

Klasifikasi Sentimen Ulasan Aplikasi Aci Menggunakan N-Gram Dan Algoritma Support Vector Machine (SVM)

Sentiment Classification of Aci Application Reviews Using N-Gram Features And Support Vector Machine (SVM) Algorithm

Ageng Wijaya Kusuma¹, Moh. Dasuki^{*2}, Wiwik Suharso³

^{1,2}Program Studi Teknik Informatika, Fakultas Teknik, Universitas Muhammadiyah Jember

³Program Studi Sistem Informasi, Fakultas Teknik, Universitas Muhammadiyah Jember

Email: ¹agengwijaya1717@gmail.com, ²moh.dasuki22@unmuhjember.ac.id, ³wiwiksuharso01@gmail.com

^{*}Penulis Koresponden

Received: 16 Januari 2026

Accepted: 27 Januari 2026

Published: 02 Februari 2026



This work is licensed under a [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/).
Copyright (c) 2026 JUSTINDO

ABSTRAK

Transformasi teknologi informasi membuka peluang besar dalam penerapan pemrosesan bahasa alami (*Natural Language Processing*) untuk analisis sentimen teks, khususnya dalam menggali opini pengguna terhadap layanan berbasis aplikasi. Penelitian ini bertujuan untuk menganalisis sentimen ulasan pengguna aplikasi ojek daring ACI (Aku Cinta Indonesia) yang tersedia di *Google Play Store* dengan menerapkan metode *N-gram* dan algoritma *Support Vector Machine* (SVM). Data yang digunakan berjumlah 1.419 ulasan, yang setelah melalui tahap pengolahan data dan pelabelan sentimen berbasis *lexicon-based* menghasilkan 239 data akhir dengan kategori sentimen positif dan negatif. Ekstraksi fitur dilakukan menggunakan kombinasi *unigram*, *unigram + bigram*, dan *unigram + trigram*, dengan pembobotan kata menggunakan *Term Frequency–Inverse Document Frequency* (TF-IDF). Selanjutnya, proses klasifikasi dilakukan menggunakan *Support Vector Machine* (SVM) dengan kernel linear melalui pembagian data latih dan data uji dengan rasio 80:20. Hasil pengujian menunjukkan bahwa *unigram + bigram* memperoleh akurasi tertinggi sebesar 96%, diikuti oleh *unigram + trigram* sebesar 94%, dan *unigram* sebesar 90%, dengan seluruh nilai *precision*, *recall*, dan *F1-score* pada ketiga model berada di atas 88%. Hasil ini menunjukkan bahwa kombinasi *unigram + bigram* mampu merepresentasikan konteks kata secara lebih optimal dibandingkan *unigram*, namun tetap lebih efisien daripada *unigram + trigram*, sehingga meningkatkan akurasi klasifikasi sentimen oleh model SVM tanpa menambah kompleksitas secara signifikan.

Kata kunci: Analisis Sentimen, N-Gram, Support Vector Machine

ABSTRACT

The transformation of information technology has created significant opportunities for the application of *Natural Language Processing* (NLP) in text-based sentiment analysis, particularly in exploring user opinions toward application-based services. This study aims to analyze the sentiment of user reviews of the ACI (Aku Cinta Indonesia) online motorcycle taxi application available on the *Google Play Store* by applying the *N-gram* method and the *Support Vector Machine* (SVM) algorithm. A total of 1,419 reviews were collected, and after data preprocessing and *lexicon-based* sentiment labeling, 239 final samples were obtained and categorized into positive and negative sentiments. Feature extraction was performed using combinations of *unigram*, *unigram + bigram*, and *unigram + trigram*, with *Term Frequency–Inverse Document Frequency* (TF-IDF) weighting. Furthermore, the classification process was carried out using a linear kernel *Support Vector Machine* with an 80:20 split between training and testing data. The experimental results show that the *unigram+bigram* model achieved the highest accuracy of 96%, followed by *unigram + trigram* at 94% and *unigram* at 90%, with all *precision*, *recall*, and *F1-score* values across the three models exceeding 88%. These findings indicate that the *unigram + bigram* combination represents word context more effectively than *unigram* while remaining more efficient than *unigram + trigram*, thereby improving the sentiment classification accuracy of the SVM model without significantly increasing computational complexity.

