DEWANTARA. J. Tech., Vol. 07, No. 01 (2026), Hal. 01 - 17
E ISSN : 2776-6764
P ISSN : 2774-2032

Analisis Sentimen Ulasan Pengguna Aplikasi Kredivo menggunakan Algoritma Naive Bayes dan Logistic Regression

Diki Dwi Anggara-1[a*], Farid Fitriyadi-2[a], Dahlan Susilo-3[a]

[a]Prodi Informatika, Universitas Sahid Surakarta,
Jl. Adi Sucipto No.154, Jajar, Kec. Laweyan, Kota Surakarta, Jawa Tengah, Indonesia

*Email : anggaradiki0403@gmail.com

Abstrak

Perkembangan pesat industri *Financial Technology* (Fintech) *Buy Now Pay Later* (BNPL) di Indonesia mendorong tingginya akumulasi ulasan pengguna pada aplikasi Kredivo di Google Play Store. Ulasan tersebut memuat data opini yang sangat berharga terkait tingkat kepuasan, keandalan sistem, maupun keluhan layanan, namun volumenya yang besar menyulitkan evaluasi secara manual. Penelitian ini bertujuan untuk mengidentifikasi sentimen publik serta membandingkan performa algoritma Multinomial *Naive Bayes* dan *Logistic Regression* dalam mengklasifikasikan sentimen ulasan pengguna Kredivo. Dataset sebanyak 17.212 ulasan terfilter periode Januari 2021 hingga Januari 2026 diproses melalui tahapan *preprocessing* (*cleansing*, *case folding*, *tokenizing*, normalisasi kata Indo-Normalizer, *stopword removal*, dan *stemming sastrawi*) hingga menghasilkan 10.798 data ulasan bersih. Pelabelan otomatis berbasis rating dengan verifikasi manual pada rating 3 menghasilkan distribusi sentimen yang tidak seimbang, yaitu 80,55% positif dan 19,45% negatif. Pembobotan kata menggunakan *Term Frequency-Inverse Document Frequency* (TF-IDF) menghasilkan 4.558 fitur kata. Untuk mengatasi ketidakseimbangan kelas tanpa memicu data leakage, teknik *Synthetic Minority Oversampling Technique* (SMOTE) diterapkan secara terisolasi pada data latih di setiap fold menggunakan 5-Fold *Cross Validation*. Hasil eksperimen menunjukkan *Logistic Regression* unggul secara konsisten dibanding *Naive Bayes* dengan menggapai rata-rata akurasi sebesar 90,14%; *precision* 91,43%; *recall* 90,14%; dan F1-score 90,52% pada 5-Fold *Cross Validation*. Pada pengujian akhir menggunakan 20% data uji murni (2.160 ulasan), *Logistic Regression* kembali membuktikan daya generalisasi terbaik tanpa *overfitting* dengan akurasi 90,42% dan F1-score 90,74%. Analisis *Word Cloud* mengungkapkan bahwa kepuasan pengguna didominasi oleh efisiensi pencairan dana dan kemudahan aplikasi, sedangkan keluhan terfokus pada penolakan limit transaksi mendadak serta mekanisme penagihan.

Kata Kunci : *Analisis Sentimen, Kredivo, Naive Bayes, Logistic Regression, Fintech*

1. Latar Belakang

Perkembangan dibidang teknologi informasi telah mendorong adanya perubahan digital di berbagai industri, termasuk dalam sektor keuangan melalui layanan *Financial Technology* (FinTech). Pertumbuhan ini didorong oleh meningkatnya akses internet serta penerimaan layanan keuangan digital oleh masyarakat di Indonesia. Hingga bulan Februari 2026, tercatat ada 95 penyedia Layanan Pendanaan Bersama Berbasis Teknologi Informasi (LPBBTI) yang sudah mendapatkan izin dari Otoritas Jasa Keuangan (OJK), sedangkan jumlah pengguna internet di Indonesia mencapai sekitar 235 juta orang atau 81,72% dari total populasi [1], [2]. Situasi ini

menunjukkan bahwa masyarakat semakin mengandalkan layanan keuangan digital untuk memenuhi kebutuhan transaksi sehari-hari.

