

**KOMPARASI ALGORITMA NAIVE BAYES DAN LOGISTIC REGRESSION TERHADAP ANALISIS SENTIMEN PERGANTIAN PELATIH TIMNAS INDONESIA DI MEDIA SOSIAL X (TWITTER)**Bustami Nur Qalbi <sup>1</sup>, Herry Sujaini <sup>2</sup>, Rina Septiriana <sup>3</sup>

Fakultas Teknik, Universitas Tanjungpura

email: [tamisidas45@gmail.com](mailto:tamisidas45@gmail.com)**Abstract**

The appointment of Patrick Kluivert as the new head coach of the Indonesian National Football Team in early 2025 sparked diverse public reactions, widely expressed on the social media platform X (formerly Twitter). This study aims to compare the performance of Naive Bayes and Logistic Regression algorithms in analyzing public sentiment toward the coaching change on X from January 6 to March 6, 2025. A dataset of 4,847 Indonesian-language tweets was collected through crawling using Google Colab and labeled for sentiment (positive, negative, neutral) with the assistance of four AI platforms: ChatGPT, Copilot, Grok, and Blackbox AI. The text preprocessing steps included case folding, cleaning, normalization, tokenizing, stopword removal, and stemming, followed by feature extraction using TF-IDF. To address class imbalance, the SMOTE technique was applied. Eight testing scenarios were conducted based on data split ratios (70:30 and 80:20) with and without SMOTE, with performance evaluated using accuracy, precision, recall, and F1-score metrics. The results show that Logistic Regression with SMOTE achieved the best performance, with an accuracy of 83.80%, precision of 82.80%, recall of 82.70%, and F1-score of 82.80%. Sentiment analysis revealed a predominance of negative sentiment, reflecting public criticism of the coaching change. This study provides insights into public perception and demonstrates that Logistic Regression outperforms Naive Bayes in the context of social media sentiment analysis.

**Article History**

Submitted: 23 April 2026

Accepted: 26 April 2026

Published: 27 April 2026

**Key Words**

Sentiment Analysis, Patrick Kluivert, Twitter/X, Naive Bayes, Logistic Regression, SMOTE

**Abstrak**

Perubahan pelatih Timnas Indonesia pada awal 2025, khususnya penunjukan Patrick Kluivert, memicu beragam respons dari masyarakat yang disuarakan secara masif melalui media sosial X (sebelumnya Twitter). Penelitian ini bertujuan untuk membandingkan performa algoritma Naive Bayes dan Logistic Regression dalam menganalisis sentimen publik terhadap pergantian pelatih Timnas Indonesia, Patrick Kluivert, pada media sosial X selama periode 6 Januari hingga 6 Maret 2025. Data sebanyak 4.847 tweet berbahasa Indonesia dikumpulkan melalui teknik crawling menggunakan Google Colab dan diberi label sentimen (positif, negatif, netral) dengan bantuan empat platform AI: ChatGPT, Copilot, Grok, dan Blackbox AI. Proses text-preprocessing meliputi case folding, cleaning, normalization, tokenizing, stopword removal, dan stemming, diikuti ekstraksi fitur menggunakan TF-IDF. Untuk mengatasi ketidakseimbangan kelas, teknik SMOTE diterapkan. Pengujian dilakukan dengan delapan skenario berdasarkan variasi rasio data (70:30 dan 80:20) dan penggunaan SMOTE, dengan evaluasi performa menggunakan matriks akurasi, precision, recall, dan F1-score. Hasil menunjukkan bahwa Logistic Regression dengan SMOTE menghasilkan performa terbaik dengan akurasi 83,80%, precision 82,80%, recall 82,70%, dan F1-score 82,80%. Analisis sentimen mengindikasikan dominasi sentimen negatif, mencerminkan kritik masyarakat terhadap pergantian pelatih. Penelitian ini memberikan wawasan tentang persepsi publik dan menunjukkan bahwa Logistic Regression lebih unggul dibandingkan Naive Bayes dalam konteks analisis sentimen berbasis data media sosial.

**Sejarah Artikel**

Submitted: 23 April 2026

Accepted: 26 April 2026

Published: 27 April 2026

**Kata Kunci**

Analisis Sentimen, Pergantian Pelatih, Patrick Kluivert, Twitter/X, Naive Bayes, Logistic Regression, SMOTE

## 1. PENDAHULUAN

Sepak bola merupakan olahraga paling populer di dunia yang tidak hanya berperan sebagai ajang kompetisi, tetapi juga sebagai sarana pemersatu lintas budaya, usia, dan latar belakang sosial[1]. Di Indonesia, sepak bola memiliki tempat istimewa di hati masyarakat, di mana Tim Nasional (Timnas) Indonesia menjadi simbol kebanggaan nasional dengan dukungan luar biasa dari para penggemar di seluruh penjuru negeri. Keberhasilan Timnas Indonesia lolos ke putaran ketiga Kualifikasi Piala Dunia 2026 menandai tonggak penting dalam perkembangan sepak bola nasional dan menunjukkan peningkatan performa tim yang signifikan. Namun, di tengah euforia tersebut, Persatuan Sepak Bola Seluruh Indonesia (PSSI) membuat keputusan kontroversial pada Januari 2025 dengan mengganti pelatih kepala Timnas Indonesia, meskipun pelatih sebelumnya dinilai telah membawa dampak positif dan konsisten sejak 2020. Keputusan ini menimbulkan beragam tanggapan dari publik, baik berupa kritik maupun dukungan, yang disampaikan secara masif melalui media sosial, khususnya platform X (sebelumnya *Twitter*).

Fenomena ini menunjukkan bagaimana keputusan strategis dalam dunia olahraga, khususnya yang melibatkan Timnas Indonesia, tidak hanya berdampak secara teknis, tetapi juga memicu dinamika opini publik. Media sosial menjadi medium utama bagi masyarakat untuk mengekspresikan pandangan mereka, sehingga menghasilkan beragam sentimen yang mencerminkan penerimaan maupun penolakan terhadap kebijakan PSSI.

Dalam konteks ini, analisis sentimen menjadi pendekatan yang relevan untuk memahami persepsi masyarakat terhadap pergantian pelatih Timnas Indonesia. Dengan menerapkan teknik klasifikasi, opini publik yang terekam dalam media sosial dapat dikelompokkan ke dalam beberapa kategori sentimen, seperti positif, negatif, dan netral. Klasifikasi ini memungkinkan peneliti untuk mengidentifikasi kecenderungan umum respon masyarakat—apakah mendukung atau menolak terhadap kebijakan tersebut. Beragam algoritma klasifikasi dapat diterapkan dalam analisis sentimen, di antaranya adalah *Naive Bayes* dan *Logistic Regression*. *Naive Bayes* merupakan salah satu algoritma dalam pembelajaran mesin yang berlandaskan pada prinsip *Teorema Bayes*, dengan asumsi dasar bahwa setiap atribut atau fitur pada data bersifat independen secara kondisional terhadap kelas target[2], sedangkan *Logistic Regression* adalah metode klasifikasi yang digunakan untuk membangun model hubungan antara variabel independen dan variabel dependen biner, dengan cara memetakan *input* berupa variabel diskrit maupun kontinu ke dalam probabilitas *output* yang terbagi ke dalam dua kelas, seperti positif atau negatif[3].

Data yang diperoleh dalam penelitian ini memiliki distribusi kelas yang tidak seimbang, sehingga metode SMOTE (*Synthetic Minority Over-sampling Technique*) digunakan untuk menyeimbangkan data tersebut. SMOTE merupakan teknik yang digunakan untuk menyeimbangkan distribusi data dengan cara menghasilkan sampel sintesis pada kelas minoritas hingga jumlahnya setara dengan kelas mayoritas[4]. *Confusion Matrix* adalah metode evaluasi kinerja model klasifikasi yang digunakan untuk mengidentifikasi ketepatan prediksi terhadap data. Matriks ini menyajikan perbandingan antara label hasil prediksi dan label aktual, yang mencerminkan sejauh mana model berhasil mengklasifikasikan data dengan benar atau keliru[5].

Penelitian terkait analisis sentimen di media sosial telah banyak dilakukan. Sebagai contoh, penelitian yang menganalisis sentimen masyarakat Indonesia terhadap perkembangan teknologi *metaverse* di platform X menggunakan *Naive Bayes* dan *Logistic Regression* menemukan bahwa *Logistic Regression* dengan optimasi SMOTE menghasilkan akurasi tertinggi (95%) dibandingkan *Naive Bayes* (91%), dengan peningkatan signifikan pada *precision*, *recall*, dan *F1-score* untuk sentimen negatif setelah penerapan SMOTE[3]. Temuan ini menunjukkan bahwa *Logistic Regression* lebih unggul dalam menangani data tidak seimbang, yang relevan dengan konteks penelitian ini, mengingat sentimen negatif terhadap

pergantian pelatih cenderung lebih sedikit dibandingkan sentimen positif atau netral. Wawasan dari penelitian tersebut memperkuat pentingnya komparasi algoritma dan optimasi data untuk memastikan hasil analisis sentimen yang akurat dan representatif.

