EVALUASI PERFORMA NAIVE BAYES DAN SVM DALAM MEMPREDIKSI SENTIMEN ULASAN PENGGUNA APLIKASI PINTEREST

Indri Arianti Wibowo, Uce Indahyanti*, Irwan Alnarus Kautsar, Rohman Dijaya
Informatika, Universitas Muhammadiyah Sidoarjo
Jl. Raya Gelam No.250, Pagerwaja, Gelam, Kec. Candi, Kabupaten Sidoarjo, Jawa Timur, Indonesia
uceindahyanti@umsida.ac.id

ABSTRAK

Kemajuan teknologi digital telah mendorong pertumbuhan signifikan dalam penggunaan aplikasi mobile di berbagai kalangan pengguna, termasuk Pinterest yang menghasilkan banyak ulasan pengguna sebagai sumber informasi kepuasan. Meskipun demikian, banyaknya ulasan yang tersedia dalam bentuk teks yang tidak tersusun membuat proses analisis secara manual sulit dilakukan dan berpotensi menimbulkan bias akibat ketidakseimbangan data sentimen. Penelitian ini dilaksanakan guna membandingkan tingkat performa algoritma *Naive Bayes* dan *Support Vector Machine* (SVM) dalam pengelompokan sentimen ulasan pengguna aplikasi Pinterest. Metode penelitian meliputi *pre-processing*, pelabelan, ekstraksi fitur melalui penerapan *Term Frequency-Inverse Document Frequency* (TF-IDF), pembagian data 80:20 dan 70:30, serta penerapan *Random Oversampling* (ROS) dilakukan untuk mengatasi ketidakseimbangan data. Evaluasi diterapkan dengan memanfaatkan *confusion matrix* melalui metrik akurasi, *precision*, *recall*, dan *F1-score*. Temuan penelitian menunjukkan bahwa *Support Vector Machine* (SVM) pada kondisi data *Imbalanced* rasio 80:20 memperoleh akurasi global tertinggi sebesar 89,01% dan *F1-score* 88,71%, sedangkan *Naive Bayes* dengan *Random Oversampling* (ROS) rasio 70:30 menghasilkan *recall* kelas negatif tertinggi sebesar 87,03% dan *F1-score* 72,68%. Penelitian menegaskan bahwa evaluasi model pada data tidak seimbang perlu menitikberatkan pada performa per kelas, khususnya *recall* kelas negatif, bukan hanya akurasi global.

Kata kunci : *Analisis Sentimen, Pinterest, Naive Bayes, Support Vector Machine, Random Oversampling*

1. PENDAHULUAN

Seiring berkembangnya teknologi digital, penggunaan aplikasi mobile seperti Pinterest terus meningkat, dengan lebih dari 522 juta pengguna aktif bulanan di dunia pada tahun 2024 [1]. Pinterest merupakan platform media sosial berbasis visual yang memberi kesempatan pengguna untuk menjelajah, mengoleksi, dan membagikan ide serta inspirasi dalam format gambar atau video [2].

Banyaknya pengguna menghasilkan ulasan beragam yang dapat memengaruhi persepsi dan ekspektasi terhadap kualitas aplikasi [3]. Namun, jumlah ulasan yang sangat besar membuat analisis manual tidak efisien [4]. Sehingga diperlukan analisis sentimen untuk mengetahui tingkat kepuasan pengguna serta mengolah teks tidak terstruktur menjadi informasi yang bermanfaat dalam pengambilan keputusan [5].

Berdasarkan permasalahan tersebut, penelitian ini menganalisis perbandingan performa algoritma *Naive Bayes* dan *Support Vector Machine* (SVM) dalam mengklasifikasikan sentimen ulasan pengguna aplikasi Pinterest, guna mengidentifikasi opini ke dalam sentimen positif dan negatif serta mengevaluasi performa per kelas, khususnya pada sentimen negatif.

Meskipun berbagai penelitian telah menganalisis sentimen pada aplikasi mobile, belum ditemukan penelitian yang secara khusus mengkaji perbandingan kedua algoritma pada ulasan pengguna aplikasi Pinterest. Dengan demikian, penelitian ini dilakukan guna mengisi celah tersebut.

Untuk mengatasi ketidakseimbangan data, penelitian ini diawali dengan *pre-processing* teks dan ekstraksi fitur menggunakan TF-IDF. Data latih diseimbangkan menggunakan *Random Oversampling* (ROS) agar model tidak bias terhadap kelas mayoritas. Klasifikasi dilakukan dengan algoritma *Naive Bayes* dan *Support Vector Machine* (SVM). Performa model dievaluasi menggunakan *confusion matrix* melalui metrik akurasi, *presisi*, *recall*, dan *F1-score.*

Penelitian ini diharapkan mampu menyajikan gambaran yang lebih mendalam terkait persepsi pengguna serta mendukung peningkatan kualitas aplikasi Pinterest ke depannya.

2. TINJAUAN PUSTAKA

2.1. Teori dan Konsep Utama

Dalam analisis sentimen, opini dikelompokkan ke dalam kelas sentimen, seperti positif atau negatif. Penelitian ini membandingkan performa *Naive Bayes* dan *Support Vector Machine* (SVM). *Naive Bayes* merupakan algoritma klasifikasi yang menggunakan pendekatan probabilistik dengan asumsi bahwa setiap fitur bersifat independen, berdasarkan Teorema Bayes [6]. Sedangkan, *Support Vector Machine* (SVM) merupakan algoritma klasifikasi yang menentukan *hyperplane* terbaik sebagai batas pemisah antar kelas dengan memaksimalkan jarak terhadap *support vector* terdekat [7].

