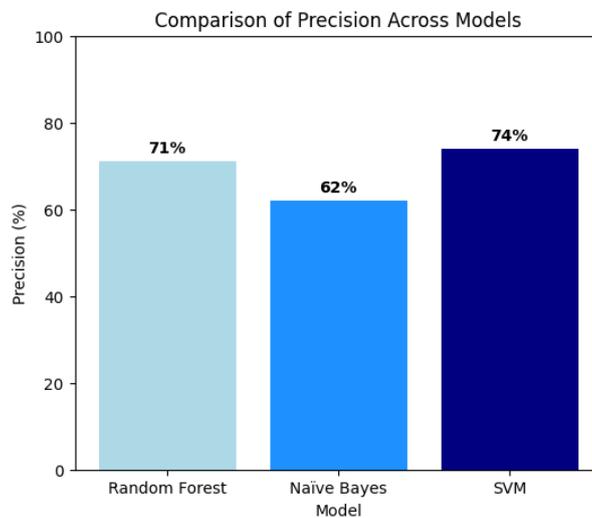
Hasil penelitian yang dilakukan untuk membandingkan akurasi model Random Forest, Naive Bayes, dan SVM terbukti bahwa yang memiliki akurasi tertinggi adalah SVM sebesar 76,25%, Random Forest berada di posisi kedua dengan akurasi 72,35%, dan Naïve Bayes dengan akurasi 60,66%. Akurasi didapatkan tanpa *hyperparameter tuning*. Random Forest memiliki nilai *precision* yang sama sebesar 77% pada kelas positif dan negatif, dengan *recall* tertinggi mencapai 90% dan *f1-score* tertinggi sebesar 83% untuk kelas negatif. Di sisi lain, Naïve Bayes mencatat *precision* tertinggi sebesar 90%, meskipun akurasi keseluruhan rendah, dengan *recall* kelas negatif mencapai 99% dan *f1-score* sebesar 74% untuk kelas yang sama. Sementara itu, SVM memiliki nilai *precision* tertinggi sebesar 82%, *recall* tertinggi mencapai 92%, serta *f1-score* tertinggi sebesar 87% pada kelas

negatif. Visualisasi perbandingan dapat dilihat pada Gambar 3 yang berupa diagram yang jelas menggambarkan performa masing-masing algoritma. Gambar 4, 5 dan 6 menunjukkan visualisasi *precision*, *recall*, dan *f1-score* setiap algoritma, yang angkanya diperoleh dari *weighted avg*.



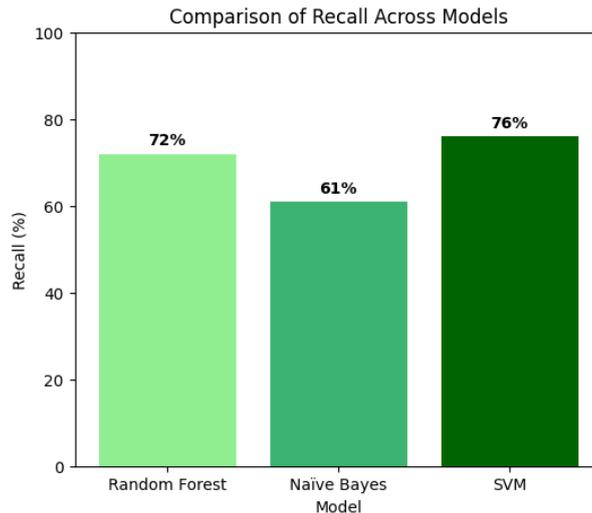
**Gambar 3.** Hasil Komparasi Model

Pada gambar 3, setiap batang diagram menunjukkan akurasi dari model yang diuji yaitu, Random Forest, Naive Bayes, dan SVM. Diagram ini memberikan gambaran singkat tentang seberapa efektif masing-masing model dalam analisis sentimen terhadap kebijakan PPN 12%. Hasil menunjukkan bahwa SVM memiliki akurasi tertinggi yaitu 76,25%, disusul oleh Random Forest yang memiliki akurasi 72,35%, dan terakhir Naive Bayes yang memiliki akurasi 60,66%. Dengan demikian, gambar ini menyoroti keunggulan SVM sebagai algoritma yang paling efektif dalam penelitian ini.