Keywords: Sentiment Analysis, N-Gram, Support Vector Machine.

1. Pendahuluan

Perkembangan teknologi informasi yang pesat telah mendorong peningkatan dalam bidang pemrosesan bahasa alami (*Natural Language Processing*), terutama dalam analisis sentimen berbasis teks. Salah satu teknik yang krusial dalam ekstraksi fitur dari data teks adalah *N-gram*, yaitu metode statistika yang digunakan untuk membentuk rangkaian kata sebanyak n berdasarkan urutan kemunculannya dalam kalimat atau dokumen (Dhinora and Mailoa, 2025). *N-gram* dapat berupa *unigram* (satu kata), *bigram* (dua kata), *trigram* (tiga kata) dan seterusnya (Mantik *et al.*, 2022). Dengan pendekatan ini, relasi antar kata dalam sebuah konteks menjadi lebih terlihat sehingga mampu menangkap makna yang lebih kompleks dibandingkan representasi kata tunggal. Dalam analisis sentimen, teknik *N-gram* memegang peranan penting karena dapat mendeteksi pola-pola kalimat yang mengindikasikan sentimen tertentu, baik positif maupun negatif, secara lebih presisi. Dengan penerapan *N-gram*, informasi kontekstual antar kata dapat lebih mudah dipetakan sehingga mendukung proses klasifikasi, khususnya dalam analisis sentimen (Nurhidayat and Dewi, 2023).

Analisis sentimen dilakukan terhadap ulasan aplikasi ojek daring yang tersedia di *Google Play Store*. Aplikasi yang menjadi objek pada penelitian ini adalah ACI, yaitu aplikasi layanan transportasi daring asli karya anak bangsa Indonesia. ACI merupakan singkatan dari "Aku Cinta Indonesia", yang tidak hanya berfungsi sebagai aplikasi ojek *online*, tetapi juga mengusung semangat nasionalisme dan pemberdayaan ekonomi lokal. Secara singkat, ACI dapat didefinisikan sebagai aplikasi transportasi daring berbasis lokal yang menyediakan layanan ojek, pengantaran makanan, kurir, dan layanan lainnya. Seiring dengan meningkatnya penggunaan layanan transportasi berbasis aplikasi, ulasan dan opini pengguna menjadi indikator penting dalam menilai kualitas layanan yang diberikan (Sahabuddin *et al.*, 2025). Oleh karena itu, untuk memperkuat pemahaman terhadap definisi tersebut, dilakukan analisis teks menggunakan metode *N-gram* untuk mengidentifikasi pola kata yang dominan dalam ulasan aplikasi ACI.

Dalam penelitian ini, pendekatan yang dipilih adalah menggunakan metode *Support Vector Machine* (SVM). *Support Vector Machine* (SVM) ditemukan oleh Vladimir N. Vapnik dan Alexey Ya. Chervonenkis pada tahun 1963, sebagai bagian dari pengembangan teori *statistical learning*. Pada awalnya, SVM digunakan untuk menyelesaikan masalah klasifikasi linier sederhana. Namun, seiring berkembangnya teknologi dan kebutuhan akan metode klasifikasi yang lebih kompleks, SVM mengalami berbagai penyempurnaan, terutama melalui pengenalan konsep kernel pada tahun 1992 yang memungkinkan algoritma ini digunakan untuk klasifikasi data *non-linier* (Iqbal, Afdal and Novita, 2024). SVM banyak digunakan karena efektif dalam menyelesaikan masalah klasifikasi, regresi, dan prediksi pada data linier maupun *non-linier* dalam jumlah besar (Mukhtar *et al.*, 2022).

