Jurnal Informatika dan Tekonologi Komputer
E-ISSN: 2809-9230
P-ISSN: 2809-9249

Penerapan Algoritma Support Vector Machine dan XGBoost Dalam Mengklasifikasikan Sentimen Opini Publik Terhadap Aplikasi Uber

Rizky Rizaldi [1]*, M.Ridho [2], Arraihan Tahta Ainullah [3], Lusiana Efrizoni [4], Rahmaddeni [5], M.Fahrel Dea Putra [6]

[1] Universitas Sains dan Teknologi Indonesia, Pekanbaru Riau, e-mail : rrcode9@gmail.com
[2] Universitas Sains dan Teknologi Indonesia, Pekanbaru Riau, e-mail : 2210031802002@sar.ac.id
[3] Universitas Sains dan Teknologi Indonesia, Pekanbaru Riau, e-mail : arraihantahta1701@gmail.com
[4] Universitas Sains dan Teknologi Indonesia, Pekanbaru Riau, e-mail : lusiana@stmik-amik-riau.ac.id
[5] Universitas Sains dan Teknologi Indonesia, Pekanbaru Riau, e-mail : rahmaddeni@usti.ac.id
[6] Universitas Sains dan Teknologi Indonesia, Pekanbaru Riau, e-mail : farelpku1@gmail.com
* Corresponding Author : Rizky Rizaldi

Abstract: *The development of application-based transportation services such as Uber has driven an increase in the number of public opinions distributed through various digital platforms. Sentiment analysis of this public opinion is important to understand user perceptions of Uber services. This study applies the Support Vector Machine (SVM) and Extreme Gradient Boosting (XGBoost) algorithms to classify public opinion sentiment, by optimizing data imbalance using the Synthetic Minority Oversampling Technique (SMOTE). The data used comes from Uber reviews on public platforms, which are grouped into positive, negative, and neutral sentiments. The experimental results show that the SVM algorithm has superior performance with an accuracy of 94%, while XGBoost experienced an increase in accuracy of up to 93% after applying SMOTE. This study provides insight into the effectiveness of machine learning algorithms in sentiment analysis and its implementation in the development strategy of application-based transportation services.*

Keywords: *XGBoost; Support Vector Machine; Uber Reviews; SMOTE; NLP*

Abstrak: Perkembangan layanan transportasi berbasis aplikasi seperti Uber telah mendorong peningkatan jumlah opini publik yang disalurkan melalui berbagai platform digital. Analisis sentimen terhadap opini publik ini menjadi penting untuk memahami persepsi pengguna terhadap layanan Uber. Penelitian ini menerapkan algoritma Mesin Vektor Pendukung (SVM) dan Peningkatan Gradien Ekstrem (XGBoost) untuk mengklasifikasikan sentimen opini publik, dengan mengoptimalkan ketidakseimbangan data menggunakan *Synthetic Minority Oversampling Technique* (SMOTE). Data yang digunakan berasal dari ulasan Uber di platform publik, yang dikategorikan ke dalam sentimen positif, negatif, dan netral. Hasil eksperimen menunjukkan bahwa algoritma SVM memiliki performa lebih unggul dengan akurasi mencapai 94%, sementara XGBoost mengalami peningkatan akurasi hingga 93% setelah penerapan SMOTE. Penelitian ini memberikan wawasan mengenai efektivitas algoritma pembelajaran mesin dalam analisis sentimen serta implikasinya terhadap strategi pengembangan layanan transportasi berbasis aplikasi.

Kata kunci: XGBoost; Support Vector Machine; Ulasan Uber; SMOTE; NLP

Received: 22 Februari 2025
Revised: 8 Maret 2025
Accepted: 23 April 2025
Published: 28 April 2025
Curr. Ver.: 28 April 2025

1. Pendahuluan

Transportasi online dimulai oleh aplikasi Uber di Amerika Serikat pada tahun 2010 dan kini telah melayani lebih dari 10 milyar perjalanan [1]. Seiring berkembangnya teknologi, metode transportasi pun mulai berkembang. Saat ini masyarakat Indonesia tengah mengembangkan transportasi daring yang akan menjadi pilihan alternatif bagi masyarakat. Keberadaan transportasi daring menjadi isu kontrove rsial yang menimbulkan ketidak-konsistenan dalam pilihan transportasi masyarakat [2]. Munculnya Internet telah mempercepat perkembangan di bidang teknologi informasi. Hal ini dimanfaatkan oleh pelaku bisnis yang menawarkan jasa transportasi untuk mengembangkan usahanya. Sekarang ini umumnya disebut sebagai transportasi online [3].