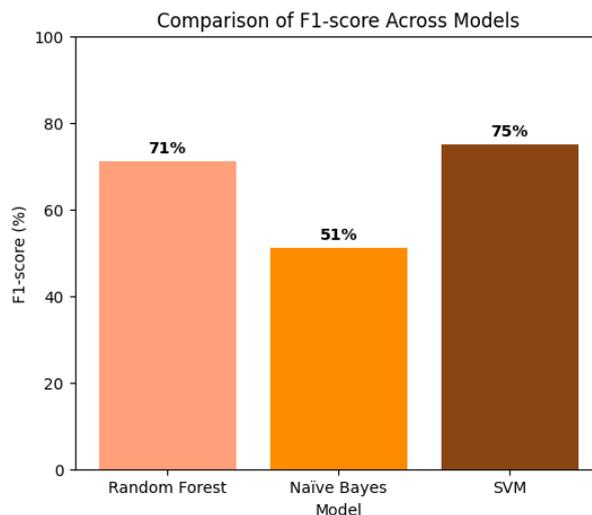
**Gambar 4.** Precision dari Setiap Model

Pada gambar 4 menunjukkan perbandingan *precision* dari ketiga model, di mana SVM memiliki *precision* tertinggi sebesar 74%, diikuti oleh Random Forest dengan 71%, dan Naive Bayes dengan 62%. *Precision* yang lebih tinggi mengindikasikan bahwa model lebih efektif dalam memberikan prediksi yang akurat dan meminimalkan kesalahan klasifikasi. Dengan demikian, SVM menunjukkan performa terbaik dalam menghasilkan prediksi yang lebih tepat dibandingkan dua model lainnya.



**Gambar 5.** Recall dari Setiap Model

Pada gambar 5 *recall* dari setiap model ditampilkan, dengan hasil tertinggi dicapai oleh SVM sebesar 76%, diikuti oleh Random Forest dengan 72%, dan terakhir Naïve Bayes dengan 61%. *Recall* yang lebih tinggi menunjukkan bahwa model memahami data yang seharusnya diklasifikasikan dengan benar dengan lebih baik. SVM kembali unggul dalam hal ini, sementara Naïve Bayes memiliki nilai *recall* terendah, yang berarti model ini cenderung lebih banyak melewatkan data yang sebenarnya relevan.



**Gambar 6.** F1-score dari Setiap Model

Pada gambar 6 *f1-score* menunjukkan perbandingan antara ketiga model, di mana SVM memiliki nilai tertinggi sebesar 75%, diikuti oleh Random Forest dengan 71%, dan Naïve Bayes dengan 51%. *F1-score* adalah metrik yang mengintegrasikan *precision* dan *recall* guna memberikan penilaian komprehensif mengenai keseimbangan antara kedua metrik tersebut. Dengan *f1-score* tertinggi, SVM menunjukkan kinerja paling optimal untuk menghasilkan prediksi yang akurat dan konsisten.

Berdasarkan hasil perbandingan *precision*, *recall*, dan *f1-score*, dapat disimpulkan bahwa SVM merupakan model dengan performa terbaik dalam analisis sentimen kebijakan PPN 12%, diikuti oleh Random Forest, sementara Naïve Bayes menunjukkan performa terendah. Nilai yang ditampilkan dalam gambar di atas diambil berdasarkan *weighted average*, sehingga mencerminkan kinerja keseluruhan model pada berbagai kelas sentimen.

#### 4. KESIMPULAN

Penelitian ini bertujuan untuk membandingkan kinerja tiga algoritma klasifikasi yaitu Random Forest, Naïve Bayes, dan Support Vector Machine (SVM), dalam menilai persepsi masyarakat terhadap kebijakan PPN 12%

menggunakan data yang diperoleh dari platform Twitter. *Dataset* yang digunakan mencakup 2747 *tweet* yang diperoleh melalui proses *crawling* menggunakan kata kunci “PPN 12%” pada rentang waktu April 2024 hingga Januari 2025. Setelah melalui tahapan *preprocessing* dan pelabelan, data dibagi menjadi 80% data latih dan 20% data uji secara acak untuk menjaga distribusi sentimen tetap seimbang. Jumlah data pelatihan sebanyak 2154, dan data uji sebanyak 539, dengan distribusi sentimen meliputi 689 data positif, 544 data netral, dan 1460 data negatif. Hasil komparasi menunjukkan bahwa algoritma SVM memberikan performa paling tinggi dengan akurasi sebesar 76,25%, diikuti oleh Random Forest sebesar 72,35%, dan Naïve Bayes sebesar 60,66%. Karena *dataset* yang digunakan mengalami ketidakseimbangan kelas, penelitian ini juga mengukur kinerja dengan menggunakan *precision*, *recall*, dan *f1-score* untuk meningkatkan hasil evaluasi. Metrik ini diambil berdasarkan *weighted average* untuk mempertimbangkan distribusi masing-masing kelas secara proporsional.