Support Vector Machine (SVM) adalah algoritma pembelajaran mesin yang berfungsi untuk melakukan klasifikasi maupun regresi dengan cara mencari sebuah *hyperplane* atau garis pemisah terbaik yang mampu memisahkan data dari dua kelas secara optimal (Pratama, Triawan and Artikel, 2025). Algoritma ini memaksimalkan margin antara dua kelas data, sehingga meningkatkan kemampuan generalisasi model terhadap data baru. Titik data yang paling berpengaruh dalam pembentukan *hyperplane* disebut sebagai *support vectors*. SVM sangat efektif dalam menangani data berdimensi tinggi dan sering digunakan dalam analisis sentimen karena kemampuannya dalam mengolah data teks, terutama ketika dikombinasikan dengan teknik representasi fitur seperti *N-gram* (Br Sinulingga and Sitorus, 2024).

Beberapa penelitian terdahulu telah mengkaji penggunaan metode klasifikasi dalam analisis sentimen dengan SVM. Salah satu penelitian berjudul "Analisis Sentimen Review Skincare Skintific dengan Algoritma Support Vector Machine (SVM)". Sebanyak 958 ulasan dari hasil penelitian menunjukkan bahwa model SVM mampu mencapai akurasi sebesar 94%, dengan nilai precision, recall, dan f1-score yang masing-masing melebihi 0,9. Dari hasil klasifikasi, ditemukan sebanyak 862 ulasan dengan sentimen positif dan 96 ulasan dengan sentimen negatif.

2. Metode Penelitian

Metode penelitian disusun secara terstruktur untuk memastikan penelitian berjalan sesuai tujuan, mencakup proses. Tahapan penelitian ini menjelaskan proses yang dilakukan secara sistematis, dimulai dari pengumpulan data, pengolahan data, penerapan metode yang digunakan, hingga evaluasi terhadap hasil penelitian yang diperoleh.



Gambar 1. Alur Penelitian

2.1. Pengumpulan Data

Tahapan ini dilakukan proses pengumpulan data dengan mengambil data berupa ulasan pengguna aplikasi ACI yang tersedia pada platform *Google Play Store*. Pengambilan data dilakukan menggunakan metode *web scraping*, yaitu teknik ekstraksi data secara otomatis dari halaman web, yang dijalankan melalui *Google Colaboratory (Google Colab)*.

2.2. Pengolahan Data

Tahap ini bertujuan untuk mengolah data teks hasil *scraping* dari aplikasi ACI agar siap digunakan dalam proses pelatihan dan pengujian model. Proses yang dilakukan meliputi pengolahan data teks, yaitu *cleansing*, *normalization*, *stopword removal*, serta *stemming* menggunakan pustaka Sastrawi. Tahap *preprocessing* dilakukan untuk membersihkan dan menyiapkan teks sehingga dapat diproses lebih lanjut secara optimal dengan mengurangi unsur *noise* dalam data (Nurlaely, Sartika Simatupang and Lucia Kharisma, 2023). Setelah itu, dilakukan proses tokenisasi berbasis *N-Gram* dengan kombinasi *unigram*, *unigram + bigram*, dan *unigram + trigram* untuk membentuk fitur teks. Tahapan ini menghasilkan data teks yang bersih, terstruktur, dan siap digunakan pada tahap klasifikasi sentimen.

2.3. Penerapan Metode

Tahap ini bertujuan untuk membangun model klasifikasi teks berdasarkan data yang telah melalui proses *pre-processing*. Pelabelan data dilakukan menggunakan pendekatan *lexicon-based*, di mana setiap ulasan diberi label sentimen positif atau negatif. Data kemudian dibagi menjadi data latih dan data uji menggunakan rasio tertentu. Setelah itu, dilakukan proses pembobotan kata menggunakan metode *Term Frequency–Inverse Document Frequency (TF-IDF)* untuk merepresentasikan teks dalam bentuk numerik (Rahayu *et al.*, 2022). Hasil pembobotan TF-IDF digunakan sebagai vektor fitur yang merepresentasikan setiap ulasan pengguna (Apriliyanti *et al.*, 2026). Model klasifikasi selanjutnya dibangun menggunakan *Support Vector Machine (SVM)* dengan kernel linear, yang digunakan untuk mengklasifikasikan ulasan ke dalam kelas sentimen positif dan negatif. SVM linear dipilih karena mampu bekerja secara efektif pada data teks serta memiliki kinerja yang baik dalam memanfaatkan fitur berbasis frekuensi maupun pembobotan TF-IDF.