Selain itu, penelitian yang membandingkan kinerja *Multinomial Naive Bayes* dan *Logistic Regression* dalam analisis sentimen ulasan film pada *dataset* IMDB, yang terdiri dari 40.000 ulasan pengguna, menggunakan teknik pra-pemrosesan seperti *case folding*, *tokenizing*, *stemming*, *stopwords removal*, dan TF-IDF untuk mempersiapkan data teks, diikuti dengan klasifikasi menggunakan kedua algoritma tersebut. Hasilnya menunjukkan bahwa *Multinomial Naive Bayes* mencapai akurasi sebesar 85%, sedikit lebih rendah dibandingkan *Logistic Regression*[6]. Penelitian ini menegaskan bahwa *Multinomial Naive Bayes* sangat efektif untuk klasifikasi teks dengan data berdimensi tinggi, seperti ulasan film, yang memiliki kemiripan dengan data teks pendek di media sosial X. Temuan ini relevan untuk penelitian ini karena menunjukkan potensi *Multinomial Naive Bayes* dalam menangani data teks singkat seperti cuitan, sekaligus memperkuat perlunya komparasi dengan *Logistic Regression* untuk menentukan algoritma yang optimal.

Berdasarkan pendekatan serupa, penelitian ini bertujuan untuk mengetahui hasil analisis sentimen yang dapat memberikan wawasan bagi pengambil kebijakan di federasi sepak bola, pelatih, dan pemangku kepentingan lainnya untuk memahami penerimaan publik terhadap keputusan strategis. Selain itu, komparasi performa algoritma *Naive Bayes* dan *Logistic Regression* akan memberikan gambaran tentang algoritma mana yang lebih optimal dalam konteks analisis sentimen berbasis data media sosial. Dengan demikian, penelitian ini tidak hanya berkontribusi pada pengembangan metode analisis sentimen tetapi juga pada pengambilan kebijakan yang lebih tepat di dunia olahraga Indonesia.

## 2. METODE PENELITIAN

Dalam penelitian ini, data yang digunakan merupakan kumpulan *tweet* dari pengguna X (*Twitter*) yang memuat kata kunci terkait Patrick Kluivert pada tanggal *upload* mulai dari tanggal 6 Januari 2025 sampai 6 Maret 2025. Kata kunci tersebut mengindikasikan bahwa pengguna sedang mendiskusikan topik terkait pergantian pelatih Timnas Indonesia pada fase kualifikasi piala dunia 2026. Dengan memanfaatkan teknik *machine learning*, data tersebut akan dianalisis untuk mengevaluasi opini pengguna X (*Twitter*) terhadap pergantian pelatih Timnas Indonesia.

## 3. Hasil dan pembahasan

### Pengumpulan Data

Pada tahap ini, data dikumpulkan untuk mendukung analisis sentimen terhadap opini publik mengenai pergantian pelatih Timnas Indonesia pada fase kualifikasi piala dunia 2026. Data dikumpulkan dari *platform* media sosial X (*Twitter*) dengan menggunakan metode *crawling*.

### Metode Pengumpulan Data

Pengambilan data dilakukan secara otomatis menggunakan teknik web *crawling* yang memanfaatkan *Application Programming Interface* (API) dari *Twitter*. Proses ini bertujuan untuk mengumpulkan *tweet* yang relevan berdasarkan kata kunci tertentu dan rentang waktu yang telah ditentukan. Kata kunci yang digunakan dalam proses *crawling* adalah "Patrick Kluivert".

Pengumpulan data dilakukan dalam rentang waktu 6 Januari 2025 hingga 6 Maret 2025. Periode ini dipilih karena mencakup momen-momen signifikan yang berkaitan dengan aktivitas Patrick Kluivert, terutama terkait pergantian posisi pelatih Timnas Indonesia dari Shin Tae-yong kepada Patrick Kluivert. Diharapkan, pada periode tersebut, akan ditemukan berbagai opini publik yang dapat mencerminkan persepsi masyarakat terhadap keputusan PSSI tersebut.

Dalam proses *crawling* ini, peneliti menggunakan alat bantu bernama *Tweet-Harvest*, yaitu sebuah *library open-source* yang dapat digunakan untuk mengumpulkan *tweet* berdasarkan kata kunci tertentu, dengan jumlah *tweet* yang dapat disesuaikan sesuai keinginan pengguna[21]. *Tweet-Harvest* memungkinkan pengguna untuk mengekstraksi data *tweet* secara sistematis berdasarkan kata kunci, rentang waktu, serta parameter pencarian lainnya. *Tool* ini dipilih karena kemudahannya dalam konfigurasi, keandalannya dalam pengambilan data dalam jumlah besar, serta kompatibilitasnya dengan format penyimpanan data yang umum digunakan dalam proses analisis lebih lanjut.

Proses *crawling* berhasil mengumpulkan sebanyak 8.908 *tweet* yang mengandung kata kunci “Patrick Kluivert” dalam periode yang telah ditentukan serta telah diperiksa dan dipastikan tidak ada duplikasi pada *dataset*. Data tersebut mencakup berbagai bentuk opini, seperti dukungan, kritik, harapan, serta komentar netral dari para pengguna *Twitter*.

### Kode Program 3.1 *Crawling*

```

1 # Crawl Data
2 filename = 'patrick2025032022.csv'
3 search_keyword = 'Patrick Kluivert since:2025-03-20 until:2025-03-22 lang:id'
4 limit = 520

5 !npx -y tweet-harvest@2.6.1 -o "{filename}" -s "{search_keyword}" --tab "LATEST" -l
   {limit} --token {twitter_auth_token}

```

### Labeling

Setelah data dikumpulkan dan diproses, tahap selanjutnya adalah *labeling* atau pelabelan sentimen terhadap masing-masing *tweet*. Tujuan dari proses ini adalah untuk mengklasifikasikan setiap *tweet* ke dalam kategori sentimen tertentu, yaitu positif, negatif, atau netral.

Dari total 8.908 *tweet* yang telah dianalisis dan diberi label menggunakan empat model AI, ditemukan bahwa 4.847 *tweet* memiliki kesesuaian label secara konsisten di keempat AI tersebut. Sementara itu, 4.061 *tweet* memiliki setidaknya satu perbedaan label di antara keempat AI tersebut. Hanya *tweet* yang konsisten yaitu sebanyak 4.847 *tweet* yang kemudian digunakan sebagai dasar untuk analisis sentimen, demi menjaga keakuratan hasil. Pada Gambar 3.1 menampilkan proses pelabelan dengan bantuan salah satu AI, yaitu *ChatGPT*.

No.	Sentimen	Penggalan Tweet	Alasan
951	Netral	Isbar jadi asisten manager nya Patrick Kluivert.	Hanya menyampaikan informasi lowongan, tanpa opini.
952	Positif	Erick Thohir: Patrick Kluivert dan Ajaran Pelatihnya Paket Lengkap.	Pengakuan terhadap kualitas dan kelengkapan staf pelatih.
953	Netral	Isdra Sjafri Kirim Video Timnas U-20 kepada Patrick Kluivert.	Informasi kegiatan, tidak memuat penilaian emosional.
954	Netral	Kurniawan Akai sudah Bertemu Patrick Kluivert.	Hanya menyampaikan fakta.
955	Positif	Isi siap menerima? *tutan apapun	Menunjukkan sikap profesional dan menghargai proses.

Proses dari pelabelan sentimen dengan menggunakan bantuan empat AI dapat dilihat pada Tabel 3.1