Pada penelitian ini, proses klasifikasi diterapkan dengan memanfaatkan dua model algoritma yang berbeda, yaitu *Mesin Vektor Pendukung (SVM)* dan *Peningkatan Gradien Ekstrem (XGBoost)*. Kedua model tersebut tersebut dievaluasi dengan menerapkan teknik data mining dan pemrosesan bahasa alami (NLP). Selanjutnya, dilakukan penggabungan dengan fungsi seleksi, yang berperan penting dalam mengoptimalkan kinerja klasifikasi. Small menghasilkan sampel sintetis untuk kelas terkecil dengan menghapus setiap data titik dan memperhitungkan gabungan linier dari tetangga terdekat pada kelas terkecil yang sudah ada [4]. Hasil pengujian oleh [5] mengindikasikan bahwa metode ini efektif digunakan pada berbagai model prediktif. Di sisi lain, studi dalam [6] membahas perbandingan kinerja klasifikasi serta penerapan strategi SMOTE pada data tidak seimbang terkait kesalahan kartu kredit dengan menggunakan algoritma Random Forest dan XGBoost. Hasilnya menunjukkan bahwa Extreme Gradient Boosting (XGBoost) memiliki akurasi tertinggi dalam kombinasi dengan SMOTE, mencapai 72,78%. Sementara, Penelitian oleh [8] menunjukkan bahwa SVM memiliki akurasi tertinggi sebesar 83% dengan pemisahan data 90:10. Di sisi lain, XGBSVM menghasilkan akurasi 79% dengan pemisahan data 90:10. Selain itu, Penelitian oleh [9] Setelah menggunakan pemilihan fitur XGBoost menghasilkan akurasi sebesar 60,18%, 67,69%, 70,45%, dan 77,27%. Penelitian ini juga menghasilkan skor rata-rata tertinggi pada 10-Fold Cross-Validation pada percobaan kedua dengan skor sebesar 65,62%. Dalam penelitian [10] mereka memperoleh presisi 84,9% dan perolehan kembali 85,3% untuk tujuh moda transportasi, dengan menggunakan AdaBoost dan Decision Tree (klasifikasi dua tahap). Berdasarkan penelitian yang dilakukan oleh [11], hasil algoritma SVM dengan seleksi fitur menunjukkan nilai akurasi yang baik sebesar 62,3% saat menggunakan split data 80:20. Dalam Penelitian [12] Algoritma Extra-Tree dapat dianggap sebagai solusi terbaik: ditemukan menghasilkan kinerja terbaik dalam hal akurasi rata-rata (95,3%) dan memakan waktu.

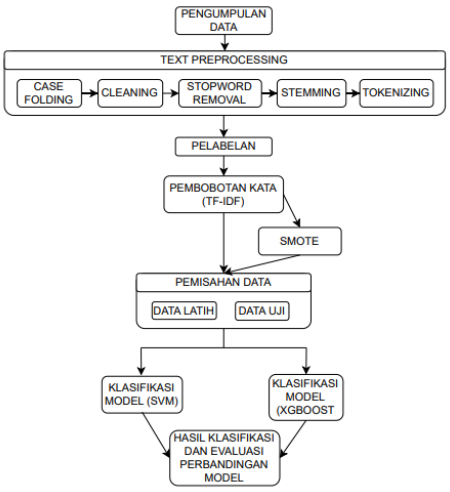
Studi sebelumnya menunjukkan bahwa penggunaan *Synthetic Minority Oversampling Technique* (SMOTE) pada algoritma *Mesin Vektor Pendukung (SVM)* dan *Peningkatan Gradien Ekstrem (XGBoost)* mampu meningkatkan kinerja model karena kesamaan karakteristik keduanya. Namun, sejauh ini belum ada penelitian yang secara khusus menganalisis perbedaan kedua model tersebut dalam penerapan SMOTE untuk menganalisis sentimen opini publik terhadap aplikasi Uber.

Penelitian ini bertujuan untuk menganalisis dan menganalisis perbedaan kinerja algoritma *Mesin Vektor Pendukung (SVM)* dan *Peningkatan Gradien Ekstrem (XGBoost)* dalam penerapan teknik SMOTE untuk mengklasifikasikan opini publik mengenai aplikasi Uber. Oleh karena itu, studi ini menyediakan pengetahuan yang lebih mendalam tentang efektivitas kedua model algoritma serta strategi yang digunakan untuk menangani ketidakseimbangan data. Perbandingan dilakukan berdasarkan hasil klasifikasi dan dievaluasi menggunakan metrik seperti *accuracy, precision, recall,* serta *F1-score.*

2. Metode yang Diusulkan

Penelitian ini menerapkan algoritma *Mesin Vektor Pendukung (SVM)* dan *Peningkatan Gradien Ekstrem (XGBoost)* untuk mengklasifikasikan sentimen opini publik terhadap aplikasi *Uber*, dengan menggunakan *Synthetic Minority Oversampling Technique* (SMOTE) guna mengatasi ketidakseimbangan data.

Ilustrasi 1 menunjukkan tahapan dalam penelitian ini, dimulai dari pengumpulan data, diikuti oleh tahap pra-pemrosesan data. Selanjutnya, dilakukan pelabelan data dan pemisahan data menjadi data training serta data testing. Pada tahap berikutnya, data training digunakan untuk membandingkan performa model klasifikasi dengan menerapkan teknik SMOTE oversampling. Selanjutnya, dilakukan proses evaluasi terhadap model. Penelitian ini diakhiri dengan penyimpulan berdasarkan hasil analisis perbedaan metode *Mesin Vektor Pendukung (SVM)* dan *Peningkatan Gradien Ekstrem (XGBoost).*



Ilustrasi 1. Alur Penelitian

2.1. Pengumpulan Data

Penelitian ini menggunakan data publik yang diperoleh dari Kaggle. Dataset tersebut berisi sekitar 2000 ulasan terkait aplikasi Uber, yang diperoleh melalui proses crawling data.