Secara keseluruhan, SVM tidak hanya unggul dalam akurasi tetapi juga memiliki nilai *precision*, *recall*, dan *f1-score* tertinggi dibandingkan dua algoritma lainnya. Ini menunjukkan kemampuan algoritma untuk memberikan hasil prediksi yang lebih tepat dan seimbang. Hal ini mengindikasikan bahwa SVM memiliki kemampuan yang lebih besar untuk menangani atribut data sentimen Twitter yang kompleks dan tidak selalu eksplisit.

Namun demikian, penelitian ini juga memiliki keterbatasan, terutama dalam aspek pelabelan data yang memiliki potensi bias, termasuk dalam mengenali sarkasme. Selain itu, model yang digunakan belum dioptimasi melalui proses *hyperparameter tuning*, sehingga masih terdapat ruang untuk peningkatan performa. Penelitian selanjutnya disarankan untuk menggunakan pendekatan pelabelan dengan metode yang lebih baik, menyertakan teknik optimasi parameter, serta mengeksplorasi algoritma *deep learning* yang berpotensi memberikan hasil klasifikasi yang lebih akurat dalam menangani data teks yang bersifat dinamis dan kontekstual seperti Twitter.

## REFERENCES

- [1] M. Fachriza and Munawar, “Analisis Sentimen Kalimat Depresi Pada Pengguna Twitter Dengan Naive Bayes, Support Vector Machine, Random Forest,” 2023. [Online]. Available: <http://studentjournal.umpo.ac.id/index.php/komputek>
- [2] S. Sekioka, R. Hatano, and H. Nishiyama, “Market Prediction using Machine Learning Based on Social Media Specific Features,” *Artif Life Robot*, vol. 28, no. 2, pp. 410–417, May 2023, doi: 10.1007/s10015-023-00857-z.
- [3] T. Krisdiyanto, E. Maricha, and O. Nurharyanto, “Analisis Sentimen Opini Masyarakat Indonesia Terhadap Kebijakan PPKM pada Media Sosial Twitter Menggunakan Naïve Bayes Clasifiers,” *Jurnal CoreIT*, vol. 7, no. 1, 2021.
- [4] A. Azrul, A. I. Purnamasari, and I. Ali, “Analisis Sentimen Pengguna Twitter Terhadap Perkembangan Artificial Intelligence dengan Penerapan Algoritma Long Short-Term Memory (LSTM),” 2024.
- [5] R. E. Yani, E. Simandalahi, and A. R. Nasution, “Pengaruh PPN (Pajak Pertambahan Nilai) terhadap Pendapatan Nasional,” *Eksis: Jurnal Ilmiah Ekonomi dan Bisnis*, vol. 15, no. 1, p. 30, May 2024, doi: 10.33087/eksis.v15i1.424.
- [6] I. Mega Putri, “Kenaikan PPN 12% dan Dampaknya Terhadap Ekonomi,” vol. 8, no. 2, 2024.
- [7] P. Elisa and A. Rahman Isnain, “Comparison of Random Forest, Support Vector Machine and Naive Bayes Algorithms to Analyze Sentiment Towards Mental Health Stigma,” *Jurnal Teknik Informatika (JUTIF)*, vol. 5, no. 1, pp. 321–329, 2024, doi: 10.52436/1.jutif.2024.5.1.1817.
- [8] M. Amien, “Sejarah dan Perkembangan Teknik Natural Language Processing (NLP) Bahasa Indonesia: Tinjauan tentang sejarah, perkembangan teknologi, dan aplikasi NLP dalam bahasa Indonesia,” 2023.
- [9] I. Irawan, Wardianto, M. H. Wathan, and M. B. Prayogi, “Studi Perbandingan: Algoritma Random Forest, Naive Bayes Dan Support Vector Machine Dalam Analisis Sentimen Pada Aplikasi Capcut Di Google Play Store,” vol. Vol. 5, 2024.
- [10] A. Miftahusalam, A. Febby Nuraini, A. A. Khoirunisa, and H. Pratiwi, “Perbandingan Algoritma Random Forest, Naïve Bayes, dan Support Vector Machine Pada Analisis Sentimen Twitter Mengenai Opini Masyarakat Terhadap Penghapusan Tenaga Honorer,” 2022.
- [11] M. I. Fikri, T. S. Sabrila, and Y. Azhar, “Perbandingan Metode Naive Bayes dan Support Vector Machine pada Analisis Sentimen Twitter,” *SMATIKA JURNAL*, vol. 10, no. 02, pp. 71–76, Dec. 2020, doi: 10.32664/smatika.v10i02.455.