2.2. Penelitian Terdahulu

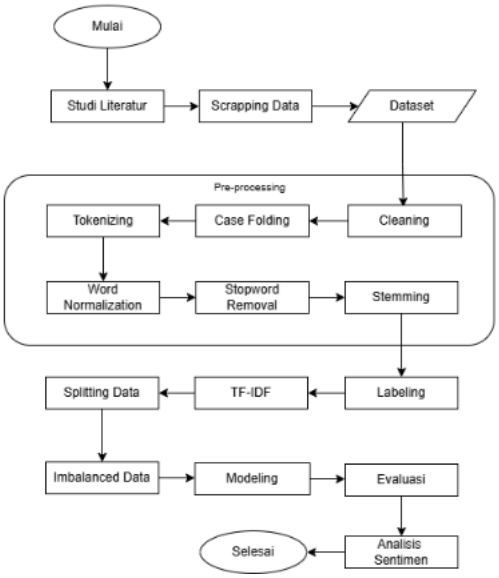
Beberapa penelitian telah melakukan perbandingan performa algoritma *Naive Bayes* dan *Support Vector Machine* (SVM) dalam klasifikasi sentimen ulasan pengguna dan menunjukkan hasil yang bervariasi. Pada penelitian platform *e-commerce* Tokopedia dan TikTok Shop, bahwa *Naive Bayes* dilaporkan lebih unggul dengan akurasi 97,50% dan *F1-score* 84,00%, sedangkan *Support Vector Machine* (SVM) memperoleh akurasi 94,90% dan *F1-score* 76,80% [8]. Penelitian lain pada aplikasi Ceria by BRI menghasilkan bahwa *Support Vector Machine* (SVM) memiliki performa akurasi lebih tinggi yaitu 88% daripada *Naive Bayes* yang hanya 84% [9]. Selain itu, penelitian yang menganalisis sentimen publik terhadap kinerja pemerintah dalam penanganan serangan ransomware pada algoritma *Support Vector Machine* (SVM), *Random Forest*, dan *Naive Bayes* pada 6.484 komentar *youtube*. Pengujian dilakukan dengan dan tanpa SMOTE, hasilnya menunjukkan bahwa penerapan dengan SMOTE meningkatkan performa seluruh model, di mana *Support Vector Machine* (SVM) mencapai akurasi tertinggi yaitu 96%, disusul oleh *Random Forest* 95% dan *Naive Bayes* 89% [10]. Penelitian lain juga menunjukkan algoritma *Support Vector Machine* (SVM) mampu menghasilkan performa yang baik, seperti pada prediksi pemilihan karir bagi alumni UMSIDA dengan akurasi mencapai 97% [11]. Sementara itu, pada analisis aplikasi Weverse, algoritma *Naive Bayes* rasio 80:20 menunjukkan performa optimal dengan akurasi sebesar 86,81% dan *F1-score* sebesar 83,88% [12].

2.3. Research GAP

Meskipun banyak penelitian mengkaji mengenai analisis sentimen dengan menggunakan algoritma *Naive Bayes* dan *Support Vector Machine* (SVM) di berbagai aplikasi, seperti game, *e-commerce*, dan keuangan. Belum terdapat penelitian yang melakukan perbandingan pada kedua algoritma tersebut dalam menganalisis sentimen ulasan pengguna pada aplikasi Pinterest. Sehingga, gap penelitian ini menjadi kesempatan untuk mengevaluasi dan membandingkan performa antara algoritma *Naive Bayes* dan *Support Vector Machine* (SVM) pada data ulasan aplikasi Pinterest, serta bertujuan untuk mengisi kekosongan literatur terkait topik ini.

3. METODE PENELITIAN

Data bersumber dari ulasan pengguna aplikasi Pinterest di *Google Play Store*, yang kemudian dipublikasikan secara terbuka dan dapat diakses melalui platform *Kaggle* pada tautan berikut: https://www.kaggle.com/datasets/indriarianti/dataset-pinterest. Selain itu, *source code* yang digunakan dalam proses pengolahan dan analisis data diunggah pada repositori *GitHub* yang tersedia pada: https://github.com/indriarianti12/SentimenPinterest.



Gambar 1. Tahapan Penelitian

Gambar 1 menjelaskan proses analisis sentimen ini mencakup tahapan seperti scraping data, *pre-processing*, labeling, ekstraksi fitur (TF-IDF), splitting data, *Imbalanced* data, modeling dan evaluasi.

3.1. Studi Literatur

Studi literatur dilakukan untuk mengumpulkan berbagai sumber ilmiah yang relevan, seperti buku dan jurnal, guna memperkuat landasan teori serta mendukung analisis terhadap permasalahan penelitian.

3.2. Scraping Data

Data penelitian berasal dari ulasan pengguna aplikasi Pinterest yang diambil dari *Google Play Store.* Hasil pengumpulan data tersebut kemudian diunggah ke *Kaggle* sebagai repositori data publik agar dapat diakses dan digunakan kembali oleh peneliti selanjutnya. Proses tahapan pengolahan data, pemodelan, serta evaluasi dilakukan dengan menggunakan *Google Colab* dengan alur pengujian yang bersifat umum dan dapat diterapkan pada dataset lain dengan karakteristik serupa. Data yang digunakan terdiri dari beberapa atribut, yaitu id, rating, ulasan, dan *date*.

3.3. Pre-processing

Pre-processing adalah tahapan dalam pengolahan data guna mengoptimalkan kualitas data mentah agar analisis yang dilakukan menjadi akurat serta mengurangi potensi permasalahan. Tahapan ini meliputi *cleaning*, *case folding*, *tokenizing*, *word normalization*, *stopword removal* dan *stemming* [13]. Dalam penelitian ini, data yang digunakan difokuskan pada ulasan berbahasa Indonesia.