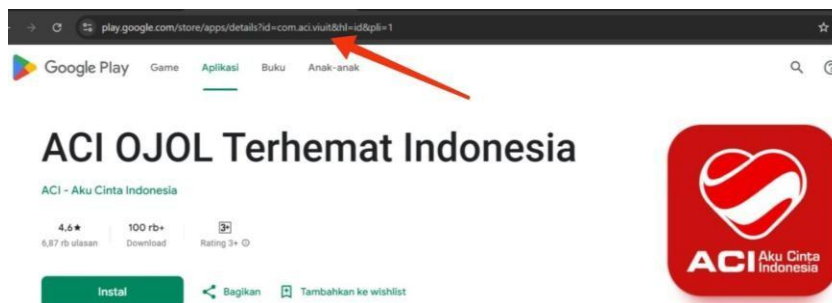
2.4. Evaluasi

Tahap evaluasi menggunakan *Confusion Matrix* dilakukan pada data uji untuk menilai kinerja model SVM. Tahapan ini bertujuan untuk mengukur sejauh mana model mampu mengklasifikasikan data sentimen secara tepat. Metrik performa seperti *accuracy*, *precision*, *recall*, dan *F1-score* dihitung sebagai bagian dari proses evaluasi untuk mengetahui kualitas hasil klasifikasi. Setelah itu, dilakukan visualisasi dalam bentuk *wordcloud* untuk mengetahui kata-kata yang paling sering muncul dan dominan pada masing-masing kelas sentimen (positif dan negatif), sehingga dapat memberikan gambaran interpretatif terhadap hasil analisis sentimen yang telah dilakukan.

3. Hasil dan Pembahasan

3.1. Pengumpulan Data

Data yang digunakan dalam penelitian ini berasal dari ulasan pengguna aplikasi ACI yang diperoleh dari platform Google Play Store. Pengambilan data dilakukan menggunakan metode *web scraping*, yaitu teknik ekstraksi data secara otomatis dari halaman web, dengan bantuan *Google Colaboratory (Google Colab)* sebagai lingkungan pemrograman. Data ulasan diperoleh dengan menggunakan fungsi *scraper* yang mengambil input ID aplikasi “com.aci.viuit”, yaitu identitas unik aplikasi ACI pada *Google Play Store*. ID ini digunakan untuk mengakses halaman aplikasi yang sesuai sehingga ulasan yang dikumpulkan berasal langsung dari aplikasi ACI dan bersifat valid. Melalui proses tersebut, berhasil dikumpulkan sebanyak 1.419 ulasan.



Gambar 2. Url Id Aplikasi ACI

	userName	score	at	content
0	Izza Isma	5	2025-12-16 13:17:09	sengaja sy kasi bintang 5 biar g d take down,hal...
1	eko eko	3	2025-12-16 05:19:01	apk nya bagus cuma ga bisa diakses di Jawa Ten...
2	Sabrina Auzora Labibah	5	2025-12-16 03:43:49	permisi, maaf ini saya baru pertama kali make ...
3	umi asfiah	5	2025-12-15 19:51:42	cakep driver respon sangat baik cepat tanggap
4	Moch Noer Rizki	5	2025-12-15 16:29:11	bagus. promo banyak. hemat

Gambar 3. Screenshot Hasil *scraping* ulasan ACI

3.2. Pengolahan Data

Tahap pengolahan data teks diawali dengan proses *cleansing*, yaitu menghilangkan karakter yang tidak relevan seperti tanda baca, angka, simbol, serta emotikon yang tidak memiliki pengaruh terhadap analisis sentimen. Proses ini bertujuan untuk mengurangi noise pada data teks sehingga hanya informasi yang relevan yang dipertahankan.