- [12] G. K. V. Natalianus and R. Feta, "Perbandingan Akurasi Algoritma Naïve Bayes Dan Support Vector Machine Dalam Analisis Sentimen Pengguna Twitter Terhadap Aplikasi Sirekap," *Fountain of Informatics Journal*, vol. 9, no. 2, pp. 2548–5113, 2024, doi: 10.21111/fij.v9i2.12717.
- [13] M. R. Kusuma, W. Gata, L. Kurniawati, A. F. Rivian, and T. Kurniawan, "Analisis sentimen masyarakat terhadap informasi penerapan PPN atas renovasi dan membangun rumah sendiri pada media sosial youtube dengan metode svm dan naive bayes," *Jurnal Digital Teknologi Informasi*, vol. 6, no. 2, Sep. 2023, doi: 10.32502/digital.v6i2.4758.
- [14] A. A. Syahidin, M. R. Alwi, Subandi, and M. R. Hariyadi, "Analisis Sentimen Data Twitter Terhadap Pelaksanaan Pembelajaran Online di Indonesia Pada Masa Pandemi Covid-19 Menggunakan Metode Natural Language Processing," *JURISISTEKNI (Jurnal Sistem Informasi dan Teknologi Informasi)*, 2023.
- [15] T. Widyanto, I. Ristiana, and A. Wibowo, "Komparasi Naïve Bayes dan SVM Analisis Sentimen RUU Kesehatan di Twitter," 2023, [Online]. Available: <https://doi.org/10.31598>
- [16] A. I. Tanggraeni and M. N. N. Sitokdana, "Analisis Sentimen Aplikasi E-Government Pada Google Play Menggunakan Algoritma Naïve Bayes," vol. 9, no. 2, pp. 785–795, 2022.
- [17] C. Michael Lauwl, Husain, B. Nadila Nuzululnisa, and H. Wijaya, "Komparasi Metode Random Forest Dan Support Vector Machine (SVM) Untuk Pemodelan Klasifikasi Serangan DDos," *Journal of Information System Research*, vol. 6, no. 2, pp. 1471–1478, 2025, doi: 10.47065/josh.v6i2.6684.
- [18] T. Misriati and R. Aryanti, "Optimalisasi Random Forest dan Support Vector Machine dengan Hyperparameter GridSearchCV untuk Analisis Sentimen Ulasan PrimaKu," *Journal of Information System Research (JOSH)*, vol. 5, no. 4, pp. 1333–1341, 2024, doi: 10.47065/josh.v5i4.5347.
- [19] A. Dirgahayu Zentrato, S. Sandino Berutu, el Pieter Sumihar, and H. Budiati, "Pengembangan Model Klasifikasi Sentimen Dengan Pendekatan Vader dan Algoritma Naive Bayes Terhadap Ulasan Aplikasi Indodax," *Journal of Information System Research (JOSH)*, vol. 5, no. 3, pp. 755–764, 2024, doi: 10.47065/josh.v5i3.5050.
- [20] R. Rifaldi *et al.*, "Analisis Sentimen Pemboikotan Produk dengan Pendekatan Algoritma Naïve Bayes Media Sosial X," *Journal of Information System Research (JOSH)*, vol. 5, no. 4, pp. 940–946, 2024, doi: 10.47065/josh.v5i4.5420.
- [21] R. Torhino and N. Andono, "Penerapan Algoritma Random Forest dalam Prediksi Curah Hujan untuk Mendukung Analisis Cuaca," *Technology and Science (BITS)*, vol. 6, no. 3, pp. 1688–1699, 2024, doi: 10.47065/bits.v6i3.6404.
- [22] H. Gemasih and I. Nizar, "Implementasi Algoritma Naive Bayes Untuk Memprediksi Tingkat Penyebaran Covid," 2022.
- [23] K. Verena, S. Toy, Y. A. Sari, and I. Cholissodin, "Analisis Sentimen Twitter menggunakan Metode Naive Bayes dengan Relevance Frequency Feature Selection (Studi Kasus: Opini Masyarakat Mengenai Kebijakan New Normal)," 2021. [Online]. Available: <http://j-ptiik.ub.ac.id>
- [24] H. Eldo, A. Ayuliana, D. Suryadi, G. Chrisnawati, and L. Judijanto, "Penggunaan Algoritma Support Vector Machine (SVM) Untuk Deteksi Penipuan pada Transaksi Online," *Jurnal Minfo Polgan*, vol. 13, no. 2, pp. 1627–1632, Oct. 2024, doi: 10.33395/jmp.v13i2.14186.
- [25] R. Nurhidayat and K. E. Dewi, "KOMPUTA : Jurnal Ilmiah Komputer dan Informatika Penerapan Algoritma K-Nearest Neighbor Dan Fitur Ekstraksi N-Gram Dalam Analisis Sentimen Berbasis Aspek," vol. 12, no. 1, 2023, [Online]. Available: <https://www.kaggle.com/datasets/hafidahmusthaanah/skincare-review?select=00.+Review.csv>.