3.4. Labeling

Pelabelan dilakukan untuk mengategorikan sentimen ulasan pengguna menjadi positif atau negatif. Proses ini memanfaatkan rating pengguna dan metode berbasis leksikon (*lexicon-based*) [14]. Metode ini memberikan label secara otomatis dengan mencocokkan kata dalam ulasan dengan kamus sentimen yang dikelompokkan menjadi positif dan negatif [15]. Pelabelan dilakukan secara semi-otomatis, dengan rating 1-2 sebagai negatif, rating 4-5 sebagai positif, dan rating 3 dikategorikan sebagai ambigu. Oleh karena itu, ulasan dengan rating 3 diberi label menggunakan pendekatan *lexicon-based*, di mana skor sentimen dihitung dari bobot kata, skor > 0 diklasifikasikan sebagai positif dan skor ≤ 0 sebagai negatif. Namun, pendekatan ini memiliki keterbatasan yang dapat menimbulkan bias karena rating tidak selalu selaras dengan isi ulasan, serta metode *lexicon-based* kurang mampu menangkap konteks, ironi, dan makna implisit, sehingga berpotensi menimbulkan bias dalam hasil klasifikasi.

3.5. Ekstraksi Fitur (TF-IDF)

Term Frequency-Inverse Document Frequency (TF-IDF) ialah metode yang mentransformasikan teks dalam format numerik dengan mempertimbangkan frekuensi kemunculan kata dalam suatu dokumen (TF) serta tingkat kelangkaan kemunculan kata pada seluruh kumpulan dokumen (IDF) [16]. Dengan demikian, dapat membantu model dalam mengenali kata-kata yang paling berpengaruh terhadap sentimen setiap ulasan [17]. Berikut perhitungan *Term Frequency* (TF):

$$\mathrm{TF}_{t,d} = \frac{\text{Jumlah kemunculan kata } t \text{ dalam dokumen } d}{\text{Jumlah total kata dalam dokumen } d} \quad (1)$$

Rumus *Inverse Document Frequency* (IDF):

$$\mathrm{IDF}_t = \log\left(\frac{\text{Jumlah total dokumen } (N)}{\text{Jumlah dokumen yang mengandung kata } t (DF_t)}\right) \quad (2)$$

Hasil dari TF-IDF:

$$\mathrm{TF-IDF}_{t,d} = \mathrm{TF}_{t,d} \times \mathrm{IDF}_t \quad (3)$$

3.6. Splitting Data

Dataset dibagi dengan memisahkan ke dalam dua subset yaitu data latih (*training data*) dan data uji (*testing data*) menggunakan rasio pembagian sebesar 80:20 dan 70:30 [18]. Rasio 80:20 memberikan proporsi lebih besar untuk data latih, sehingga model dapat mempelajari pola dengan baik. Sementara rasio 70:30 menyediakan data uji lebih banyak untuk menguji kemampuan generalisasi model. Kedua rasio digunakan untuk menganalisis perbedaan pembagian data terhadap performa model pada tiap kelas sentimen, terutama dalam mendeteksi kelas negatif sebagai kelas minoritas, serta mengevaluasi dampak *Random Oversampling* (ROS) terhadap *recall* dan *F1-score*.

3.7. Imbalanced Data

Pada banyaknya dataset terutama dalam klasifikasi, sering terjadi *Imbalanced* data, di mana satu kelas memiliki jumlah sampel yang lebih banyak. Untuk mengatasi hal ini, *Random Oversampling* (ROS) digunakan untuk menambah sampel pada kelas minoritas. Dengan demikian, distribusi data pada kelas menjadi lebih merata [19].

3.8. Modeling

Tahap pemodelan, *Naive Bayes* dan *Support Vector Machine* (SVM) diterapkan untuk melakukan klasifikasi sentimen pada ulasan pengguna. Dengan hanya dua kategori sentimen yaitu positif dan negatif, penelitian ini menggunakan pendekatan klasifikasi biner.

3.9. Evaluasi

Pada tahap terakhir, model dievaluasi melalui *confusion matrix* guna mengukur performa algoritma menggunakan metrik akurasi, *precision*, *recall*, dan *F1-score*. Akurasi menilai persentase prediksi benar dari total keseluruhan, *precision* merepresentasikan tingkat akurat model dalam mengklasifikasi kelas positif, *recall* mengevaluasi kemampuan model menangkap seluruh data pada kelas minoritas, dan *F1-score* metrik yang menyeimbangkan antara *precision* dan *recall* [20].

4. HASIL DAN PEMBAHASAN

4.1. Scraping Data

Sebanyak 10.000 ulasan dikumpulkan dalam rentang waktu Januari 2020 hingga Januari 2025 sebagai data penelitian.

Tabel 1. Hasil Scraping Data

Id	Rating	Ulasan	Date
c6a3f68c-b438-4e51-856a-32284abfa451	1	Ga bisa daftar tolong diperbaiki!!	2025-01-06
d1532942-9f9e-4bdb-aed7-35cdfed7e746	2	Kenapa akun saya di nonaktifkan:((	2024-12-15
8688c0c7-67e8-4c62-bac9-88c1bbe429fe	3	Bagus tapi ada kurangnya	2023-12-04
41f736a0-41fa-4576-a3f5-ebaae33ec6d6	4	Ini sangat keren	2022-12-04
342a1020-7d80-43a0-b0b0-f4cbe8c8d981	5	Aplikasi ini bagus pake banget, bisa bikin nyari inspirasi baru...	2021-12-23

Tabel 1 menampilkan hasil scraping ulasan pengguna aplikasi Pinterest yang mencakup id, rating, ulasan, dan *date*. Data ini menjadi data mentah yang selanjutnya diproses pada tahap *pre-processing* sebelum dilakukan analisis sentimen.