**Tabel 3.1** Pelabelan Sentimen Dengan Bantuan Empat AI

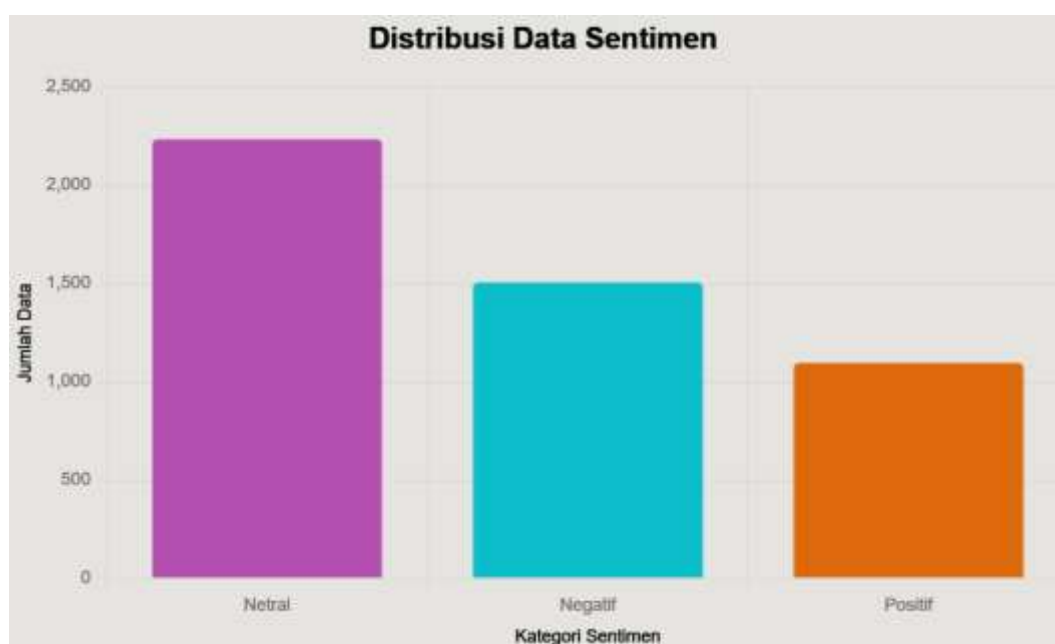
full_text	Sentimen	kosong	ChatGPT	Copilot	Grok	Blackbox
Rafael Struick yang merupakan striker Timnas Indonesia dan Brisbane Roar mengaku ingin belajar banyak dengan Patrick Kluivert agar menjadi pemain yang tajam di lini depan! Source : @omroepwestsport <a href="https://t.co/8IYqbhSwAN">https://t.co/8IYqbhSwAN</a>	Positif		Positif	Positif	Positif	Positif
Patrick Kluivert juga mantan striker dan dia juga pernah bermain untuk beberapa klub besar. Memang benar (berharap belajar banyak dari Kluivert) saya juga berasumsi demikian. <a href="https://t.co/OnfssNuEXp">https://t.co/OnfssNuEXp</a>	-		Netral	Positif	Positif	Netral
RAFAEL STRUICK KAGUMI SOSOK PATRICK KLUIVERT! Merupakan suatu kehormatan untuk saya bisa dilatih Patrick Kluivert. Saya tidak sabar untuk bekerja sama dengannya. <a href="https://t.co/FyvIKvLA29">https://t.co/FyvIKvLA29</a>	Positif		Positif	Positif	Positif	Positif
Ambil Sumpah WNI 10 Maret 2025 Ini Daftar 27 Pemain Timnas Indonesia yang Dipanggil Patrick Kluivert Lawan Australia dan Bahrain? <a href="https://t.co/MuGH7Q8DVx">https://t.co/MuGH7Q8DVx</a> lewat @okezonenews	Netral		Netral	Netral	Netral	Netral
Pemain Ini Dicoret Patrick Kluivert meski Jadi Starting XI di Laga Timnas Indonesia vs Jepang? <a href="https://t.co/k6NCGVceYi">https://t.co/k6NCGVceYi</a>	Negatif		Negatif	Negatif	Negatif	Negatif
6 Pemain Debutan yang Berpotensi Dipanggil Patrick Kluivert ke Timnas	-		Netral	Positif	Positif	Netral

Indonesia <a href="https://t.co/YxVRkVGK4J">https://t.co/YxVRkVGK4J</a>						
Ragnar Oratmangoen Tak Bisa Main Lawan Australia Egy Maulana Vikri Jadi Pilihan Patrick Kluivert di Lini Depan Timnas Indonesia <a href="https://t.co/XH0kLdBIIG">https://t.co/XH0kLdBIIG</a>	-	Netral	Netral	Netral	Negatif	

Kategori	Jumlah
Jumlah Sentimen Valid	4847
Jumlah '-' (tidak valid)	4061

**Gambar 3.2** Jumlah Validasi Sentimen

Tabel 3.1 menampilkan kolom 'Sentimen' dan kolom empat AI yaitu 'ChatGPT', 'Copilot', 'Grok', dan 'Blackbox' yang menunjukkan hasil pelabelan dari keempat AI, serta terdapat kolom 'kosong' untuk memisahkan kolom 'Sentimen' dan kolom empat AI. Jika keempat AI memberikan label sentimen yang sama, maka label tersebut akan dituliskan pada kolom tersebut. Namun, jika terdapat perbedaan meskipun hanya dari satu AI, maka kolom tersebut akan diisi dengan tanda '-' yang berarti tidak valid. Pada Gambar 3.2 dari total data yang dianalisis, jumlah data dengan sentimen valid berjumlah 4.847, sedangkan data dengan sentimen tidak valid ('-') sebanyak 4.061, sehingga data yang digunakan untuk analisis sentimen adalah 4.847 data.



**Gambar 3.3** Distribusi Label Sentimen

Dataset yang dikumpulkan melalui proses *crawling tweet* berjumlah 4.847 data, masing-masing telah diberi label sentimen untuk analisis distribusi emosi dalam konten media sosial. Berdasarkan visualisasi histogram pada Gambar 3.3 sentimen netral merupakan kategori dominan dengan 2.238 *tweet* (46,2% dari total data), mengindikasikan sebaran konten yang bersifat informatif atau tidak memihak. Sentimen negatif menyusul dengan 1.509 *tweet* (31,1%), mencerminkan adanya ekspresi ketidakpuasan atau kritik yang signifikan. Sebaliknya,

sentimen positif memiliki representasi terkecil, yakni 1.100 *tweet* (22,7%), menunjukkan frekuensi yang lebih rendah untuk konten yang menyampaikan kepuasan atau dukungan. Distribusi ini memberikan wawasan awal mengenai pola sentimen dalam *dataset*, yang dapat menjadi dasar untuk analisis lebih lanjut terkait dinamika opini publik di *platform* media sosial.

### Text-Preprocessing

Data yang diperoleh melalui proses *crawling* dari *API Twitter* selanjutnya melalui tahapan *text-preprocessing* untuk membersihkan dan menormalkan teks. Tahapan ini bertujuan untuk mempermudah sistem dalam mengenali istilah-istilah penting (*terms*) pada setiap kalimat, sehingga data yang dihasilkan menjadi lebih siap digunakan dalam proses pelatihan dan pengujian model. Berikut ini hasil dari tahapan *text-preprocessing*:

#### a. Case folding

Proses ini menghasilkan perubahan seluruh karakter teks menjadi huruf kecil, tanpa membedakan antara huruf besar dan kecil sebelumnya. Pada Tabel 3.2 dapat dilihat kalimat sebelum dan sesudah proses *case folding*.

### Kode Program 3.2 Case Folding

```
1 def case_folding(text):
2     return text.lower()
```

Tabel 3.2 merupakan contoh dari proses *Case Folding*.

Tabel 3.2 Contoh Case Folding

Proses	Teks
Sebelum	Rafael Struick yang merupakan striker Timnas Indonesia dan Brisbane Roar mengaku ingin belajar banyak dengan Patrick Kluivert agar menjadi pemain yang tajam di lini depan! Source : @omroepwestsport <a href="https://t.co/8IYqbhSwAN">https://t.co/8IYqbhSwAN</a>
Sesudah	rafael struick yang merupakan striker timnas indonesia dan brisbane roar mengaku ingin belajar banyak dengan patrick kluivert agar menjadi pemain yang tajam di lini depan! source : @omroepwestsport <a href="https://t.co/8iyqbhswan">https://t.co/8iyqbhswan</a>

#### b. Cleaning

Proses ini bertujuan untuk membersihkan data dengan menghapus tanda baca, tab, URL, angka, serta kata-kata yang dianggap tidak relevan. Tabel 3.3 menunjukkan perbandingan antara kalimat asli hasil *crawling* dan kalimat yang telah melalui tahap *cleaning*, termasuk proses *case folding* (mengubah semua huruf menjadi huruf kecil). Pada tahap pembersihan ini, elemen-elemen seperti URL dan frasa yang tidak dibutuhkan telah dihilangkan dari teks.

### Kode Program 3.3 Cleaning

```
1 def cleaning(text):
2     text = re.sub(r"http\S+|www\S+|https\S+", "", text, flags=re.MULTILINE)
3     text = re.sub(r'@\w+', "", text)
4     text = re.sub(r'#\w+', "", text)
5     text = re.sub(r'd+', "", text)
6     text = text.translate(str.maketrans("", "", string.punctuation))
7     text = re.sub(r'^[\x00-\x7F]+', "", text)
8     text = text.strip()
9     return text
```

Tabel 3.3 merupakan contoh dari proses *Cleaning*.

**Tabel 3.3** Contoh *Cleaning*

Proses	Teks
Sebelum	rafael struick yang merupakan striker timnas indonesia dan brisbane roar mengaku ingin belajar banyak dengan patrick kluivert agar menjadi pemain yang tajam di lini depan! source : @omroepwestsport <a href="https://t.co/8iyqbhswan">https://t.co/8iyqbhswan</a>
Sesudah	rafael struick yang merupakan striker timnas indonesia dan brisbane roar mengaku ingin belajar banyak dengan patrick kluivert agar menjadi pemain yang tajam di lini depan source

c. *Normalize*

Proses ini menyetarakan kata-kata tidak baku, slang, singkatan, atau variasi penulisan menjadi bentuk kata yang standar dan sesuai kaidah bahasa, seperti "gk" untuk "tidak", "bgt" untuk "banget", atau "skrg" untuk "sekarang" dan lain-lain. Pada Tabel 3.4 memperlihatkan hasil dari proses normalisasi.

**Kode Program 3.4** *Normalize*

```

1 # Fungsi normalisasi
2 def normalize_text_only(text):
3     tokens = str(text).split()
4     normalized_tokens = [normalization_dict.get(token, token) for token in tokens]
5     return ' '.join(normalized_tokens)

```

Tabel 3.4 merupakan contoh hasil proses *normalize*.

**Tabel 3.4** Contoh *Normalize*

Proses	Teks
Sebelum	Nggak sabar pengen liat racikan Patrick Kluivert
Sesudah	tidak sabar ingin lihat racikan patrick kluivert

d. *Tokenizing*

Setelah melalui tahap pembersihan dan normalisasi, data kemudian diproses lebih lanjut dengan *tokenizing*. Proses ini memisahkan kalimat menjadi unit-unit kata atau token, sehingga setiap kata dapat dianalisis secara individual. Tabel 3.5 memperlihatkan hasil tokenisasi dari kalimat yang telah melalui tahap *cleaning* dan *case folding*. Pada tahap ini, kalimat dipecah menjadi serangkaian kata terpisah yang siap digunakan dalam proses analisis selanjutnya, seperti pembobotan *term* atau pelatihan model.