Salah satu layanan teknologi keuangan yang mengalami kemajuan pesat adalah *Buy Now Pay Later* (BNPL). Sistem ini memudahkan pengguna untuk melakukan pembelian terlebih dahulu dan membayarnya kemudian dengan cara mencicil melalui berbagai pilihan jangka waktu. Di tanah air, Kredivo menjadi salah satu penyedia BNPL yang populer karena menyuguhkan berbagai fitur seperti cicilan, pinjaman tunai, pembayaran tagihan, serta pembiayaan untuk transaksi di berbagai *platform e-commerce*. Berdasarkan informasi dari *Google Play Store*, aplikasi Kredivo telah diinstal lebih dari 50 juta kali, mendapatkan lebih dari 4,77 juta ulasan, dan memperoleh rating rata-rata 4,8 dari 5. Banyaknya pengguna dan ulasan ini memperlihatkan tingkat penggunaan aplikasi yang sangat tinggi dan sekaligus menghasilkan data opini yang signifikan dan bernilai untuk menilai kualitas layanan yang disediakan [3].

Ulasan pengguna merupakan bentuk umpan balik (*feedback*) langsung yang merefleksikan pengalaman nyata terkait keandalan sistem, layanan pelanggan, proses verifikasi limit, hingga skema biaya [4]. Namun, melakukan evaluasi secara manual terhadap ulasan di dataset besar ternyata tidak efisien, memakan waktu lama, dan mudah terpengaruh oleh bias dari sudut pandang pengamat. Karena itu, dibutuhkan metode otomatisasi yang didasarkan pada *Natural Language Processing* (NLP) dan *text mining* dengan menggunakan analisis sentimen untuk mengelompokkan pandangan pengguna ke dalam kategori sentimen yang positif dan negatif dengan cara yang objektif dan tepat [5].

Analisis sentimen adalah salah satu contoh penerapan teknologi *Natural Language Processing* (NLP) dan *text mining* yang digunakan untuk mengetahui pendapat pengguna tentang produk atau layanan, apakah bersifat positif atau negatif. Dengan cara mengolah data teks yang benar, data yang awalnya tidak terorganisir bisa diubah menjadi data yang siap digunakan oleh algoritma klasifikasi, sehingga menghasilkan informasi yang lebih tepat [6], [7]. Karena itu, analisis sentimen kerap dimanfaatkan untuk mengetahui pandangan pengguna tentang aplikasi digital serta mendukung penilaian kualitas layanan.

Dalam ranah *machine learning*, berbagai algoritma klasifikasi telah diterapkan untuk analisis sentimen, di antaranya *Naive Bayes* dan *Logistic Regression. Naive Bayes* merupakan algoritma probabilistik yang efisien secara komputasi dan sangat tangguh dalam menangani fitur teks berdimensi tinggi. Di sisi lain, *Logistic Regression* merupakan model klasifikasi linear yang mampu mengukur hubungan *log-odds* antara fitur kata dan kelas target, sehingga memiliki daya generalisasi yang sangat kuat terhadap data baru. Sejumlah studi empiris terdahulu menunjukkan performa unggul *Logistic Regression* dibanding *Naive Bayes* pada data ulasan digital. Studi pada 200.000 ulasan mobile banking menunjukkan Logistic Regression mencapai akurasi 92%–93%, jauh melampaui Naive Bayes yang hanya memperoleh 71%–74% [8]. Temuan serupa pada ulasan produk skincare Tokopedia *Logistic Regression* mendapatkan 92,3% dan *Naive Bayes* 75,9 [9], serta pada ulasan *game Free Fire Logistic Regression* mendapat 81,63% dan *Naive Bayes* 74,20%[10].