2.2. Text Preprocessing

Text preprocessing merupakan tahap krusial dalam analisis data yang bertujuan untuk membersihkan dan mentransformasikan data mentah agar sesuai untuk analisis lebih lanjut. Langkah ini bertujuan untuk meningkatkan kualitas data dengan menangani berbagai permasalahan seperti data yang hilang, pencilan, dan inkonsistensi [13]. Pada studi ini, proses text-preprocessing mencakup 5 tahap berikut:

1. *Case Folding* bertujuan untuk mengonversi semua huruf dalam teks menjadi huruf kecil, contohnya kata "Good" diubah menjadi "good" [14].
2. *Cleaning* adalah proses menghapus kata atau karakter yang tidak relevan dalam dokumen [15], contohnya tanda baca, tanda, tag mention, emotikon, tagar, dan tautan URL.
3. *Stopword Removal* digunakan untuk menghilangkan kata-kata umum yang tidak memiliki arti penting dalam analisis teks, seperti "dan", "di", atau "yang".
4. *Stemming* adalah proses pada mekanisme Information Retrieval (IR) yang mengubah kata-kata dalam dokumen menjadi bentuk dasarnya (kata dasar) mengikuti aturan tertentu [16].
5. *Tokenisasi* adalah proses memecah teks menjadi unit-unit kecil yang disebut token, seperti kata atau frasa individual [17].

2.3. Proses Pelabelan Data

Setelah melewati tahap *text preprocessing*, dataset yang sudah dibersihkan akan dilabeli pada setiap entri untuk mempermudah proses klasifikasi. Proses pelabelan dilakukan secara manual dengan tiga kategori, yaitu *Positive, Negative, dan Neutral.*

2.4. Pembobotan Kata

Pembobotan kata merupakan strategi praproses data dengan menetapkan bobot yang sesuai untuk setiap istilah sehingga bobot mewakili relevansi istilah dengan dokumen, model ini memainkan peran penting dalam meningkatkan kinerja klasifikasi teks[18]. Salah satu diantara metode penentuan bobot kata yang sering digunakan adalah *Frekuensi Term-Kebalikan Frekuensi Dokumen* (TF-IDF). Metode ini menggabungkan dua konsep perhitungan bobot, yaitu frekuensi kemunculan sebuah kata dalam dokumen dan kebalikan dari jumlah dokumen yang memuat kata tersebut. TF-IDF dimanfaatkan untuk menghitung bobot suatu istilah (term) t dalam dokumen d [19]. Berikut ini adalah persamaan penentuan bobot kata menggunakan TF-IDF:

$$W_{ij} = tf_{ij} \times \log(\text{ndf}) \quad (1)$$

Wij merupakan bobot yang menunjukkan tingkat kepentingan suatu term, *tfij* merupakan frekuensi kemunculan term dalam dokumen, dan *ndf* mengacu pada jumlah keseluruhan dokumen yang mengandung term tersebut dalam koleksi dokumen.

2.5. Pemisahan Data

Tahap *split* data atau pemisahan data ke dalam set pelatihan dan pengujian merupakan langkah krusial dalam pengembangan model Machine Learning. Dalam konteks ini, set pelatihan digunakan untuk melatih model agar dapat memahami pola-pola yang terdapat dalam data. Proporsi data yang dialokasikan ke dalam set pelatihan biasanya lebih besar, sekitar 70%, untuk memastikan model memiliki cukup informasi untuk belajar. Selama proses pelatihan, model akan menyesuaikan parameter internalnya berdasarkan data pelatihan sehingga dapat mempelajari hubungan antara fitur yang ada dalam data dan target output yang diinginkan[20]. Dalam penelitian ini, dilakukan pemisahan data dengan tiga rasio tidak sama, yaitu 70:30, 80:20, dan 90:10.

2.6. *Synthetic Minority Oversampling Technique* (SMOTE)

SMOTE (Synthetic Minority Oversampling Technique) adalah salah satu metode oversampling yang paling umum digunakan untuk menyelesaikan masalah ketidak seimbangan distribusi data pada pemodelan machine learning. SMOTE (Synthetic Minority Oversampling Technique) bertujuan untuk menyeimbangkan distribusi kelas dengan meningkatkan jumlah kelas minoritas secara acak dengan cara membuat data sintesis untuk tujuan oversampling [21]. SMOTE (Synthetic Minority Oversampling Technique) menghasilkan data pelatihan sintesis yang baru dengan menginterpolasi linier untuk kelas minoritas. Setelah proses oversampling, data direkonstruksi dan beberapa model klasifikasi dapat diterapkan untuk data yang sudah diproses.