**Kode Program 3.5** *Tokenizing*

```

1 def tokenize(text):
2     return word_tokenize(text)

```

Tabel 3.5 merupakan contoh hasil proses *tokenizing*.

**Tabel 3.5** Contoh *Tokenizing*

Proses	Teks
Sebelum	rafael struick yang merupakan striker timnas indonesia dan brisbane roar mengaku ingin belajar banyak dengan patrick kluivert agar menjadi pemain yang tajam di lini depan source
Sesudah	['rafael', 'struick', 'yang', 'merupakan', 'striker', 'timnas', 'indonesia', 'dan', 'brisbane', 'roar', 'mengaku', 'ingin', 'belajar', 'banyak', 'dengan', 'patrick', 'kluivert', 'agar', 'menjadi', 'pemain', 'yang', 'tajam', 'di', 'lini', 'depan', 'source']

e. *Stopword Removal*

Setelah proses tokenisasi, tahap selanjutnya adalah *stopword removal*, yaitu menghapus kata-kata umum yang tidak memiliki makna signifikan dalam analisis teks, seperti 'dan', 'yang', 'di', dan sebagainya. Kata-kata ini umumnya tidak memberikan kontribusi penting terhadap pemahaman konteks atau klasifikasi teks. Tabel 3.6 menampilkan hasil teks sebelum dan sesudah dilakukan *stopword removal*. Pada tahap ini, hanya kata-kata yang dianggap bermakna atau relevan yang dipertahankan untuk keperluan analisis lebih lanjut.

**Kode Program 3.6** *Stopword Removal*

```

1 def remove_stopwords(tokens):
2     return [token for token in tokens if token not in stopwords_list]

```

Tabel 3.6 merupakan contoh hasil proses *stopword removal*.

**Tabel 3.6** Contoh *Stopword Removal*

Proses	Teks
Sebelum	rafael struick yang merupakan striker timnas indonesia dan brisbane roar mengaku ingin belajar banyak dengan patrick kluivert agar menjadi pemain yang tajam di lini depan source
Sesudah	rafael struick merupakan striker timnas indonesia brisbane roar mengaku belajar banyak patrick kluivert menjadi pemain tajam lini depan source

f. *Stemming*

Proses ini mengembalikan setiap kata ke bentuk dasarnya (*root word*). Tujuannya adalah untuk menyamakan variasi kata yang memiliki makna serupa, seperti 'bermain', 'bermainlah', dan 'memainkan', menjadi bentuk dasar 'main'. Hal ini membantu mengurangi jumlah fitur yang harus dianalisis dan meningkatkan konsistensi data. Tabel 3.7 menunjukkan perbandingan antara kata-kata sebelum dan sesudah dilakukan *stemming*. Pada tahap ini, kata-kata yang telah melalui proses *tokenizing* dan *stopword removal* diubah ke bentuk dasarnya agar lebih efisien dalam proses analisis selanjutnya.

**Kode Program 3.7** *Stemming*

```

1 def stemming(tokens):
2     return [stemmer.stem(token) for token in tokens]

```

Tabel 3.7 merupakan contoh dari proses *stemming*.

**Tabel 3.7** Contoh *Stemming*

Proses	Teks
Sebelum	rafael struick merupakan striker timnas indonesia brisbane roar mengaku belajar banyak patrick kluyvert menjadi pemain tajam lini depan source
Sesudah	rafael struick rupa striker timnas indonesia brisbane roar aku ajar banyak patrick kluyvert jadi main tajam lini depan source

### Hasil Ekstraksi Fitur

Setelah dilakukan proses *text-preprocessing* terhadap data *tweet*, tahap selanjutnya adalah ekstraksi fitur menggunakan metode TF-IDF (*Term Frequency–Inverse Document Frequency*). TF-IDF digunakan untuk mengonversi teks mentah menjadi representasi numerik dengan memberikan bobot pada setiap kata berdasarkan frekuensinya di dalam dokumen dan keberadaannya di seluruh korpus.

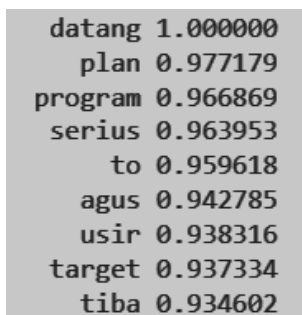
Hasil ekstraksi fitur menunjukkan bahwa terdapat beberapa kata dengan nilai TF-IDF tinggi, yang menunjukkan pentingnya kata tersebut dalam konteks masing-masing dokumen. Pada Gambar 3.4 menampilkan hasil ekstraksi fitur dengan TF-IDF yang membantu dalam mengidentifikasi kata-kata kunci yang berkontribusi signifikan terhadap isi dokumen dan dapat digunakan secara efektif dalam tahap klasifikasi sentimen.

### Kode Program 3.8 TF-IDF

```

1 from sklearn.feature_extraction.text import TfidfVectorizer
2 # Ekstraksi fitur dengan TF-IDF
3 vectorizer = TfidfVectorizer(min_df=1, max_df=0.8)
4 X_train_tfidf = vectorizer.fit_transform(X_train)
5 X_test_tfidf = vectorizer.transform(X_test)

```



**Gambar 3.4** Hasil TF-IDF

### Hasil SMOTE

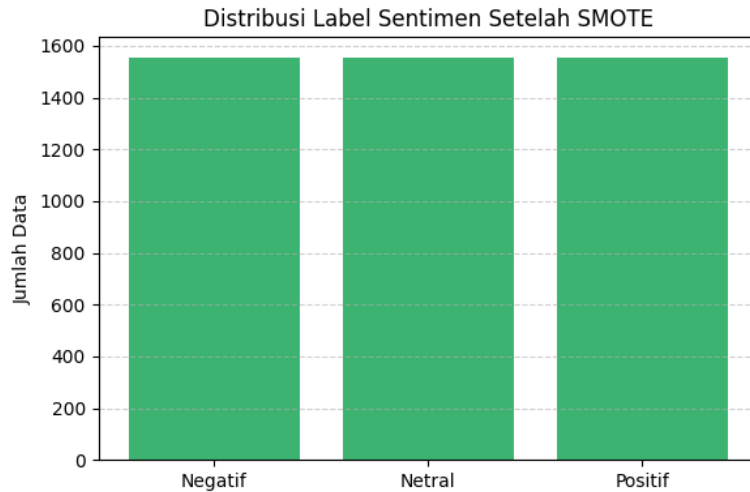
Analisis distribusi label sentimen setelah penerapan teknik SMOTE menunjukkan peningkatan keseimbangan pada dataset. Sebelum SMOTE, distribusi label cenderung tidak seimbang dengan jumlah data yang signifikan pada salah satu kelas. Pada Gambar 3.5 dan Gambar 3.6 Setelah proses SMOTE, jumlah data untuk setiap kelas (Negatif, Netral, dan Positif) menjadi lebih merata.

### Kode Program 3.9 SMOTE

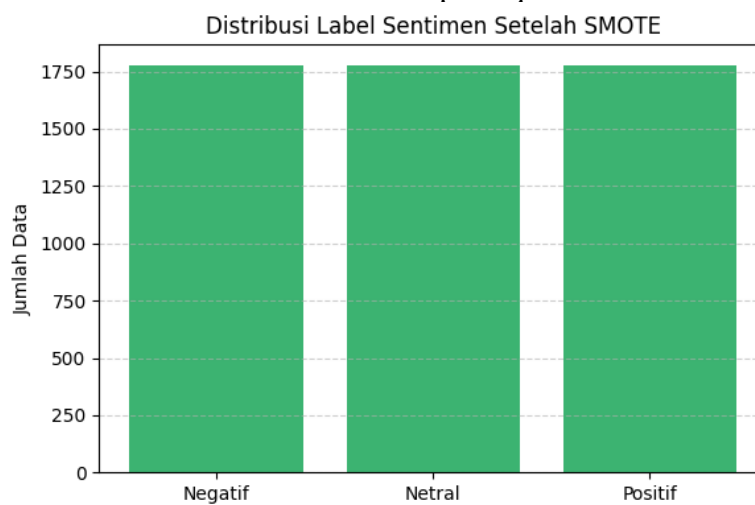
```

1 from imblearn.over_sampling import SMOTE
2 # SMOTE
3 smote = SMOTE(random_state=42)
4 X_resampled, y_resampled = smote.fit_resample(X_train_tfidf, y_train)

```



**Gambar 3.5** Hasil SMOTE pada *split* data 70:30



**Gambar 3.6** Hasil SMOTE pada *split* data 80:20

Berdasarkan visualisasi pada grafik distribusi label sentimen setelah SMOTE, SMOTE berhasil mengatasi ketidakseimbangan data dengan melakukan oversampling pada kelas minoritas, sehingga model *machine learning* yang digunakan dapat belajar lebih baik dari data yang lebih seimbang.