Tabel 1. Hasil *Cleansing*

Sebelum	Sesudah
Aplikasinya ini sgt bagus!	aplikasinya ini sgt bagus
Aplikasinya duhh, sgt buruk!!!	aplikasinya duhh sgt buruk

Selanjutnya dilakukan *normalization*, yaitu memperbaiki kata tidak baku atau singkatan agar sesuai dengan bentuk kata yang baku. Langkah ini dilakukan untuk mengurangi variasi kata dan membantu model menghasilkan klasifikasi yang lebih tepat.

Tabel 2. Hasil *Normalization*

Sebelum	Sesudah
aplikasinya ini sgt bagus	aplikasinya ini sangat bagus
aplikasinya duhh sgt buruk	aplikasinya duhh sangat buruk

Setelah itu, dilakukan *stopword removal* untuk menghapus kata-kata umum yang sering muncul namun tidak memiliki makna sentimen yang signifikan, seperti kata hubung dan kata depan, sehingga fitur yang dihasilkan menjadi lebih representatif.

Tabel 3. Hasil *Stopword Removal*

Sebelum	Sesudah
aplikasinya ini sangat bagus	aplikasinya sangat bagus
aplikasinya duhh sangat buruk	aplikasinya sangat buruk

Tahap selanjutnya adalah *stemming*, yaitu proses mengonversi kata berimbuhan menjadi kata dasar dengan memanfaatkan pustaka Sastrawi. Proses ini bertujuan untuk meningkatkan konsistensi data, mengurangi jumlah variasi kata, serta mendukung agar proses klasifikasi teks dapat berlangsung secara lebih efektif dan optimal.

Tabel 4. Hasil *Stemming*

Sebelum	Sesudah
aplikasinya sangat bagus	aplikasi sangat bagus
aplikasinya sangat buruk	aplikasi sangat buruk

Setelah proses *stemming* selesai, dilakukan *tokenizing* untuk memecah teks menjadi unit-unit kata yang lebih kecil sebagai fitur masukan model. Pada penelitian ini, proses tokenisasi dilakukan menggunakan pendekatan *N-Gram*, dengan kombinasi *unigram*, *unigram + bigram*, dan *unigram + trigram*, guna menangkap pola kata tunggal maupun hubungan antar kata dalam teks.

Tabel 5. Hasil *Tokenizing*

N-gram	sesudah
Unigram	['aplikasi', 'sangat', 'bagus']
Unigram + Bigram	['aplikasi', 'sangat', 'bagus', 'aplikasi sangat', 'sangat bagus']
Unigram + Trigram	['aplikasi', 'sangat', 'bagus', 'aplikasi sangat bagus']

3.3. Penerapan Metode

Data hasil pengolahan selanjutnya dilakukan proses pelabelan, yaitu dengan memberikan label positif atau negatif pada setiap ulasan menggunakan pendekatan *lexicon-based*. Ulasan dengan label netral dihapus karena tidak memiliki polaritas sentimen yang jelas serta berpotensi menimbulkan ambiguitas dalam proses klasifikasi. Setelah proses pelabelan diterapkan pada fitur *unigram*, *unigram + bigram*, dan *unigram + trigram*, masing-masing menghasilkan 239 data, yang terdiri atas 159 ulasan berkategori positif dan 80 ulasan berkategori negatif.

TF-IDF dimanfaatkan untuk mengubah data teks menjadi vektor numerik dengan memberikan bobot pada setiap kata berdasarkan frekuensi kemunculan dan tingkat kepentingannya. Data yang telah direpresentasikan dalam bentuk vektor selanjutnya dibagi menjadi data latih dan data uji dengan rasio 80:20 untuk keperluan pelatihan dan evaluasi model. Setelah itu, *Support Vector Classifier* (SVC) dengan kernel linear diterapkan sebagai algoritma klasifikasi untuk memisahkan data ke dalam kelas sentimen positif dan negatif berdasarkan nilai fungsi keputusan. Model SVM menggunakan kernel linear karena representasi TF-IDF berbentuk *sparse matrix* yang umumnya lebih efektif dipisahkan secara linear. Persamaan (1) merupakan perhitungan *term frequency (TF)*, nilai *t* merepresentasikan jumlah kemunculan suatu kata tertentu dalam dokumen *d*, sedangkan *d* mengacu pada total keseluruhan kata dalam dokumen tersebut.