4.2. Pre-processing

Tahap ini adalah pre-processing. Data yang digunakan hanya pada ulasan berbahasa Indonesia. Proses scraping mengumpulkan 10.000 ulasan, kemudian dilakukan penyaringan dengan hanya mengambil ulasan berbahasa Indonesia dan menghapus ulasan dalam bahasa lainnya. Sehingga, diperoleh 5.681 ulasan berbahasa Indonesia yang selanjutnya digunakan dalam tahap analisis.

4.3. Cleaning

Cleaning ialah membersihkan karakter yang bukan relevan. Termasuk tanda baca, angka, emoji dan simbol.

Tabel 2. Hasil *Cleaning*

Sebelum	Sesudah
Aplikasi ini bagus pake banget, bisa bikin nyari inspirasi baru...	Aplikasi ini bagus pake banget bisa bikin nyari inspirasi baru

Tabel 2 memperlihatkan hasil *cleaning* dengan karakter yang tidak diperlukan. Dengan demikian, data menjadi lebih bersih dan siap diproses lebih lanjut.

4.4. Case Folding

Case folding ialah menyeragamkan huruf dalam teks menjadi huruf kecil.

Tabel 3. Hasil *Case Folding*

Sebelum	Sesudah
Aplikasi ini bagus pake banget bisa bikin nyari inspirasi baru	aplikasi ini bagus pake banget bisa bikin nyari inspirasi baru

Tabel 3 memperlihatkan hasil *case folding* dengan mengubah huruf menjadi huruf kecil guna meminimalkan perbedaan yang disebabkan oleh kapitalisasi.

4.5. Tokenizing

Tokenizing adalah tahap pemrosesan dengan membagi kalimat menjadi unit-unit lebih kecil.

Tabel 4. Hasil *Tokenizing*

Sebelum	Sesudah
aplikasi ini bagus pake banget bisa bikin nyari inspirasi baru	['aplikasi', 'ini', 'bagus', 'pake', 'banget', 'bisa', 'bikin', 'nyari', 'inspirasi', 'baru']

Tabel 4 memperlihatkan hasil *tokenizing* yaitu pemisahan kalimat ulasan menjadi unit kata (*token*). Hal ini untuk memudahkan analisis teks pada proses ekstraksi fitur.

4.6. Word Normalization

Tahap normalisasi yaitu mengubah singkatan, kata tidak baku, typo, dan slang menjadi kata baku menurut kaidah bahasa Indonesia. Proses ini dilakukan menggunakan *dictionary* kata baku berupa dataset kamus slang yang diperoleh dari platform *Kaggle* yang tersedia pada tautan berikut: https://www.kaggle.com/datasets/fornigulo/kamus-slag [21]. Pada *dictionary* tersebut digunakan untuk mengonversi kata tidak baku ke bentuk baku secara sistematis.

Tabel 5. Hasil *Word Normalization*

Sebelum	Sesudah
['aplikasi', 'ini', 'bagus', 'pake', 'banget', 'bisa', 'bikin', 'nyari', 'inspirasi', 'baru']	['aplikasi', 'ini', 'bagus', 'pakai', 'banget', 'bisa', 'bikin', 'mencari', 'inspirasi', 'baru']

Tabel 5 memperlihatkan hasil normalisasi kata, di mana kata tidak baku, singkatan, dan slang diubah menjadi bentuk kata baku untuk meningkatkan konsistensi teks.

4.7. Stopword Removal

Stopword removal yaitu tahap membuang kata yang tidak memiliki informasi penting.

Tabel 6. Hasil *Stopword Removal*

Sebelum	Sesudah
['aplikasi', 'ini', 'bagus', 'pakai', 'banget', 'bisa', 'bikin', 'mencari', 'inspirasi', 'baru']	['aplikasi', 'bagus', 'pakai', 'banget', 'bikin', 'mencari', 'inspirasi', 'baru']

Tabel 6 memperlihatkan hasil *stopword removal* yang membuat analisis lebih terfokus pada kata-kata penentu sentimen.

4.8. Stemming

Stemming adalah tahapan mengembalikan kata ke bentuk kata dasarnya dengan memanfaatkan *library* Sastrawi.

Tabel 7. Hasil *Stemming*

Sebelum	Sesudah
['aplikasi', 'bagus', 'pakai', 'banget', 'bikin', 'mencari', 'inspirasi', 'baru']	['aplikasi', 'bagus', 'pakai', 'banget', 'bikin', 'cari', 'inspirasi', 'baru']

Tabel 7 memperlihatkan hasil *stemming* yaitu pengubahan kata ke bentuk dasarnya untuk meningkatkan konsistensi ekstraksi fitur.

4.9. Labeling

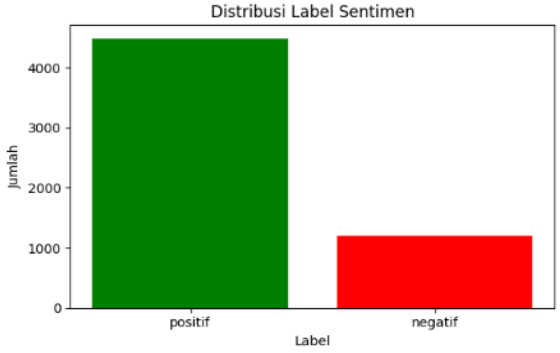
Tahap labeling bertujuan menandai setiap ulasan ke dalam sentimen positif atau negatif dengan memanfaatkan kombinasi rating pengguna.