### Implementasi Algoritma

#### *Naive Bayes*

#### Kode Program 3.10 *Naive Bayes*

```
1 # Bagi data menjadi data latih dan uji
2 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

3 # Model Naive Bayes
4 model = MultinomialNB()
5 model.fit(X_train_tfidf, y_train)

6 # Prediksi dan evaluasi
7 y_pred = model.predict(X_test_tfidf)

8 # Tampilkan hasil
9 print("Akurasi:", accuracy_score(y_test, y_pred))
```

```

10 print("Laporan Klasifikasi:")
11 print(classification_report(y_test, y_pred))

```

Setelah data melewati berbagai tahapan hingga siap digunakan, langkah selanjutnya adalah mengimplementasikan algoritma. Pada tahap ini digunakan algoritma *Naive Bayes*. Sebelum implementasi dilakukan, data yang telah dipersiapkan, baik yang telah melalui proses SMOTE maupun yang tidak, dibagi terlebih dahulu menjadi data *training* dan data *testing*. Dalam penelitian ini, pembagian data dilakukan dengan rasio 70:30 dan 80:20. Selanjutnya, dilakukan pembuatan model *Naive Bayes*, pelatihan model menggunakan data latih, serta evaluasi terhadap model yang mencakup perhitungan akurasi dari algoritma *Naive Bayes*.

### Logistic Regression

#### Kode Program 3.11 Logistic Regression

```

1 # Bagi data menjadi data latih dan uji
2 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

3 from sklearn.linear_model import LogisticRegression
4 # Model Logistic Regression
5 model = LogisticRegression(max_iter=200)
6 model.fit(X_train_tfidf, y_train)

7 # Prediksi dan evaluasi
8 y_pred = model.predict(X_test_tfidf)

9 # Tampilkan hasil
10 print("Akurasi:", accuracy_score(y_test, y_pred))
11 print("Laporan Klasifikasi:")
12 print(classification_report(y_test, y_pred))

```

Setelah data melewati berbagai tahapan hingga siap digunakan, langkah selanjutnya adalah mengimplementasikan algoritma. Pada tahap ini digunakan algoritma *Logistic Regression* dengan maksimal 200 iterasi. Sebelum implementasi dilakukan, data yang telah dipersiapkan, baik yang telah melalui proses SMOTE maupun yang tidak, dibagi terlebih dahulu menjadi data *training* dan data *testing*. Dalam penelitian ini, pembagian data dilakukan dengan rasio 70:30 dan 80:20. Selanjutnya, dilakukan pembuatan model *Logistic Regression*, pelatihan model menggunakan data latih, serta evaluasi terhadap model yang mencakup perhitungan akurasi dari algoritma *Logistic Regression*.

### Pengujian dan Evaluasi

Dalam penelitian ini, pengujian dilakukan terhadap delapan skenario untuk mengevaluasi performa model klasifikasi menggunakan algoritma *Naive Bayes* dan *Logistic Regression*. Setiap algoritma diuji dengan rasio pembagian data 70:30 (3392 data latih, 1455 data uji) dan 80:20 (3877 data latih, 970 data uji), dengan dan tanpa penerapan teknik *Synthetic Minority Oversampling Technique* (SMOTE). Parameter evaluasi meliputi akurasi, *precision*, *recall*, dan *F1-score*.

### Skenario Pengujian

Berikut ini adalah penjelasan masing-masing dari delapan skenario pengujian:

- a. Skenario 1: *Naive Bayes*, 70:30, Tanpa SMOTE

**Tabel 3.8** Skenario 1 *Naive Bayes*, 70:30, Tanpa SMOTE.

Confusion matrix					
Data Uji	Prediksi	Akurasi	<i>Precision</i>	<i>Recall</i>	<i>f1-score</i>

		Negatif	Netral	Positif				
Label Manual	Negatif	342	104	13	0,802	0,841	0,756	0,781
	Netral	34	643	6				
	Positif	16	115	182				

Pada Tabel 3.8 menampilkan skenario 1 menggunakan algoritma *Naive Bayes* dengan pembagian data 70:30 (3392 data latih, 1455 data uji), skenario ini menghasilkan akurasi 80,2%. *Confusion matrix* menunjukkan prediksi yang baik untuk kelas netral (643 benar), namun kelas negatif dan positif memiliki kesalahan yang cukup signifikan (104 dan 115 salah klasifikasi). *Precision* tinggi 84,1% menunjukkan kemampuan model mendeteksi kelas positif dengan baik, tetapi *recall* rendah 75,6% mengindikasikan banyak kelas positif yang terlewat.

b. Skenario 2: *Naive Bayes*, 80:20, Tanpa SMOTE

**Tabel 3.9** Skenario 2 *Naive Bayes*, 80:20, Tanpa SMOTE

Confusion matrix								
Data Uji		Prediksi			Akurasi	<i>Precision</i>	<i>Recall</i>	<i>f1-score</i>
		Negatif	Netral	Positif				
Label Manual	Negatif	228	66	6	0,808	0,851	0,764	0,790
	Netral	25	431	4				
	Positif	6	79	125				

Skenario 2 menghasilkan akurasi meningkat menjadi 80,8%. Peningkatan data latih memperbaiki prediksi kelas netral (431 benar), tetapi kelas negatif dan positif masih menunjukkan kesalahan (66 dan 79 salah klasifikasi). *Precision* 85,1% tetap tinggi, namun *recall* 76,4% menunjukkan keterbatasan dalam mendeteksi semua kelas positif.

c. Skenario 3: *Naive Bayes*, 70:30, SMOTE

**Tabel 3.10** Skenario 3 *Naive Bayes*, 70:30, SMOTE

Confusion matrix								
Data Uji		Prediksi			Akurasi	<i>Precision</i>	<i>Recall</i>	<i>f1-score</i>
		Negatif	Netral	Positif				
Label Manual	Negatif	378	40	41	0,812	0,799	0,819	0,806
	Netral	69	540	74				
	Positif	11	38	264				

Dengan SMOTE pada rasio 70:30, akurasi meningkat menjadi 81,2%. SMOTE membantu menyeimbangkan kelas, meningkatkan *recall* 81,9% dan *F1-score* 80,6%, tetapi *precision* menurun 79,9% karena lebih banyak *false positive* (40 untuk negatif, 74 untuk netral). Prediksi kelas positif membaik (264 benar).

d. Skenario 4: *Naive Bayes*, 80:20, SMOTE

**Tabel 3.11** Skenario 4 *Naive Bayes*, 80:20, SMOTE

Confusion matrix								
Data Uji		Prediksi			Akurasi	<i>Precision</i>	<i>Recall</i>	<i>f1-score</i>
		Negatif	Netral	Positif				
Label Manual	Negatif	247	27	26	0,815	0,802	0,824	0,810
	Netral	48	364	48				
	Positif	5	25	180				

Pada Tabel 3.11 Kombinasi SMOTE dan rasio 80:20 menghasilkan akurasi tertinggi untuk *Naive Bayes* yaitu 81,5%. *Confusion matrix* menunjukkan peningkatan prediksi kelas positif (180 benar), dengan *recall* 82,4% dan *F1-score* 81% yang seimbang, meskipun *precision* tetap 80,2% karena *false positive* yang signifikan.

e. Skenario 5: *Logistic Regression*, 70:30, Tanpa SMOTE

**Tabel 3.12** Skenario 5 *Logistic Regression*, 70:30, Tanpa SMOTE

Confusion matrix								
Data Uji		Prediksi			Akurasi	Precision	Recall	f1-score
		Negatif	Netral	Positif				
Label Manual	Negatif	361	79	19	0,832	0,842	0,807	0,820
	Netral	39	625	19				
	Positif	18	70	225				

Pada Tabel 3.12 skenario menggunakan *Logistic Regression* dengan rasio 70:30, skenario ini mencapai akurasi 83,2%. *Confusion matrix* menunjukkan prediksi yang lebih seimbang untuk semua kelas (361 negatif, 625 netral, 225 positif benar), dengan *precision* 84,2%, *recall* 80,7%, dan *F1-score* 82% yang lebih baik dibandingkan *Naive Bayes*.

f. Skenario 6: *Logistic Regression*, 80:20, Tanpa SMOTE

**Tabel 3.13** Skenario 6 *Logistic Regression*, 80:20, Tanpa SMOTE

Confusion matrix								
Data Uji		Prediksi			Akurasi	Precision	Recall	f1-score
		Negatif	Netral	Positif				
Label Manual	Negatif	242	45	13	0,843	0,852	0,818	0,832
	Netral	23	424	13				
	Positif	8	50	152				

Skenario ini mencapai akurasi tertinggi keseluruhan 84,3%. Dengan rasio 80:20, model memanfaatkan lebih banyak data latih (3877), menghasilkan prediksi akurat untuk kelas negatif (242 benar), netral (424 benar), dan positif (152 benar). *Precision* 85,2%, *recall* 81,8%, dan *F1-score* 83,2% menunjukkan performa optimal.

g. Skenario 7: *Logistic Regression*, 70:30, SMOTE

**Tabel 3.14** Skenario 7 *Logistic Regression*, 70:30, SMOTE

Confusion matrix								
Data Uji		Prediksi			Akurasi	Precision	Recall	f1-score
		Negatif	Netral	Positif				
Label Manual	Negatif	382	52	25	0,839	0,830	0,826	0,828
	Netral	56	539	34				
	Positif	17	49	247				

Pada Tabel 3.14 skenario dengan SMOTE pada rasio 70:30, akurasi sedikit menurun menjadi 83,9% dibandingkan Skenario 5. SMOTE meningkatkan *recall* 82,6% dan *F1-score* 82,8%, tetapi *precision* menurun 83% karena lebih banyak *false positive* (52 untuk negatif, 34 untuk netral). Prediksi kelas positif tetap kuat (247 benar).

h. Skenario 8: *Logistic Regression*, 80:20, SMOTE

**Tabel 3.15** Skenario 8 *Logistic Regression*, 80:20, SMOTE

Data Uji		Prediksi			Akurasi	Precision	Recall	f1-score
		Negatif	Netral	Positif				
Label Manual	Negatif	250	32	18	0,836	0,827	0,829	0,828
	Netral	38	393	29				
	Positif	8	34	168				

Pada Tabel 3.15 skenario dengan rasio 80:20 dan SMOTE, akurasi adalah 83,6% lebih rendah dari Skenario 6. *Confusion matrix* menunjukkan prediksi yang baik untuk kelas negatif (250 benar) dan netral (393 benar), tetapi kelas positif memiliki lebih banyak kesalahan (34 salah klasifikasi).