2.7. *Peningkatan Gradient Extreme* (XGBoost)

XGBoost (Extreme Gradiant Boosting) merupakan tree bossting yang dapat diskalakan yang telah terbukti sebagai metode machine learning yang sangat efektif dan luas digunakan untuk melukan prediksi nilai dari suatu data. XGBoost menggunakan pohon keputusan sebagai model dasarnya dan menerapkan teknik penguatan untuk meningkatkan kinerja model. Sistem ini mencakup fitur-fitur regularisasi, parallel processing, dan handling missing values. Selain itu, XGBoost memperhatikan pola akses cache, kompresi data, dan sharding untuk membangun sistem penguatan pohon yang dapat diskalakan. Dengan menggabungkan berbagai teknik ini, XGBoost mampu menangani data besar menggunakan sumber daya yang lebih sedikit daripada sistem yang sudah ada [22].

Algortima XGBoost secara sederhana dituliskan pada Persamaan 1 sampai Persamaan 4 sebagai berikut. Integrasi model pohon dengan metode penjumlahan, diasumsikan jumlah dari K pohon, dan F menggambarkan model pohon dasar, maka :

$$\hat{y}_l = \sum_{k=1}^{K} f_k(xi), f_k \in F \quad (2)$$

Penjelasan:
K = Total decision tree
$f_k(xi)$ = Fungsi input pada decision tree ke-k
$\hat{y}_l$ = Hasil Prediksi
F = Kumpulan seluruh kemungkinan CART

2.8. *Mesin Vektor Pendukung* (SVM)

Mesin Vektor Pendukung (SVM) adalah metode pembelajaran terawasi yang bertujuan untuk mencari hyperplane yang dapat memisahkan dua atau lebih kategori data yang berlainan [23]. Prinsip utama SVM adalah membangun *hyperplane* dengan margin yang seimbang sehingga tidak terlalu dekat dengan salah satu kelas [24]. Berikut adalah rumus umum yang digunakan dalam perhitungan Support Vector Machine (SVM):

$$f(x) = sign\,(w.x + b) \quad (3)$$

Keterangan:
$F(x)$ = Fungsi perkiraan
w = Vektor normal hyperplane
x = Vektor fitur masukan(input)
b = bias atau intercept

2.9. Evaluasi Model

Langkah selanjutnya adalah mengevaluasi algoritma dengan confusion matrix, yang berfungsi untuk merepresentasikan akurasi hasil klasifikasi [25]. Confusion matrix digunakan untuk membandingkan kelas prediksi dengan kelas aktual pada data. Dalam penelitian ini, kinerja model dievaluasi dengan metrik *accuracy*, *recall, precision,* dan *F1-Score*. Contoh model confusion matrix ditampilkan pada tabel dibawah ini:

Tabel 1. ***Confusion matrix***

		Kelas Sebenarnya	
		True	***False***
Kelas	***True***	**TP**	**FP**
Prediksi	***False***	**FN**	**TN**

Keterangan:
True Positive (TP) = Jumlah dokumen dari kelas true yang benar diklasifikasikan sebagai kelas true.
True Negative (TN) = Jumlah dokumen yang berasal dari kelas true salah yang diklasifikasikan sebagai kelas false.
False Positive (FP) = Jumlah dokumen dari kelas false yang keliru diklasifikasikan sebagai kelas false.
False Negative (FN) = Total dokumen dari kategori true yang secara keliru dikelompokkan se bagai kategori false.

Rumus matematika untuk metrik *accuracy*, *precision*, *recall*, dan F1-Score ditampilkan di bawah ini:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Precision = \frac{TP}{TP + FN}$$

$$Recall = \frac{TP}{TP + FP}$$

$$F1 - score = 2 \times \frac{recall\ x\ precision}{recall + precision}$$

3. Hasil dan Pembahasan

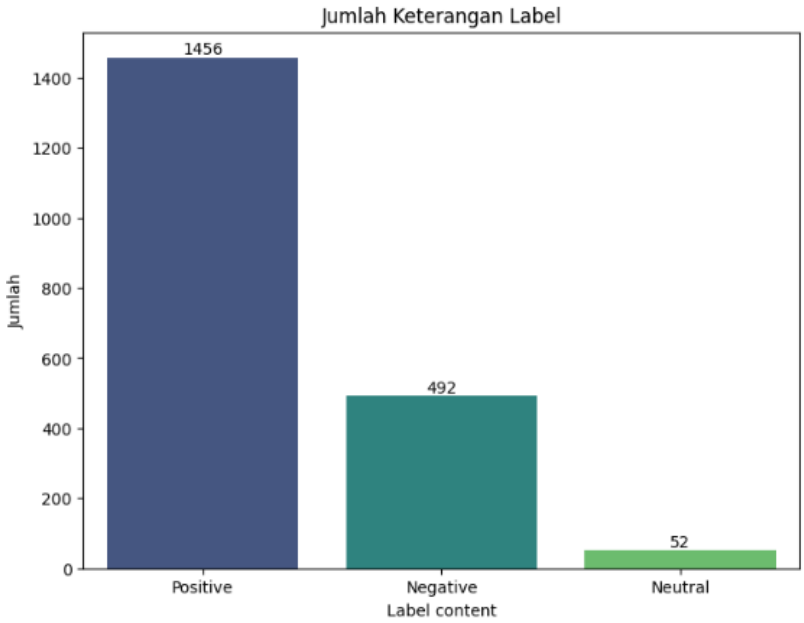
Dalam penelitian ini, Sebanyak 2000 data kotor berupa ulasan aplikasi Uber diproses melalui tahap preprocessing text untuk memastikan data lebih bersih dan akurat. Proses preprocessing text mencakup beberapa langkah, yaitu Case Folding, yang mengubah seluruh huruf menjadi huruf kecil; Cleaning, yang menghapus karakter tidak relevan seperti tanda baca, angka, tautan, dan simbol; Stopword Removal, yang menghilangkan kata-kata umum yang tidak memiliki makna signifikan seperti "dan", "atau", dan "yang"; Stemming, yang mengubah kata ke bentuk dasarnya, misalnya "bermain" menjadi "main"; serta Tokenization, yang memisahkan teks menjadi kata-kata individu untuk analisis lebih lanjut. Setelah proses preprocessing text selesai, jumlah data tetap sebanyak 2000, dengan pelabelan ke dalam tiga kategori sentimen, yaitu "Positive", "Neutral", dan "Negative".