$$TF = \frac{t}{d} \tag{1}$$

Pada persamaan (2), *N* mewakili jumlah total dokumen dalam korpus, sedangkan *df(t)* menunjukkan jumlah dokumen yang mengandung kata *t*.

$$idf = \log \frac{N}{df(t)} \tag{2}$$

Pada persamaan (3), Nilai TF-IDF dari suatu kata dihitung berdasarkan kombinasi dari kedua nilai tersebut.

$$TFidf = TF \cdot idf \tag{3}$$

Tabel 6. Contoh Dokumen Uji

Sebelum	Sesudah
---------	---------

D1	aplikasi, sangat, bagus, aplikasi sangat, sangat bagus
D2	aplikasi, sangat, buruk, aplikasi sangat, sangat buruk

Tabel 7. Perhitungan TF-IDF

Term	TF	df(t)	IDF = log(2 / df)	TF-IDF
aplikasi	1/5 = 0,2	2	log(2/2) = 0	0
sangat	1/5 = 0,2	2	log(2/2) = 0	0
bagus	1/5 = 0,2	1	log(2/1) = 0.301	0.060206
buruk	1/5 = 0,2	1	log(2/1) = 0.301	0.060206
aplikasi sangat	1/5 = 0,2	2	log(2/2) = 0	0
sangat bagus	1/5 = 0,2	1	log(2/1) = 0.301	0.060206
sangat buruk	1/5 = 0,2	1	log(2/1) = 0.301	0.060206

Pada persamaan (4), Di mana $f(x)$ merupakan fungsi prediksi, w adalah vektor normal yang menentukan arah *hyperplane*, x merupakan vektor fitur input, dan b adalah bias.

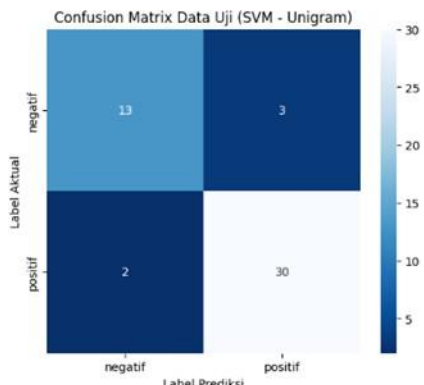
$$f(x) = w \cdot x + b \tag{4}$$

Tabel 8. Perhitungan SVM

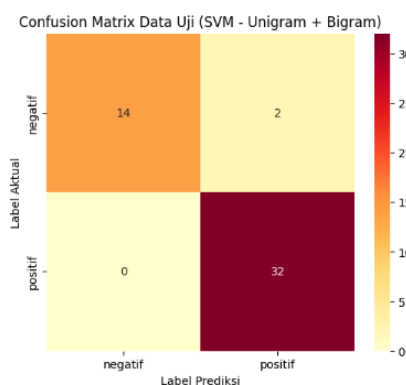
Dokumen	Term Aktif (TF- IDF)	Bobot	Perhitungan $f(x)$	Nilai $f(x)$
D1	0.060206, 0.060206	1,1	$(1 \times 0.060206) + (1 \times 0.060206) + 0$	0.120412
D2	0.060206, 0.060206	-1, -1	$(-1 \times 0.060206) + (-1 \times 0.060206) + 0$	-0.120412