Tabel 8. Sesudah Pre-processing dan Labeling

Rating	Ulasan Awal	Ulasan Akhir	Date	Label
1	Ga bisa daftar tolong diperbaiki!!	daftar perbaiki	2025-01-06	Negatif
2	Kenapa akun saya di nonaktifkan:(	akun nonaktif	2024-12-15	Negatif
3	Bagus tapi ada kurangnya	bagus kurang	2023-12-04	Positif

Rating	Ulasan Awal	Ulasan Akhir	Date	Label
4	Ini sangat keren	sangat keren	2022-12-04	Positif
5	Aplikasi ini bagus pake banget, bisa bikin nyari inspirasi baru...	aplikasi bagus pakai banget bikin cari inspirasi baru	2021-12-23	Positif

Tabel 8 menunjukkan contoh data ulasan setelah melalui seluruh tahapan *pre-processing* dan pelabelan sentimen. Setiap ulasan telah dikategorikan ke dalam sentimen positif atau negatif.



Gambar 2. Distribusi Label Sentimen

Berdasarkan gambar 2 distribusi label sentimen pada dataset ulasan aplikasi Pinterest. Bahwa jumlah ulasan positif jauh lebih banyak, yaitu sebanyak 4.482 data, dibanding dengan ulasan negatif yang berjumlah 1.199 data. Kondisi ini memperlihatkan distribusi data yang tidak merata, di mana kelas positif menjadi kelas mayoritas dan negatif sebagai kelas minoritas.

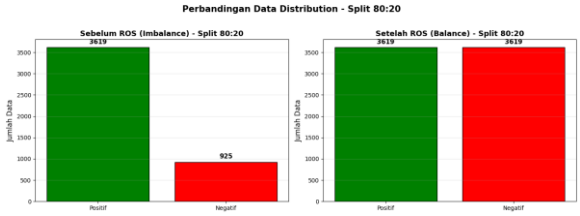
4.10. Penyeimbangan Data

Penyeimbangan data dilakukan dengan metode *Random Oversampling* (ROS) untuk mengatasi ketidakseimbangan antar kelas. Dengan metode ini, sampel kelas minoritas digandakan secara acak hingga sebanding dengan kelas mayoritas, sehingga proporsi data antar kelas menjadi lebih seimbang.

Tabel 9. Distribusi dan Penyeimbangan Data

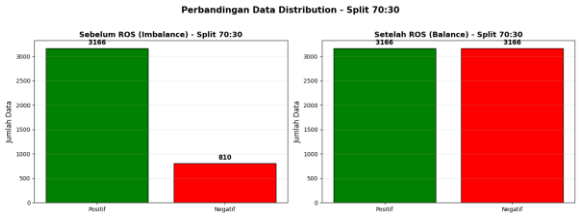
Rasio Split	Kelas	Data Sebelum ROS	Data Setelah ROS	Data Test (Asli)
80:20	Positif (Mayoritas)	3.619	3.619	905
	Negatif (Minoritas)	925	3.619	232
	Total data train	4.544	7.238	1.137
70:30	Positif (Mayoritas)	3.166	3.166	1.358
	Negatif (Minoritas)	810	3.166	347
	Total data train	3.976	6.332	1705

Tabel 9 menampilkan perbandingan distribusi data sentimen sebelum dan sesudah penerapan ROS pada rasio pembagian data 80:20 dan 70:30. Hasilnya memperlihatkan jumlah data kelas negatif meningkat hingga setara dengan kelas positif setelah ROS diterapkan. Kondisi ini menegaskan bahwa distribusi data latih menjadi lebih seimbang sehingga proses pelatihan model berlangsung dengan representasi kelas yang lebih merata.



Gambar 3. Perbandingan Data Rasio 80:20

Berdasarkan gambar 3, rasio pembagian data 80:20 sebelum penerapan ROS menunjukkan kondisi data yang tidak seimbang, data kelas negatif jauh lebih sedikit dibanding kelas positif. Setelah ROS diterapkan, jumlah kelas negatif disetarakan sehingga setara dengan kelas positif. Sehingga, distribusi data menjadi seimbang. Penyeimbangan ini penting agar model dapat dilatih secara seimbang pada kedua kelas dan bias terhadap kelas mayoritas dapat dikurangi.



Gambar 4. Perbandingan Data Rasio 70:30

Berdasarkan gambar 4, rasio pembagian data 70:30 sebelum penerapan ROS menunjukkan kondisi data yang tidak seimbang, data kelas negatif jauh lebih sedikit dibanding kelas positif. Setelah ROS diterapkan, jumlah kelas negatif disetarakan sehingga setara dengan kelas positif. Sehingga, distribusi data menjadi seimbang. Penyeimbangan ini penting agar model dapat dilatih secara seimbang pada kedua kelas dan bias terhadap kelas mayoritas dapat dikurangi.

4.11. Hasil Pengujian Algoritma

Pengujian dilakukan dengan membandingkan dua algoritma, yaitu *Naive Bayes* dan *Support Vector Machine* (SVM) menggunakan 2 pembagian data dan 2 kondisi data latih yaitu *Imbalanced* dan *Balanced*. Tujuan pengujian ini adalah untuk membandingkan efektivitas kedua algoritma serta mengevaluasi dampak penyeimbangan data latih menggunakan metode *Random Oversampling* (ROS) terhadap metrik.

4.12. Hasil Pengujian Rasio 80:20

Pengujian ini dilakukan dengan membagi dataset, di mana 80% sebagai data latih dan 20% sebagai data uji. Hasil antara kondisi data latih yang *Imbalanced* dan *Balanced* tercantum pada Tabel 10 berikut:

Tabel 10. Hasil Rasio 80:20

Algoritma	Kondisi Train	Akurasi	Precision	Recall	F1-score
Naive Bayes	Imbalanced	86,72 %	86,07%	86,72 %	85,30 %
Naive Bayes	Balanced	85,31 %	88,44%	85,31 %	86,18 %
SVM	Imbalanced	89,01 %	88,60%	89,01 %	88,71 %
SVM	Balanced	88,30 %	88,89%	88,30 %	85,54 %

Tabel 10 menampilkan hasil performa algoritma *Naive Bayes* dan *Support Vector Machine* (SVM) rasio pembagian data 80:20, baik pada kondisi data *Imbalanced* maupun *Balanced*. Hasil ini menunjukkan perbedaan kinerja kedua algoritma serta dampak penerapan ROS terhadap nilai akurasi, *precision*, *recall*, dan *F1-score*.