Ilustrasi 2. Hasil *Text-Preprocessing* dan Pelabelan

	content	score	cleaned_content	stemming	tokenized	label
0	Good	5	Good	good	['good']	Positive
1	Nice	5	Nice	nice	['nice']	Positive
2	Very convenient	5	Very convenient	very convenient	['very', 'convenient']	Positive
3	Good	4	Good	good	['good']	Positive
4	exllence	5	exllence	exllence	['exllence']	Positive
...	...	...	...	...	...	...
1995	Bhuut bekaar app hai cab book kro to aati nhi ...	1	Bhuut bekaar app hai cab book kro to aati nhi ...	bhuut bekaar app hai cab book kro to aati nhi ...	['bhuut', 'bekaar', 'app', 'hai', 'cab', 'book...	Negative
1996	Not coming to the correct place	1	Not coming to the correct place	not coming to the correct place	['not', 'coming', 'to', 'the', 'correct', 'pla...	Negative
1997	app is good, but you have to wait for a long h...	2	app is good, but you have to wait for a long h...	app is good but you have to wait for a long ho...	['app', 'is', 'good', 'but', 'you', 'have', 't...	Negative
1998	Reasonable price and good service, quick confi...	5	Reasonable price and good service, quick confi...	reasonable price and good service quick confir...	['reasonable', 'price', 'and', 'good', 'servic...	Positive
1999	Super	5	Super	super	['super']	Positive

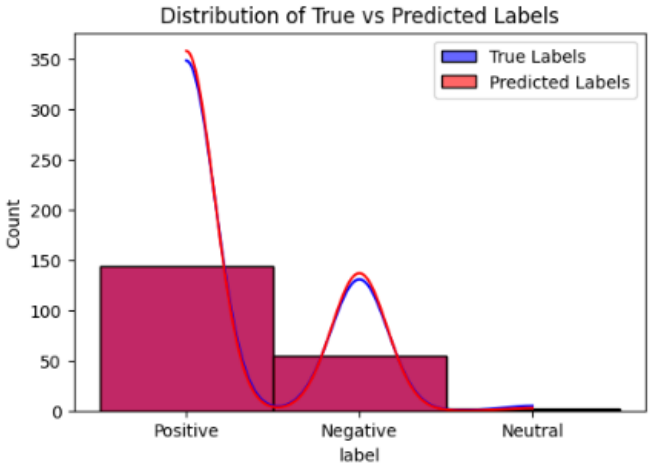
Setelah melalui tahap preprocessing text dan pelabelan data, dilakukan perhitungan jumlah label untuk menganalisis distribusi sentimen publik terhadap layanan aplikasi Uber. Penghitungan ini membantu memberikan gambaran seberapa banyak ulasan yang menunjukkan kepuasan, ketidakpuasan, atau sikap netral pengguna terhadap aplikasi tersebut.

Ilustrasi 3 mengilustrasikan bahwa sebagian besar ulasan mengenai aplikasi Uber memiliki sentimen positif, dengan total 1.456 ulasan, diikuti oleh sentimen negatif sebanyak 492 ulasan, dan sentimen netral sebanyak 52 ulasan. Hal ini mengindikasikan bahwa secara keseluruhan, pengguna cenderung memberikan tanggapan positif terhadap layanan Uber. Tahap berikutnya adalah mengimplementasikan data ke dalam berbagai model, yaitu SVM dan *SVM* dengan SMOTE. Selain itu, dilakukan analisis perbedaan dengan algoritma XGBoost dan XGBoost dengan SMOTE, menggunakan berbagai rasio pembagian data uji dan latih, yaitu 70:30, 80:20, dan 90:10. Dalam penelitian ini, metode pembobotan kata TF-IDF (Term Frequency-Inverse Document Frequency) digunakan untuk mengolah teks dan menentukan tingkat kepentingan setiap kata dalam proses klasifikasi. Visualisasi data menampilkan kata-kata yang muncul pada ulasan dengan sentimen positif, negatif, dan netral, di mana sebagian besar kata mencerminkan kepuasan pengguna terhadap layanan Uber.