3.4. Evaluasi

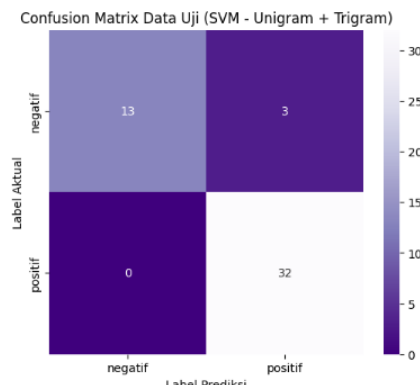
Evaluasi menggunakan *Confusion Matrix* dilakukan untuk menilai kinerja model SVM. Tahapan ini bertujuan mengukur sejauh mana model mampu mengklasifikasikan data dengan tepat. Metrik performa seperti *accuracy*, *precision*, *recall*, dan *f1-score* dihitung sebagai bagian dari proses evaluasi.



Gambar 4. Hasil *Confusion Matrix Unigram*



Gambar 5. Hasil *Confusion Matrix Unigram + Bigram*



Gambar 6. Hasil *Confusion Matrix Unigram + Trigram*

Tabel 8. Perhitungan SVM

Model	Accuracy	Precision	Recall	F1-Score
Unigram	0.90	0.89	0.88	0.88
Unigram + Bigram	0.96	0.97	0.94	0.96
Unigram + Trigram	0.94	0.96	0.91	0.93

Model SVM dengan fitur *unigram* memperoleh nilai *accuracy* 0,90, *precision* 0,89, *recall* 0,88, dan *F1-score* 0,88, yang menandakan bahwa penggunaan kata tunggal sudah mampu menangkap sentimen secara umum namun masih memiliki keterbatasan konteks. Model *unigram + bigram* menunjukkan performa terbaik dengan *accuracy* 0,96, *precision* 0,97, *recall* 0,94, dan *F1-score* 0,96, yang mencerminkan keseimbangan optimal antara ketepatan dan sensitivitas dalam mengenali sentimen. Sementara itu, model *unigram + trigram* menghasilkan *accuracy* 0,94, *precision* 0,96, *recall* 0,91, dan *F1-score* 0,93, yang menunjukkan peningkatan dibandingkan *unigram*, tetapi tidak

melampaui kinerja *unigram* + *bigram* karena kompleksitas fitur *trigram* cenderung menambah redundansi tanpa peningkatan performa yang signifikan.



Gambar 7. Wordcloud Sentimen Positif dan Negatif Unigram



Gambar 8. Wordcloud Sentimen Positif dan Negatif Unigram + Bigram



Gambar 9. Wordcloud Sentimen Positif dan Negatif Unigram + Trigram

Topik-topik yang sering dibahas oleh pengguna aplikasi ACI di *Google Play Store* ditunjukkan pada tahap interpretasi hasil. Topik ulasan dengan sentimen positif maupun negatif kemudian divisualisasikan dalam bentuk *wordcloud*. *Wordcloud* positif menunjukkan kata ‘murah’, ‘cepat’, dan ‘mudah’ sebagai dominan, yang mengindikasikan kepuasan terhadap harga dan layanan pengemudi. Sementara pada *wordcloud* negatif, kata ‘error’, ‘lemot’, dan ‘susah’ muncul paling sering, menandakan masalah pada sisi teknis aplikasi.

4. Kesimpulan

Pengujian menggunakan metode klasifikasi SVM pada ulasan aplikasi ACI menunjukkan bahwa kombinasi *unigram* + *bigram* menghasilkan akurasi tertinggi sebesar 96%, diikuti oleh *unigram* + *trigram* dengan akurasi 94%, dan *unigram* dengan akurasi 90%, di mana seluruh nilai *precision*, *recall*, dan *F1-score* masing-masing model berada di atas 88%. Hasil ini menunjukkan bahwa kombinasi *unigram* + *bigram* mampu menangkap konteks kata yang lebih luas dibandingkan *unigram*, namun tetap lebih efisien dibandingkan kombinasi *unigram* + *trigram*, sehingga meningkatkan kemampuan model SVM dalam membedakan sentimen positif dan negatif secara lebih akurat tanpa menambah kompleksitas model secara berlebihan.