**Tabel 3.16** Hasil Pengujian Delapan Skenario

Skenario	Algoritma	Split Data	SMOTE	Akurasi	Precision	Recall	F1-score
1	Naive Bayes	70:30	No	0.802	0.841	0.756	0.781
2	Naive Bayes	80:20	No	0.808	0.851	0.764	0.790
3	Naive Bayes	70:30	Yes	0,812	0,799	0,819	0,806
4	Naive Bayes	80:20	Yes	0,815	0,802	0,824	0,810
5	Logistic Regression	70:30	No	0,832	0,842	0,807	0,820
6	Logistic Regression	80:20	No	0,843	0,852	0,818	0,832
7	Logistic Regression	70:30	Yes	0,839	0,830	0,826	0,828
8	Logistic Regression	80:20	Yes	0,836	0,827	0,829	0,828

Tabel 3.16 menampilkan hasil pengujian delapan skenario pengklasifikasian sentimen menggunakan algoritma *Naive Bayes* dan *Logistic Regression* dengan variasi split data (70:30 dan 80:20) serta penerapan SMOTE untuk menangani ketidakseimbangan kelas. *Logistic Regression* tanpa SMOTE pada split 80:20 mencatat performa terbaik dengan akurasi 84,3%, *precision* 85,2%, *recall* 81,8%, dan *F1-score* 83,2%. Penerapan SMOTE pada *Logistic Regression* meningkatkan *recall* hingga 82,9%, namun sedikit menurunkan *precision*. *Naive Bayes* dengan SMOTE pada split 80:20 menghasilkan akurasi 81,5% dan *recall* 80,2%, sementara tanpa SMOTE akurasinya lebih rendah, sekitar 80,8%. Secara keseluruhan, *Logistic Regression* lebih unggul, terutama pada split 80:20 tanpa SMOTE, sedangkan SMOTE efektif untuk meningkatkan deteksi kelas minoritas.

**Perbandingan Rata-Rata Skenario**

Berdasarkan hasil pengujian yang dilakukan terhadap delapan skenario pengklasifikasian sentimen di media sosial X (*Twitter*) terkait pelatih baru Timnas, Patrick Kluivert, analisis dilakukan dengan membandingkan performa rata-rata dari pasangan skenario yang menggunakan algoritma *Naive Bayes* dan *Logistic Regression* dengan variasi rasio pembagian

data (70:30 dan 80:20) serta penerapan teknik SMOTE. Evaluasi performa model dilakukan menggunakan metrik akurasi, *precision*, *recall*, dan *F1-Score*. Berikut adalah analisis berdasarkan rata-rata performa untuk pasangan skenario 1 dan 2, skenario 3 dan 4, skenario 5 dan 6, serta skenario 7 dan 8.

a. Rata-Rata Skenario 1 dan 2 (*Naive Bayes*, Tanpa SMOTE)

**Tabel 3.17** Rata-Rata Skenario 1 dan 2 (*Naive Bayes*, Tanpa SMOTE)

	Akurasi	<i>precision</i>	<i>recall</i>	<i>f1-score</i>
Rata-Rata	80,50%	84,60%	76%	78,50%

Pada Tabel 3.17 rata-rata performa menunjukkan bahwa model *Naive Bayes* tanpa SMOTE memiliki *precision* yang tinggi 84,60%, yang mengindikasikan kemampuan model untuk meminimalkan prediksi positif yang salah (*false positive*). Namun, *recall* yang relatif lebih rendah 76% menunjukkan bahwa model masih kesulitan mendeteksi semua kelas positif dengan benar, terutama pada kelas minoritas, yang kemungkinan disebabkan oleh ketidakseimbangan data.

b. Rata-Rata Skenario 3 dan 4 (*Naive Bayes*, SMOTE)

**Tabel 3.18** Rata-Rata Skenario 3 dan 4 (*Naive Bayes*, SMOTE)

	Akurasi	<i>precision</i>	<i>recall</i>	<i>f1-score</i>
Rata-Rata	81,30%	80%	82,10%	80%

Penerapan SMOTE meningkatkan *recall* sebesar 6% dibandingkan skenario tanpa SMOTE, yang menunjukkan bahwa teknik ini efektif dalam menangani ketidakseimbangan kelas dengan meningkatkan kemampuan model untuk mendeteksi kelas positif. Namun, *precision* sedikit menurun 80% dibandingkan skenario tanpa SMOTE, yang mengindikasikan adanya *trade-off* antara *recall* dan *precision* akibat penambahan data sintetis.

c. Rata-Rata Skenario 5 dan 6 (*Logistic Regression*, Tanpa SMOTE)

**Tabel 3.19** Rata-Rata Skenario 5 dan 6 (*Logistic Regression*, Tanpa SMOTE)

	Akurasi	<i>precision</i>	<i>recall</i>	<i>f1-score</i>
Rata-Rata	83,76%	84,50%	81,50%	82,50%

Pada Tabel 3.19 rata-rata performa dari kedua skenario ini menunjukkan hasil akurasi sebesar 83,76% dan keseimbangan yang baik antara *precision* dan *recall*. *Logistic Regression* tanpa SMOTE terbukti lebih unggul dibandingkan *Naive Bayes*, terutama pada rasio pembagian data 80:20, yang memberikan akurasi dan *F1-Score* tertinggi.

d. Rata-Rata Skenario 7 dan 8 (*Logistic Regression*, Dengan SMOTE)

**Tabel 3.20** Rata-Rata Skenario 7 dan 8 (*Logistic Regression*, SMOTE)

	Akurasi	<i>Precision</i>	<i>recall</i>	<i>f1-score</i>
Rata-Rata	83,80%	82,80%	82,70%	82,80%

Tabel 3.20 menunjukkan rata-rata performa skenario 7 dan 8 (*Logistic Regression* dengan SMOTE) sebagai yang terbaik, dengan akurasi 83,80%, *precision* 82,80%, *recall* 82,70%, dan *F1-score* 82,80%. SMOTE meningkatkan *recall* (82,70% vs. 81,50% pada skenario 5 dan 6), mengoptimalkan deteksi kelas minoritas, meskipun *precision* sedikit menurun (82,80% vs. 84,50%). Hal ini menunjukkan bahwa SMOTE membantu menyeimbangkan performa antar kelas, tetapi dengan sedikit penurunan pada *precision*.

### Hasil Analisis

#### Kecenderungan Sentimen Publik

Berdasarkan analisis sentimen terhadap pelatih baru Timnas, Patrick Kluivert, di media sosial X (*Twitter*) menggunakan delapan skenario pengklasifikasian dengan algoritma *Naive Bayes* dan *Logistic Regression*, hasil menunjukkan bahwa sentimen masyarakat menunjukkan dominasi sentimen negatif dibandingkan sentimen positif ketika sentimen netral diabaikan.

Sentimen netral diabaikan dalam analisis ini karena fokus penelitian tertuju pada sentimen yang memiliki polaritas jelas, yaitu sentimen positif dan negatif, untuk memahami kecenderungan opini masyarakat terhadap pelatih baru Timnas, Patrick Kluivert, secara lebih tegas. Sentimen netral, yang mencerminkan sikap atau opini yang tidak condong ke arah positif maupun negatif, sering kali kurang memberikan wawasan spesifik tentang preferensi atau penolakan masyarakat terhadap subjek tertentu. Dalam konteks ini, sentimen netral seperti komentar yang bersifat informatif atau tidak menunjukkan emosi kuat tidak dianggap relevan untuk menilai penerimaan atau penolakan terhadap Patrick Kluivert. Dengan mengabaikan sentimen netral, analisis dapat lebih terfokus pada dinamika antara sentimen positif (dukungan atau pujian) dan negatif (kritik atau ketidakpuasan), yang memberikan gambaran lebih jelas tentang polarisasi opini publik. Selain itu, dari *confusion matrix* pada semua skenario, kelas netral (misalnya, 643 pada skenario 1, 431 pada skenario 2) memiliki jumlah yang sangat dominan dibandingkan positif dan negatif, yang dapat mengaburkan interpretasi kecenderungan sentimen jika disertakan. Oleh karena itu, mengabaikan sentimen netral memungkinkan evaluasi yang lebih tajam terhadap sentimen yang memiliki dampak emosional atau opini yang lebih signifikan.