4.13. Hasil Pengujian Rasio 70:30

Pengujian ini dilakukan dengan membagi dataset, di mana 70% sebagai data latih dan 30% sebagai data uji. Hasil antara kondisi data latih yang *Imbalanced* dan *Balanced* tercantum pada Tabel 11 berikut:

Tabel 11. Hasil Rasio 70:30

Algoritma	Kondisi Train	Akurasi	Precision	Recall	F1-score
Naive Bayes	Imbalanced	86,22 %	85,45%	86,22 %	84,66 %
Naive Bayes	Balanced	86,69 %	89,41%	86,69 %	87,43 %
SVM	Imbalanced	87,97 %	88,59%	88,97 %	88,70 %
SVM	Balanced	88,21 %	88,92%	88,21 %	88,48 %

Tabel 11 menampilkan hasil performa algoritma *Naive Bayes* dan *Support Vector Machine* (SVM) pada rasio pembagian data 70:30, baik pada kondisi data *Imbalanced* maupun *Balanced*. Hasil ini menunjukkan perbedaan kinerja kedua algoritma serta dampak penerapan ROS terhadap nilai akurasi, *prescision*, *recall*, dan *F1-score*.

4.14. Model Optimal dan Evaluasi Split Rasio

Berdasarkan hasil pengujian pada Tabel 10 dan Tabel 11, algoritma *Support Vector Machine* (SVM) dengan rasio pembagian data 80:20 pada kondisi data *Imbalanced* menunjukkan performa terbaik secara global dengan nilai akurasi sebesar 89,01% dan *F1-score* sebesar 88,71%. Rasio pembagian data 80:20 dan 70:30 digunakan dalam penelitian ini dengan pertimbangan untuk memberikan porsi data latih yang memadai agar model mampu mengenali pola sentimen dengan efektif, sekaligus menyiapkan data uji yang cukup agar dapat melakukan penilaian performa model secara tepat. Meskipun *Support Vector Machine* (SVM) menunjukkan performa yang efektif pada kelas mayoritas (positif), evaluasi per kelas menunjukkan bahwa model masih kurang efektif dalam mendeteksi kelas negatif sebagai kelas minoritas, sehingga metode ROS digunakan pada data latih agar performa model pada kelas negatif, khususnya *recall* dan *F1-score* meningkat. Oleh karena itu, evaluasi model pada penelitian ini tidak sekedar didasarkan pada akurasi global, melainkan juga pada keseimbangan performa antar kelas sentimen, khususnya kemampuan dalam mengklasifikasikan kelas negatif.

4.15. Dampak ROS dan Trade-Off

Penerapan *Random Oversampling* (ROS) dilakukan untuk mengatasi ketidakseimbangan data sentimen, di mana dominasi kelas positif menyebabkan akurasi tinggi belum tentu mencerminkan kemampuan model dalam mengenali kelas negatif sebagai kelas minoritas. Dengan demikian, evaluasi performa tidak hanya didasarkan pada akurasi global, tetapi juga pada metrik per kelas, khususnya *recall* dan *F1-score* kelas negatif [22].

Tabel 12. Perbandingan Performa Kelas Negatif

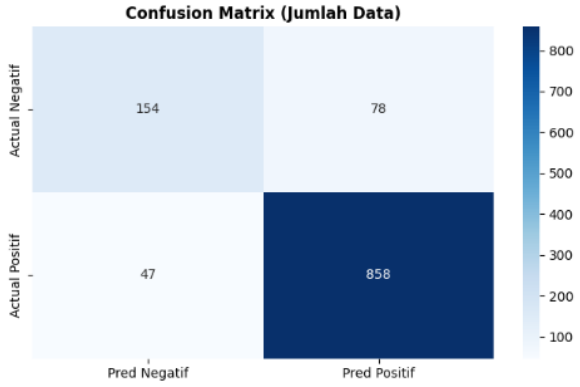
Algoritma	Rasio Split	Kondisi Train	Recall	F1-score
Naive Bayes	80:20	Imbalanced	46,55%	58,86%
		Balanced (ROS)	85,34%	70,34%
	70:30	Imbalanced	44,38%	56,72%
		Balanced (ROS)	87,03%	72,68%
SVM	80:20	Imbalanced	66,38%	71,13%
		Balanced (ROS)	77,16%	72,91%
	70:30	Imbalanced	66,86%	77,17%
		Balanced (ROS)	77,81%	72,87%

Berdasarkan Tabel 12, penerapan ROS meningkatkan performa klasifikasi kelas negatif pada kedua algoritma, yang ditunjukkan oleh kenaikan nilai *recall* dan *F1-score* setelah data latih diseimbangkan, meskipun disertai penurunan akurasi global sebagai bentuk *trade-off* pada data tidak seimbang. Peningkatan paling signifikan terjadi pada *Naive Bayes* rasio 70:30, dengan *recall* meningkat dari 44,38% menjadi 87,03% dan *F1-score* dari 56,72% menjadi 72,68%. Hal ini menunjukkan bahwa ROS membantu model mempelajari karakteristik kelas minoritas dengan lebih baik. Pada *Support Vector Machine* (SVM), ROS juga meningkatkan *recall*, namun performanya masih berada di bawah *Naive Bayes* dalam mendeteksi sentimen negatif. Secara keseluruhan, ROS terbukti efektif meningkatkan

performa klasifikasi kelas minoritas, terutama pada algoritma *Naive Bayes*.