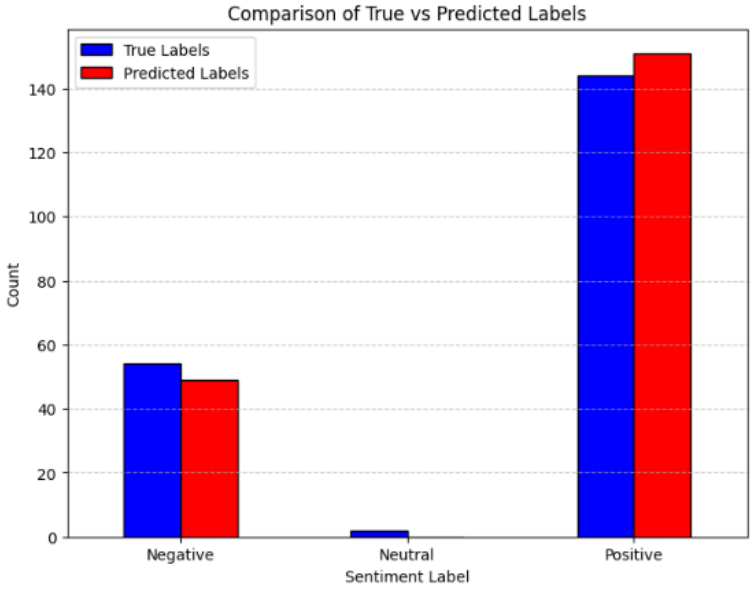
Ilustrasi 3. Hasil Jumlah Per Label

Ilustrasi 4 menampilkan hasil klasifikasi sentimen menggunakan algoritma **Support Vector Machine (SVM)** pada data ulasan Uber. Dengan berbagai pembagian data (70:30, 80:20, dan 90:10), model ini menunjukkan performa terbaiknya pada rasio **90:10**, dengan **144 ulasan positif, 54 ulasan negatif, dan 2 ulasan netral.** Secara keseluruhan, SVM mencapai akurasi **94%**, dengan presisi **93%**, recall **94%**, dan F1-score **93%**, membuktikan kemampuannya dalam mengklasifikasikan opini publik secara efektif.



Ilustrasi 4. Hasil Klasifikasi SVM

Ilustrasi 5 memperlihatkan hasil klasifikasi menggunakan **Extreme Gradient Boosting (XGBoost)** untuk membedakan sentimen positif, negatif, dan netral dalam ulasan pengguna Uber. Pada rasio **90:10**, model ini mengklasifikasikan **144 ulasan positif, 54 ulasan negatif, dan 2 ulasan netral**. Setelah penerapan **SMOTE,** akurasi meningkat hingga **93%**, hampir menyamai performa SVM. Hasil ini menunjukkan bahwa **teknik oversampling dapat meningkatkan efektivitas XGBoost,** terutama dalam menangani ketidakseimbangan data.



Ilustrasi 5. Hasil Klasifikasi XGBoost

fase terakhir pada studi ini adalah mengevaluasi model serta hasil klasifikasinya. Evaluasi ini bertujuan untuk memastikan bahwa model yang dikembangkan telah sesuai dengan persyaratan dan sasaran yang sudah ditentukan.

Tabel 2. *Hasil Klasifikasi*

Model	Positive			Negative			Neutral		
	70:30	80:20	90:10	70:30	80:20	90:10	70:30	80:20	90:10
SVM	449	296	144	139	98	54	12	6	2
SVM-SMOTE	449	296	144	139	98	54	12	6	2
XGBoost	449	296	144	139	98	54	12	6	2
XGBoost-SMOTE	449	296	144	139	98	54	12	6	2

Tabel 2 menunjukkan jumlah ulasan yang diklasifikasikan sebagai sentimen **positif, negatif, dan netral** oleh masing-masing model (SVM dan XGBoost) dengan berbagai pembagian data uji. Hasilnya menunjukkan bahwa **SVM dan XGBoost memiliki performa yang seragam**, dengan jumlah klasifikasi yang sama pada setiap skenario. **Mayoritas ulasan tergolong positif**, sedangkan sentimen netral paling sedikit. Penerapan **SMOTE tidak mengubah jumlah klasifikasi**, tetapi membantu model menangani ketidakseimbangan data.

Tabel 3. *Laporan* Evaluasi Model

Splitting	Model	Accuracy	Precision	Recall	F1-Score
90:10	SVM	94%	93%	94%	93%
	SVM-SMOTE	94%	93%	94%	93%
	XGBoost	91%	90%	91%	90%
	XGBoost-SMOTE	94%	93%	94%	93%
80:20	SVM	93%	92%	93%	93%
	SVM-SMOTE	91%	91%	91%	91%
	XGBoost	92%	91%	92%	91%
	XGBoost-SMOTE	90%	90%	90%	90%
70:30	SVM	92%	90%	92%	91%

SVM-SMOTE	**90%**	**90%**	**90%**	**90%**
XGBoost	**88%**	**86%**	**88%**	**87%**
XGBoost-SMOTE	**89%**	**87%**	**89%**	**88%**