Hal ini terlihat dari *confusion matrix* pada berbagai skenario, di mana jumlah prediksi benar untuk sentimen negatif secara konsisten lebih tinggi. Misalnya, pada skenario 1 (*Naive Bayes*, 70:30, tanpa SMOTE), terdeteksi 342 kelas negatif benar dibandingkan 182 kelas positif, dan pada skenario 6 (*Logistic Regression*, 80:20, tanpa SMOTE), terdeteksi 242 kelas negatif benar dibandingkan 152 kelas positif. Skenario lainnya, seperti skenario 3 dan 4 (*Naive Bayes*, dengan SMOTE) serta skenario 7 dan 8 (*Logistic Regression*, dengan SMOTE), juga menunjukkan pola serupa dengan jumlah kelas negatif yang lebih tinggi (378 dan 247 untuk skenario 3 dan 4; 382 dan 250 untuk skenario 7 dan 8) dibandingkan kelas positif (264 dan 180 untuk skenario 3 dan 4; 247 dan 168 untuk skenario 7 dan 8). Temuan ini mengindikasikan bahwa sentimen negatif lebih mendominasi persepsi masyarakat terhadap Patrick Kluivert sebagai pelatih baru Timnas, yang berarti masyarakat cenderung lebih banyak menyampaikan kritik, ketidakpuasan, atau keraguan terhadap kinerja atau keputusan pengangkatan Patrick Kluivert menggantikan Shin Tae-yong sebagai pelatih Timnas dibandingkan dukungan atau pujian.

### Skenario dengan Performa Paling Optimal

Pada Gambar 3.7 menampilkan *chart* hasil perbandingan rata-rata skenario yang telah dilakukan sebelumnya.



**Gambar 3.7** Chart Perbandingan Performa Rata-Rata Skenario

Berdasarkan hasil pengujian delapan skenario, rata-rata performa skenario 7 dan 8 (*Logistic Regression* dengan SMOTE) memberikan hasil terbaik dengan akurasi 83,80%, *precision* 82,80%, *recall* 82,70%, dan *F1-score* 82,80% (Tabel 3.20). Model ini menunjukkan keseimbangan optimal antara ketepatan prediksi (*precision*) dan kemampuan mendeteksi kelas positif (*recall*), terutama dalam menangani ketidakseimbangan kelas melalui penerapan teknik SMOTE. Penerapan SMOTE pada skenario 7 (rasio 70:30) dan skenario 8 (rasio 80:20) secara efektif meningkatkan *recall* hingga 82,70%, memungkinkan model untuk lebih baik mendeteksi kelas minoritas (misalnya, sentimen negatif) tanpa mengorbankan akurasi secara signifikan.

Dibandingkan dengan skenario 5 dan 6 (*Logistic Regression* tanpa SMOTE), yang mencatat akurasi rata-rata 83,76%, *precision* 84,50%, *recall* 81,50%, dan *F1-score* 82,50% (Tabel 3.19), skenario 7 dan 8 unggul dalam hal *recall* dan *F1-score*. Peningkatan *recall* ini menunjukkan bahwa *Logistic Regression* dengan SMOTE lebih efektif dalam mengidentifikasi kelas minoritas, yang krusial dalam konteks analisis sentimen di mana distribusi kelas sering kali tidak seimbang, seperti dominasi sentimen netral dalam *dataset* ini. Meskipun *precision* pada skenario 7 dan 8 sedikit lebih rendah (82,80% vs. 84,50%), *F1-score* yang seimbang 82,80% menunjukkan bahwa model ini memberikan performa yang lebih konsisten untuk semua kelas sentimen (negatif, netral, dan positif).

Skenario *Naive Bayes* (skenario 1 hingga 4) menunjukkan performa yang lebih rendah secara keseluruhan. Rata-rata skenario 1 dan 2 (*Naive Bayes* tanpa SMOTE) hanya mencapai akurasi 80,50%, *precision* 84,60%, *recall* 76,00%, dan *F1-score* 78,50% (Tabel 3.17), sedangkan skenario 3 dan 4 (*Naive Bayes* dengan SMOTE) mencapai akurasi 81,30%, *precision* 80,00%, *recall* 82,10%, dan *F1-score* 80,80% (Tabel 3.18). Meskipun SMOTE meningkatkan *recall* pada *Naive Bayes*, performa keseluruhan tetap di bawah *Logistic Regression*, terutama karena *Logistic Regression* lebih mampu menangani fitur-fitur kompleks dari data teks yang telah diproses dengan TF-IDF.

*Chart* pada Gambar 3.7 di atas memperlihatkan bahwa skenario 7 dan 8 (*Logistic Regression* dengan SMOTE) secara konsisten memiliki performa yang kompetitif, dengan *recall* dan *F1-score* yang lebih tinggi dibandingkan skenario lain, menjadikannya pilihan terbaik untuk pengklasifikasian sentimen dalam penelitian ini. Penerapan SMOTE memungkinkan model

untuk lebih sensitif terhadap kelas minoritas, seperti sentimen negatif, yang penting untuk memahami kritik publik terhadap pergantian pelatih Timnas Indonesia. Oleh karena itu, *Logistic Regression* dengan SMOTE pada rasio 70:30 dan 80:20 adalah konfigurasi optimal untuk analisis sentimen dalam konteks ini, terutama ketika ketidakseimbangan kelas menjadi faktor kunci.

### Perbandingan Hasil dengan Penelitian Terkait

Penelitian ini membandingkan performa algoritma *Naive Bayes* dan *Logistic Regression* dalam analisis sentimen publik terhadap pergantian pelatih Timnas Indonesia di media sosial X, dengan hasilnya dibandingkan terhadap penelitian Bagus Ramadhani dan Ryan Randy Suryono (2024) yang menganalisis sentimen terhadap teknologi *metaverse* menggunakan algoritma yang sama[3]. Perbandingan ini berfokus pada matriks evaluasi utama, yaitu akurasi, *precision*, *recall*, dan *F1-score*, untuk menilai efektivitas kedua algoritma.

Dalam penelitian ini, performa terbaik dicapai oleh *Logistic Regression* dengan SMOTE pada skenario 7 dan 8, dengan rata-rata akurasi 83,80%, *precision* 82,80%, *recall* 82,70%, dan *F1-score* 82,80% (Tabel 3.20). Penerapan SMOTE pada rasio data 70:30 (skenario 7) dan 80:20 (skenario 8) secara baik dalam mengatasi ketidakseimbangan kelas dalam *dataset*, yang terdiri dari 4.847 *tweet* berbahasa Indonesia, dengan distribusi sentimen yang didominasi oleh kelas netral. Peningkatan *recall* hingga 82,70% menunjukkan bahwa model ini mampu mendeteksi kelas minoritas (seperti sentimen negatif) dengan lebih baik, yang krusial untuk memahami kritik publik terhadap pengangkatan Patrick Kluivert sebagai pelatih Timnas Indonesia. Meskipun *precision* sedikit lebih rendah dibandingkan *Logistic Regression* tanpa SMOTE (82,80% vs. 84,50% pada skenario 5 dan 6, Tabel 3.19), *F1-score* yang seimbang menunjukkan bahwa *Logistic Regression* dengan SMOTE memberikan performa yang lebih konsisten untuk semua kelas sentimen.

Sebaliknya, penelitian oleh Ramadhani dan Suryono melaporkan performa yang lebih tinggi untuk *Logistic Regression* dan *Naive Bayes*. Dalam penelitian tersebut, yang menggunakan *dataset* 6.728 *tweet* dari media sosial X (Agustus 2022–April 2023), *Logistic Regression* tanpa SMOTE mencapai akurasi 91% dan *Naive Bayes* 90%. Setelah penerapan SMOTE, *Logistic Regression* meningkat menjadi akurasi 95% dengan *F1-score* 95%, sedangkan *Naive Bayes* mencapai akurasi 91%[3]. Perbedaan performa ini kemungkinan disebabkan oleh beberapa faktor. Pertama, *dataset* dalam penelitian Ramadhani dan Suryono lebih besar (6.728 vs. 4.847 *tweet*), yang dapat meningkatkan kemampuan model untuk menangkap pola sentimen. Kedua, distribusi sentimen dalam penelitian mereka sangat tidak seimbang (86% positif vs. 14% negatif), sehingga SMOTE memberikan dampak yang lebih signifikan dalam meningkatkan akurasi. Ketiga, topik *metaverse* cenderung memiliki pola sentimen yang lebih jelas (antusiasme terhadap teknologi baru), sedangkan isu pergantian pelatih Timnas Indonesia bersifat lebih emosional dan kompleks, dengan sentimen negatif yang lebih dominan sebagaimana terlihat dari *confusion matrix* (misalnya, 382 kelas negatif benar vs. 247 kelas positif benar pada skenario 7, Tabel 3.14).