4.16. Analisis Confusion Matrix dan Metrik Evaluasi

Analisis *confusion matrix* diterapkan pada model yang memberikan hasil optimal secara keseluruhan, yaitu *Support Vector Machine* (SVM) rasio 80:20 pada kondisi *Imbalanced* yang memperoleh akurasi 89,01%. Hasil *confusion matrix* menunjukkan dominasi *True Positive*, namun terlihat *False Positive* dan *False Negative* yang menyebabkan nilai *recall* kelas negatif relatif rendah sebesar 66,38%, mengindikasikan bias model terhadap kelas mayoritas akibat ketidakseimbangan data. Oleh karena itu, fokus analisis diarahkan pada kesalahan prediksi kelas negatif sebagai kelas minoritas. Penerapan ROS bertujuan menyeimbangkan data latih agar model mampu meningkatkan kemampuannya dalam mendeteksi ulasan negatif, yang tercermin dari peningkatan nilai *recall* dan *F1-score* kelas negatif. Meskipun peningkatan pada *recall* dapat diikuti oleh kenaikan *False Positive* dan penurunan akurasi sebagai bentuk *trade-off*, pola serupa juga terlihat pada algoritma *Naive Bayes* rasio 70:30, di mana *recall* kelas negatif meningkat signifikan menjadi 87,03% setelah penerapan ROS. Dengan demikian, akurasi global saja tidak memadai untuk mengevaluasi model, melainkan memperhatikan performa tiap kelas, khususnya kelas negatif sebagai fokus utama penanganan data tidak seimbang.



Gambar 5. Confusion Matrix

Pada gambar 5 menampilkan *confusion matrix* hasil klasifikasi sentimen yang menggambarkan performa model dalam mengklasifikasikan kelas negatif dan positif. Dari total data uji, model berhasil mengidentifikasi 154 data negatif dan 858 data positif secara benar. Namun, tetap ada kesalahan klasifikasi, yaitu 78 data negatif yang diidentifikasi sebagai positif (*false positive*) dan 47 data positif yang diidentifikasi sebagai negatif (*false negative*).

5. KESIMPULAN DAN SARAN

Dari hasil analisis, bahwa algoritma *Naive Bayes* dan *Support Vector Machine* (SVM) memiliki keunggulan masing-masing dalam menangani data ulasan Pinterest yang tidak seimbang, sesuai dengan fokus analisis. *Naive Bayes* dengan penerapan ROS rasio 70:30 menjadi model terbaik untuk deteksi sentimen negatif karena menghasilkan *recall* kelas negatif tertinggi, yang meningkat signifikan dari 44,38% menjadi 87,03%, sedangkan *Support Vector Machine* (SVM) meraih akurasi global paling tinggi yakni 89,01% pada rasio pembagian data 80:20. Namun, memiliki *recall* kelas negatif yang lebih rendah walaupun mengalami peningkatan setelah ROS diterapkan. Penelitian ini menegaskan bahwa pemilihan model sebaiknya didasarkan pada kemampuan mendeteksi kelas negatif bukan hanya akurasi total. Analisis ulasan menunjukkan sentimen positif berkaitan dengan sumber inspirasi, ide kreatif, tampilan menarik, dan kegunaan fitur. Sementara sentimen negatif didominasi permasalahan teknis, seperti gagal mendaftar, akun dinonaktifkan, dan masalah penggunaan aplikasi. Kontribusi penelitian ini mencakup evaluasi performa berbasis metrik per kelas, analisis dampak ketidakseimbangan data, serta *trade-off* antara akurasi global dan *recall* kelas negatif melalui ROS, dengan fokus pada evaluasi algoritma tanpa pengembangan UI/UX dan alur pengujian yang dapat direplikasi. Penelitian berikutnya disarankan untuk menguji metode penyeimbangan data lain, seperti SMOTE serta penerapan model deep learning seperti LSTM, BiLSTM, atau IndoBERT.

DAFTAR PUSTAKA

[1] B. of Apps, “Pinterest Revenue and Usage Statistics,” Business of Apps. Accessed: Jan. 02, 2026. [Online]. Available: https://www.businessofapps.com/data/pinterest-statistics/

[2] B. Uddin, D. D. F. Basam, and E. M. L. Wijayadi, “Efektivitas Pemanfaatan Pinterest Terhadap Kreativitas Pengguna,” *J. Nas. Komputasi dan Teknol. Inf.*, vol. 7, no. 4, pp. 678–684, 2024.

[3] A. Nurian and B. N. Sari, “Analisis Sentimen Ulasan Pengguna Aplikasi Google Play Menggunakan Naïve Bayes,” *J. Inform. dan Tek. Elektro Terap.*, vol. 11, no. 3s1, pp. 829–835, 2023, doi: 10.23960/jitet.v11i3s1.3348.

[4] Y. T. R. Sholiha, L. A. M. Nabilah, and Imron, “Analisis Sentimen Aplikasi Liputan6.Com pada Ulasan Pengguna di Google Playstore dengan Menggunakan Algoritma Support Vector Machine (Svm) dan Naïve Bayes,” *Saturnus J. Teknol. dan Sist. Inf.*, vol. 3, no. 3, pp. 30–51, 2025, doi: https://doi.org/10.61132/saturnus.v3i3.867.

[5] P. Anggraini and Winarsih, “KOMPARASI NAÏVE BAYES, SUPPORT VECTOR MACHINE, DAN RANDOM FOREST DALAM ANALISIS SENTIMEN APLIKASI SHOPEE DI GOOGLE PLAY STORE,” *JATI (Jurnal Mhs. Tek. Inform.*, vol. 9, no. 3, pp. 4451–4457, 2025.