Tabel 3 merangkum performa model berdasarkan ***akurasi, presisi, recall, dan F1-score***. **SVM terbukti unggul dengan akurasi tertinggi mencapai 94% pada rasio data 90:10**, menunjukkan kemampuannya dalam mengklasifikasikan opini publik dengan sangat baik. **XGBoost mengalami peningkatan setelah menggunakan SMOTE**, sehingga akurasinya hampir menyamai SVM. Pada pembagian data yang lebih kecil (70:30 dan 80:20), **SVM tetap lebih stabil dibandingkan XGBoost**, membuktikan bahwa model ini lebih tahan terhadap perubahan jumlah data uji. **Performa terbaik ditemukan pada skenario 90:10,** di mana semua model mencapai akurasi optimal. Secara keseluruhan, **SVM** menjadi pilihan terbaik untuk klasifikasi sentimen Uber, **sementara SMOTE membantu meningkatkan performa XGBoost agar lebih kompetitif**.

4. Kesimpulan

Penelitian ini membahas penerapan algoritma *Mesin Vektor Pendukung (*SVM*)* dan *Peningkatan Gradien Ekstrem (*XGBoost*)* dalam mengklasifikasikan sentimen opini publik terhadap aplikasi Uber. Berdasarkan hasil analisis, mayoritas ulasan aplikasi Uber memiliki sentimen positif, diikuti oleh sentimen negatif dan netral.

Hasil evaluasi menunjukkan bahwa algoritma SVM memiliki performa terbaik, dengan akurasi, presisi, recall, dan F1-score masing-masing sebesar 94%, 93%, 94%, dan 93% pada pembagian data 90:10, tanpa memerlukan teknik SMOTE. Sementara itu, algoritma XGBoost mengalami peningkatan performa setelah penerapan SMOTE, tetapi tetap tidak melampaui kinerja SVM. Teknik SMOTE terbukti efektif untuk meningkatkan performa XGBoost, tetapi tidak memberikan dampak yang signifikan pada SVM.

Secara keseluruhan, penelitian ini membuktikan bahwa SVM lebih unggul dalam klasifikasi sentimen ulasan Uber dibandingkan dengan XGBoost, terutama dalam kondisi dataset yang telah diproses dengan baik. Ke depan, penelitian ini dapat dikembangkan lebih lanjut dengan mencoba kombinasi algoritma lain atau menerapkan pendekatan deep learning untuk meningkatkan akurasi klasifikasi sentimen.

Daftar Pustaka

[1] T. Online and D. Indonesia, "Tranpostasi ojek online," pp. 5–14, 2019.

[2] M. A. K. Sugianto, "Tingkat Ketertarikan Masyarakat Terhadap Transportasi Online, Angkutan Pribadi Dan Angkutan Umum Berdasarkan Persepsi," vol. 1, no. 2, pp. 51–58, 2020.

[3] A. Apriliani, M. Budhiluhoer, A. Jamaludin, and K. Prihandani, "Systematic Literature Review Kepuasan Pelanggan terhadap Jasa Transportasi Online," *Systematics*, vol. 2, no. 1, p. 12, 2020, doi: 10.35706/sys.v2i1.3530.

[4] V. P. K. Turlapati and M. R. Prusty, "Outlier-SMOTE: A refined oversampling technique for improved detection of COVID-19," *Intell. Med.*, vol. 3–4, no. July, p. 100023, 2020, doi: 10.1016/j.ibmed.2020.100023.

[5] S. D. A. Bujang *et al.*, "Multiclass Prediction Model for Student Grade Prediction Using Machine Learning," *IEEE Access*, vol. 9, pp. 95608–95621, 2021, doi: 10.1109/ACCESS.2021.3093563.

[6] S. F. N. Halim and U. Azmi, "Analisis Perbandingan Klasifikasi dan Penerapan Teknik SMOTE Dalam Imbalanced Data Pada Credit Card Default," *J. Sains dan Seni ITS*, vol. 12, no. 2, 2023, doi: 10.12962/j23373520.v12i2.111833.

[7] S. Rabbani, D. Safitri, N. Rahmadhani, and M. K. Anam, "Perbandingan Evaluasi Kernel SVM untuk Klasifikasi Sentimen dalam Analisis Kenaikan Harga BBM: Comparative Evaluation of SVM Kernels for Sentiment Classification in Fuel Price Increase Analysis," *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 3, no. 2, pp. 153–160, 2023.

[8] R. Rahmaddeni, M. K. Anam, Y. Irawan, S. Susanti, and M. Jamaris, "Comparison of Support Vector Machine and XGBSVM in Analyzing Public Opinion on Covid-19 Vaccination," *Ilk. J. Ilm.*, vol. 14, no. 1, pp. 32–38, 2022, doi: 10.33096/ilkom.v14i1.1090.32-38.

[9] R. S. Putra, W. Agustin, M. K. Anam, L. Lusiana, and S. Yaakub, "The Application of Naïve Bayes Classifier Based Feature Selection on Analysis of Online Learning Sentiment in Online Media," *J. Transform.*, vol. 20, no. 1, p. 44, 2022, doi: 10.26623/transformatika.v20i1.5144.