Perbandingan ini menunjukkan bahwa *Logistic Regression* secara konsisten mengungguli *Naive Bayes* dalam kedua penelitian, terutama ketika dikombinasikan dengan SMOTE. Dalam penelitian ini, *Logistic Regression* dengan SMOTE (skenario 7 dan 8) menjadi konfigurasi optimal karena kemampuannya menyeimbangkan *precision* dan *recall*, yang penting untuk menangani ketidakseimbangan kelas dalam *dataset* sentimen. Meskipun akurasi dalam penelitian ini (83,80%) lebih rendah dibandingkan penelitian Ramadhani dan Suryono (95%), hasil ini tetap relevan mengingat kompleksitas topik dan ukuran *dataset* yang lebih kecil. Penerapan SMOTE dalam penelitian ini terbukti efektif untuk meningkatkan deteksi kelas minoritas, seperti sentimen negatif, yang mencerminkan kritik masyarakat terhadap keputusan pergantian pelatih.

**PENUTUP****Kesimpulan**

Berdasarkan hasil dan pembahasan dari penelitian yang dilakukan dapat ditarik kesimpulan bahwa komparasi algoritma menunjukkan *Logistic Regression* dengan SMOTE lebih unggul dibandingkan *Naive Bayes* untuk klasifikasi sentimen, dengan akurasi 83,80% berbanding 81,30%. Selain itu, hasil analisis sentimen menunjukkan dominasi sentimen negatif, mencerminkan banyaknya kritik masyarakat terhadap pergantian pelatih Timnas Indonesia pada kualifikasi Piala Dunia 2026.

**Saran**

Berdasarkan hasil analisis sentimen terhadap pergantian pelatih Timnas Indonesia di X (*Twitter*), disarankan untuk memperluas data dari platform lain seperti *Instagram* atau *YouTube*, melakukan pembersihan data dengan lebih cermat untuk menghilangkan *noise*, serta berkonsultasi dengan ahli bahasa untuk mendapatkan wawasan tambahan mengenai fitur-fitur kebahasaan yang relevan, termasuk struktur kalimat, penggunaan majas, hingga konteks sosial budaya yang melatarbelakangi penggunaan sarkasme dalam teks yang dapat digunakan untuk meningkatkan performa model.

**Daftar Pustaka**

- [1] D. Aulia, "Semangat Suporter Timnas Indonesia : Wujud Nasionalisme Di Arena Sepak Bola Dan Pengaruh Di Kancah Internasional," *WISSEN J. Ilmu Sos. dan Hum.*, vol. 3, no. 1, pp. 157–169, 2025.
- [2] A. Susanto, I. Agung Dzulkarnain, P. Studi Statistika, and F. Matematika Dan Ilmu Pengetahuan Alam, "Analisis Sentimen Data Twitter Topik Ekonomi Dan Industri Dengan Metode Naive Bayes Dan Random Forest," *J. Ilm. Wahana Pendidikan*, Oktober, vol. 2023, no. 20, pp. 59–65, 2023, [Online]. Available: <https://doi.org/10.5281/zenodo.8398895>
- [3] B. Ramadhani and R. R. Suryono, "Komparasi Algoritma Naive Bayes Dan Logistic Regression Untuk Analisis Sentimen Metaverse," *J. Mnemon.*, vol. 7, no. 1, pp. 31–39, 2024, doi: 10.36040/mnemonic.v7i1.8977.
- [4] M. Chieka and A. Kurniasih, "Teknik SMOTE untuk Mengatasi Imbalance Class dalam Klasifikasi Bank Customer Churn Menggunakan Algoritma Naive bayes dan Logistic Regression," *Pros. Semin. Nas. Mhs. Bid. Ilmu Komput. dan Apl.*, vol. 4, no. 2, pp. 552–559, 2023.
- [5] A. M. Pravina, I. Cholissodin, and P. P. Adikara, "Analisis Sentimen Tentang Opini Maskapai Penerbangan pada Dokumen Twitter Menggunakan Algoritme Support Vector Machine (SVM)," *J-PTIHK J. Pengemb. Teknlogi Inf. dan Ilmu Komput.*, vol. 3, no. 3, pp. 53–56, 2019, doi: 10.1061/9780784415313.ch06.
- [6] Z. B. Toyibah, Y. N. Putri, P. Puandini, Z. M. Widodo, and A. T. Ni'mah, "Perbandingan Kinerja Algoritma Multinomial Naive Bayes dan Logistic Regression pada Analisis Sentimen Movie Ratings IMDB," *J. Ilm. Edutic Pendidik. dan Inform.*, vol. 10, no. 2, pp. 181–189, 2024, doi: 10.21107/edutic.v10i2.28150.
- [7] Y. Hariyanti, S. Kacung, and B. Santoso, "Analisis Sentimen Terhadap Putusan Mahkamah Konstitusi Tentang Batasan Umur Capres Dan Cawapres Menggunakan Metode Naive Bayes," *Multidiscip. Indones. Cent. J.*, vol. 1, no. 1, pp. 517–525, 2024.
- [8] M. R. Nurhusen, J. Indra, and K. A. Baihaqi, "Analisis Sentimen Pengguna Twitter Terhadap Kenaikan Harga Bahan Bakar Minyak (BBM) Menggunakan Metode Logistic Regression," *J. Media Inform. Budidarma*, vol. 7, no. 1, p. 276, 2023, doi: 10.30865/mib.v7i1.5491.
- [9] D. Musfiroh, U. Khaira, P. E. P. Utomo, and T. Suratno, "Analisis Sentimen terhadap Perkuliahan Daring di Indonesia dari Twitter Dataset Menggunakan InSet Lexicon,"

- MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 1, no. 1, pp. 24–33, 2021, doi: 10.57152/malcom.v1i1.20.
- [10] R. N. Ikhsani and F. F. Abdulloh, “Optimasi SVM dan Decision Tree Menggunakan SMOTE Untuk Mengklasifikasi Sentimen Masyarakat Mengenai Pinjaman Online,” *J. Media Inform. Budidarma*, vol. 7, no. 4, p. 1667, 2023, doi: 10.30865/mib.v7i4.6809.
- [11] S. Kemp, “Digital 2025: Indonesia,” *datareportal.com*. Accessed: May 12, 2025. [Online]. Available: [https://datareportal.com/reports/digital-2025-indonesia?rq=indonesia 2025](https://datareportal.com/reports/digital-2025-indonesia?rq=indonesia%2025)
- [12] I. Mehta, “Twitter has officially changed its logo to ‘X,’” *techcrunch.com*. Accessed: May 12, 2025. [Online]. Available: <https://techcrunch.com/2023/07/24/twitter-has-officially-changed-its-logo-to-x/>
- [13] S. Kemp, “X Users, Stats, Data & Trends for 2025,” *datareportal.com*. Accessed: May 12, 2025. [Online]. Available: <https://datareportal.com/essential-x-stats>
- [14] Uefa, “Kluivert makes Valencia move,” *uefa.com*. Accessed: May 12, 2025. [Online]. Available: <https://www.uefa.com/news-media/news/01aa-0f8ea596d566-dc8738f31b7c-1000--kluivert-makes-valencia-move/>
- [15] H. Aditya and P. Niken, “Patrick Kluivert: Grassroots development key to homegrown soccer talent,” *thejakartapost.com*. Accessed: May 12, 2025. [Online]. Available: <https://www.thejakartapost.com/opinion/2025/04/11/patrick-kluivert-grassroots-development-key-to-homegrown-soccer-talent.html>
- [16] S. P. Kristanto and L. Hakim, “Ekstraksi Informasi Destinasi Wisata Populer Jawa Timur Menggunakan Depth-First Crawling,” *MATRIK J. Manajemen, Tek. Inform. dan Rekayasa Komput.*, vol. 21, no. 1, pp. 229–236, 2021, doi: 10.30812/matrik.v21i1.1081.
- [17] N. A. Salsabila, “Analisis sentimen pada media sosial twitter terhadap tokoh gus dur menggunakan metode naïve bayes dan support vector machine (svm),” Universitas Islam Negeri Syarif Hidayatullah Jakarta, 2023. [Online]. Available: <https://repository.uinjkt.ac.id/dspace/handle/123456789/66253>
- [18] B. Darmawan, A. Dwi Laksito, M. Resa, A. Yudianto, and A. Sidauruk, “Analisis Perbandingan Ekstraksi Fitur Teks pada Sentimen Analisis Kenaikan Harga BBM,” *J. Mhs. Inform.*, vol. 11, no. 1, pp. 53–63, 2023, doi: 10.32832/krea-tif.v11i1.13819.
- [19] R. L. Atimi and Enda Esyudha Pratama, “Implementasi Model Klasifikasi Sentimen Pada Review Produk Lazada Indonesia,” *J. Sains dan Inform.*, vol. 8, no. 1, pp. 88–96, 2022, doi: 10.34128/jsi.v8i1.419.
- [20] Kelvin, J. Banjarnahor, E. Indra, and S. H. Sinurat, “Analisis perbandingan sentimen Corona Virus Disease-2019 (Covid19) pada Twitter Menggunakan Metode Logistic Regression Dan Support Vector Machine (SVM),” *J. Sist. Inf. dan Ilmu Komput. Prima(JUSIKOM PRIMA)*, vol. 5, no. 2, pp. 47–52, 2022, doi: 10.34012/jurnalsisteminformasidanilmukomputer.v5i2.2365.
- [21] M. Rafli Ghufuron *et al.*, “Analisis Sentimen Pengguna Twitter Terhadap Pemilu 2024 Berbasis Model XLM-T,” *Inf. adn Technol.*, no. 204, pp. 307–315, 2023.