[6] F. Damayanti, A. Rahim, and N. A. Verdikha, "ANALISIS SENTIMEN ULASAN PENGGUNA TERHADAP APLIKASI K24KLIK," *JATI (Jurnal Mhs. Tek. Inform.*, vol. 9, no. 2, pp. 3224–3230, 2025.

[7] J. A. Putra, A. Dharmawan, and J. Gondohanindijo, "SENTIMEN ANALISIS APLIKASI DIGITALENT MOBILE MENGGUNAKAN NAIVE BAYES DAN SVM DENGAN EKSTRAKSI FITUR TF-IDF," *J. Inf. Technol. Comput. Sci.*, vol. 7, 2024.

[8] A. Nabilla M. P., A. Adrian Y. L., A. Indra J., Y. Arya S., Sumanto, and A. Diah H., "Komparasi Naive Bayes dan SVM untuk Analisis Sentimen Pada E-Commerce Seller Center," *J. Sains Dan Teknol.*, vol. 5, no. 3, pp. 199–208, 2025.

[9] M. Aji, A. Maldini, and S. Andryana, "ANALISIS SENTIMEN ULASAN PENGGUNA APLIKASI PERBANKAN," *JATI (Jurnal Mhs. Tek. Inform.*, vol. 9, no. 3, pp. 4098–4105, 2025.

[10] M. I. Prayugah, U. Indahyanti, and N. Ariyanti, "ANALISIS SENTIMEN PUBLIK PADA PEMERINTAH DALAM SERANGAN RANSOMWARE DENGAN PENDEKATAN SMOTE," *JOISIE (Journal Inf. Syst. Informatics Eng.*, vol. 8, no. 2, pp. 333–343, 2024.

[11] M. D. Qur'ani, H. Setiawan, and I. A. Kautsar, "PENERAPAN METODE SUPPORT VECTOR MACHINE (SVM) UNTUK MEMPREDIKSI PEMILIHAN KARIR BAGI ALUMNI UMSIDA," *JIPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.*, vol. 10, no. 4, pp. 2906–2916, 2025.

[12] A. R. Annjani, U. Indahyanti, A. W. Azinar, and R. Dijaya, "PENERAPAN ALGORITMA MACHINE LEARNING UNTUK MEMPREDIKSI SENTIMEN KEPUASAN PENGGUNA WEVERSE," *JATI(Jurnal Mhs. Tek. Inform.*, vol. 10, no. 1, pp. 784–790, 2026.

[13] R. A. P. Sari, S. Kacung, and B. Santoso, "ANALISIS SENTIMEN LAYANAN KESEHATAN BPJS MENGGUNAKAN METODE SVM," *J. Inform. Teknol. dan Sains*, vol. 7, no. 2, pp. 878–885, 2025.

[14] S. A. S. Mola, D. L. B. Baun, I. O. Nunes, and M. M. A. R. Sani, "ANALISIS SENTIMEN APLIKASI HALO BCA DI GOOGLE PLAY STORE MENGGUNAKAN METODE NAIVE BAYES , SUPPORT VECTOR MACHINE DAN RANDOM FOREST PENDAHULUAN," *HOAQ J. Teknol. Inf.*, vol. 15, no. 2, pp. 69–79, 2024.

[15] D. Fitriono, R. Indriati, and A. Ristyawan, "Analisis Sentimen Ulasan Aplikasi Chatgpt Menggunakan Algoritma Support Vector Machine dan Lexicon Based," *JSITIK*, vol. 3, no. 2, pp. 101–111, 2025.

[16] D. N. N. Husnina, D. E. Ratnawati, and B. Rahayudi, "Analisis Sentimen Pengguna Aplikasi RedBus berdasarkan Ulasan di Google Play Store menggunakan Metode Naïve Bayes," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 7, no. 2, 2023.

[17] D. W. Bhatara and R. R. Suryono, "ANALISIS SENTIMEN APLIKASI BCA MOBILE MENGGUNAKAN ALGORITMA NAÏVE BAYES DAN SUPPORT VECTOR MACHINE," *JIPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.*, vol. 9, no. 4, pp. 1907–1917, 2024.

[18] F. A. Soraya and A. D. Indriyanti, "Perbandingan Algoritma Naïve Bayes dan Support Vector Machine dalam Analisis Sentimen Aplikasi Teman Bus," *JEISBI (Journal Emerg. Inf. Syst. Bus. Intell.*, vol. 05, no. 02, pp. 118–125, 2024.

[19] N. D. H. Sadikin and S. Susanti, "Analisis Sentimen Publik Terhadap Kampanye Pengurangan Sampah Plastik Menggunakan Algoritma Naïve Bayes," *J. FASILKOM*, vol. 15, no. 2, pp. 202–212, 2025.

[20] A. Y. Lubisa and M. Y. H. Setyawan, "Analisis Sentimen Aplikasi Pospay di PlayStore Menggunakan Algoritma Support Vector Machine & Naive Bayes," *J. Teknol. Dan Sist. Inf. Bisnis*, vol. 6, no. 3, pp. 514–521, 2024.

[21] Fornieli, "kamus_slang," Kaggle. Accessed: Feb. 26, 2026. [Online]. Available: https://www.kaggle.com/datasets/fornigulo/kamus-slag

[22] H. P. Jelita, M. I. Saad, and Wahyuni, "Penerapan Algoritma Naïve Bayes Dalam Analisis Sentimen Masyarakat Terhadap STMIK Widya Cipta Dharma," *Bull. Inf. Technol.*, vol. 6, no. 2, pp. 148–160, 2025, doi: 10.47065/bit.v5i2.2029.