Daftar Pustaka

Apriliyanti, P.N., Dasuki, M. and Rahman, M. (2026) ‘Klasifikasi Sentimen Positif dan Negatif Ulasan Aplikasi GetContact Dengan Algoritma Naïve Bayes’, JUSTIFY : Jurnal Sistem Informasi Ibrahimy, 4(2), pp. 123–129. doi:10.35316/justify.v4i2.9133.

Br Sinulingga, J.E. and Sitorus, H.C.K. (2024) ‘Analisis Sentimen Opini Masyarakat terhadap Film Horor Indonesia Menggunakan Metode SVM dan TF-IDF’, Jurnal Manajemen Informatika (JAMIKA), 14(1), pp. 42–53. Available at: <https://doi.org/10.34010/jamika.v14i1.11946>.

Dhinora, M.Y. and Mailoa, E. (2025) ‘Analisa Tweet Mahasiswa untuk Deteksi Gejala Depresi dengan Penerapan Natural Language Processing’, Jurnal Indonesia : Manajemen Informatika dan Komunikasi, 6(2), pp. 1193–1211. Available at: <https://doi.org/10.63447/jimik.v6i2.1405>.

Iqbal, M., Afdal, M. and Novita, R. (2024) ‘Implementasi Algoritma Support Vector Machine Untuk Analisa Sentimen Data Ulasan Aplikasi Pinjaman Online di Google Play Store’, MALCOM: Indonesian Journal of Machine Learning and Computer Science, 4(4), pp. 1244–1252. Available at: <https://doi.org/10.57152/malcom.v4i4.1435>.

Mantik, J. et al. (2022) Application Of N-Gram On K-Nearest Neighbor Algorithm To Sentiment Analysis Of TikTok Shop Shopping Features, Jurnal Mantik. Online.

Mukhtar, H. et al. (2022) ‘Peramalan Kedatangan Wisatawan ke Suatu Negara Menggunakan Metode Support Vector Machine (SVM)’, Jurnal CoSciTech (Computer Science and

- Information Technology), 3(3), pp. 274–282. Available at: <https://doi.org/10.37859/coscitech.v3i3.4211>.
- Nurhidayat, R. and Dewi, K.E. (2023) 'KOMPUTA : Jurnal Ilmiah Komputer dan Informatika PENERAPAN ALGORITMA K-NEAREST NEIGHBOR DAN FITUR EKSTRAKSI N-GRAM DALAM ANALISIS SENTIMEN BERBASIS ASPEK', 12(1). Available at: <https://www.kaggle.com/datasets/hafidahmusthaanah/skincare-review?select=00.+Review.csv>.
- Nurlaely, R., Sartika Simatupang, D. and Lucia Kharisma, I. (2023) 'Analisis Sentimen Twitter Terhadap Cyberbullying Menggunakan Metode Support Vector Machine (SVM)', 4(2), pp. 376–384. Available at: <https://doi.org/10.37859/coscitech.v4i2.5161>.
- Pratama, S., Triawan, M. and Artikel, R. (2025) Klasifikasi Sentimen Komentar pada Video 'Rendang Hilang di Palembang' oleh Willy Salim Menggunakan Algoritma Support Vector Machine (SVM) INFORMASI ARTIKEL ABSTRAK. Available at: <https://www.ejournal.lembahdempo.ac.id/index.php/ITBis-SISKOMTI>.
- Rahayu, S. et al. (2022) 'Implementasi Metode K-Nearest Neighbor (K-NN) untuk Analisis Sentimen Kepuasan Pengguna Aplikasi Teknologi Finansial FLIP', *Edumatic: Jurnal Pendidikan Informatika*, 6(1), pp. 98–106. Available at: <https://doi.org/10.29408/edumatic.v6i1.5433>.
- Sahabuddin, R. et al. (2025) Peran Kepercayaan sebagai Variabel Intervening dalam Pengaruh Kualitas Layanan dan Kepuasan Pelanggan terhadap Loyalitas Pelanggan: Studi pada Perusahaan E-Commerce Shopee dengan Pendekatan Structural Equation Modeling (SEM).