[10] S. Hemminki, P. Nurmi, and S. Tarkoma, "Accelerometer-based transportation mode detection on smartphones," *SenSys 2013 - Proc. 11th ACM Conf. Embed. Networked Sens. Syst.*, 2013, doi: 10.1145/2517351.2517367.

[11] D. Septhya *et al.*, "Implementasi Algoritma Decision Tree dan Support Vector Machine untuk Klasifikasi Penyakit Kanker Paru," *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 3, no. 1, pp. 15–19, 2023, doi: 10.57152/malcom.v3i1.591.

[12] C. Badii, A. Difino, P. Nesi, I. Paoli, and M. Paolucci, "Classification of users' transportation modalities from mobiles in real operating conditions," *Multimed. Tools Appl.*, vol. 81, no. 1, pp. 115–140, 2022, doi: 10.1007/s11042-021-10993-y.

[13] I. M. Hamdani[1] *et al.*, "INTISARI Jurnal Inovasi Pengabdian Masyarakat Edukasi dan Pelatihan Data Science dan Data Preprocessing," *Juni*, vol. 2, no. 1, pp. 19–26, 2024, doi: 10.58227/intisari.v2i1.125.

[14] M. P. Pulungan, A. Purnomo, and A. Kurniasih, "Penerapan SMOTE untuk Mengatasi Imbalance Class dalam Klasifikasi Kepribadian MBTI Menggunakan Naive Bayes Classifier," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 10, no. 7, pp. 1493–1502, 2023, doi: 10.25126/jtiik.1077989.

[15] D. Mualfah, W. Fadila, and R. Firdaus, "Teknik SMOTE untuk Mengatasi Imbalance Data pada Deteksi Penyakit Stroke Menggunakan Algoritma Random Forest," *J. CoSciTech (Computer Sci. Inf. Technol.*, vol. 3, no. 2, pp. 107–113, 2022, doi: 10.37859/coscitech.v3i2.3912.

[16] M. A. Rayadin, M. Musaruddin, R. A. Saputra, and I. Isnawaty, "Implementasi Ensemble Learning Metode XGBoost dan Random Forest untuk Prediksi Waktu Penggantian Baterai Aki," *BIOS J. Teknol. Inf. dan Rekayasa Komput.*, vol. 5, no. 2, pp. 111–119, 2024.

[17] R. A. Rizal, I. S. Girsang, and S. A. Prasetiyo, "Klasifikasi Wajah Menggunakan Support Vector Machine (SVM)," *REMIK (Riset dan E-Jurnal Manaj. Inform. Komputer)*, vol. 3, no. 2, p. 1, 2019, doi: 10.33395/remik.v3i2.10080.

[18] M. D. Rahman, A. Djunaidy, and F. Mahananto, "Penerapan Weighted Word Embedding pada Pengklasifikasian Teks Berbasis Recurrent Neural Network untuk Layanan Pengaduan Perusahaan Transportasi," *J. Sains dan Seni ITS*, vol. 10, no. 1, 2021, doi: 10.12962/j23373520.v10i1.56145.

[19] A. Deolika, K. Kusrini, and E. T. Luthfi, "Analisis Pembobotan Kata Pada Klasifikasi Text Mining," *J. Teknol. Inf.*, vol. 3, no. 2, p. 179, 2019, doi: 10.36294/jurti.v3i2.1077.

[20] H. Tao *et al.*, "Training and Testing Data Division Influence on Hybrid Machine Learning Model Process: Application of River Flow Forecasting," *Complexity*, vol. 2020, 2020, doi: 10.1155/2020/8844367.

[21] E. Erlin, Y. Desnelita, N. Nasution, L. Suryati, and F. Zoromi, "Dampak SMOTE terhadap Kinerja Random Forest Classifier berdasarkan Data Tidak seimbang," *MATRIK J. Manajemen, Tek. Inform. dan Rekayasa Komput.*, vol. 21, no. 3, pp. 677–690, 2022, doi: 10.30812/matrik.v21i3.1726.

[22] S. Usman, "Predictive Sparepart Maintenance Menggunakan Algoritma Machine Learning Extreme Gradiant Boosting Regressor," *J. Syst. Comput. Eng.*, vol. 5, no. 2, pp. 249–258, 2024, doi: 10.61628/jsce.v5i2.1418.

[23] P. K. Handayani, "Penerapan Algoritma Support Vector Machine (Svm) Untuk Analisis Pola Klasifikasi Pada Parkinson'S Dataset," *Indones. J. Technol. Informatics Sci.*, vol. 3, no. 1, pp. 31–35, 2021, doi: 10.24176/ijtis.v3i1.7530.

[24] W. Zhong and L. Du, "Predicting Traffic Casualties Using Support Vector Machines with Heuristic Algorithms: A Study Based on Collision Data of Urban Roads," *Sustain.*, vol. 15, no. 4, 2023, doi: 10.3390/su15042944.

[25] T. Gori, A. Sunyoto, and H. Al Fatta, "Preprocessing Data dan Klasifikasi untuk Prediksi Kinerja Akademik Siswa," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 11, no. 1, pp. 215–224, 2024, doi: 10.25126/jtiik.20241118074.