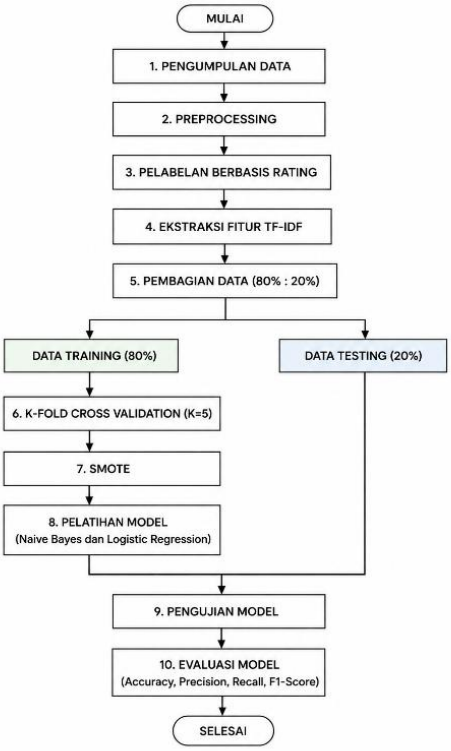
Meskipun begitu, pencarian di literatur menunjukkan adanya beberapa kekurangan dalam penelitian yang cukup penting. Pertama, penelitian tentang analisis sentimen yang membandingkan secara khusus dalam konteks aplikasi BNPL Kredivo masih jarang ditemukan. Kedua, sebagian besar penelitian terdahulu mengandalkan pelabelan berbasis kamus (*lexicon-based*) atau pembagian biner sederhana yang mengabaikan sifat ambigu pada ulasan rating sedang. Ketiga, belum

ada alur komprehensif yang mengintegrasikan pembersihan data teks bahasa Indonesia tidak baku menggunakan Indo-Normalizer, ekstraksi fitur TF-IDF, validasi silang K-Fold *Cross Validation*, serta penanganan ketidakseimbangan kelas (*class imbalance*) menggunakan *Synthetic Minority Oversampling Technique* (SMOTE) secara terisolasi pada setiap fold data latih.

Untuk melengkapi bidang penelitian tersebut, studi ini menawarkan inovasi berupa penggunaan proses klasifikasi sentimen yang menyeluruh dan sistematis pada 17.212 ulasan pengguna Kredivo periode Januari 2021 hingga Januari 2026. Penelitian ini mengombinasikan pelabelan berbasis rating dengan verifikasi manual pada rating 3, normalisasi istilah gaul/singkatan dengan Indo-Normalizer, stemming Sastrawi, pembobotan TF-IDF, *Splitting* Data 80:20, 5-Fold Cross Validation, penyeimbangan kelas SMOTE per fold, serta evaluasi komparatif antara *Naïve Bayes* dan *Logistic Regression*. Selain itu, visualisasi word cloud diterapkan untuk mengekstrak kata-kata kunci utama pada masing-masing kelompok sentimen. Riset ini diharapkan dapat menyajikan kontribusi teoritis dalam pengembangan skema NLP bahasa Indonesia sekaligus sumbangan praktis bagi pengembang Kredivo dalam mengakselerasi kualitas layanan dan kepuasan pengguna.

2. Metodologi

Riset ini memakai pendekatan kuantitatif komparatif dengan cara eksperimen berbasis komputer (*computer-based experiment*). Strategi kuantitatif dipilih karena penelitian ini menjalankan pengolahan dan kajian data secara terukur menggunakan teknik machine learning, sedangkan pendekatan komparatif dipakai untuk membandingkan performa algoritma *Naive Bayes* dan *Logistic Regression* dalam mengklasifikasikan sentimen ulasan pengguna aplikasi Kredivo. Eksperimen dijalankan melalui tahapan pengumpulan data, *preprocessing*, pelabelan berbasis rating, ekstraksi fitur menggunakan TF-IDF, pembagian data, K-Fold Cross Validation, penyeimbangan data menggunakan SMOTE, pelatihan model, pengujian model, serta penilaian menggunakan metrik accuracy, precision, recall, dan F1-score. Alur penelitian tampak jelas pada tahapan Gambar 1.



Gambar 1. Alur Penelitian

Pengumpulan data dijalankan memakai cara scraping dengan bantuan pustaka *Google Play Scraper* pada bahasa pemrograman *Python* melalui platform *Google Colaboratory* (*Google Colab*). Pengambilan data awal dilakukan sebanyak 50.000 data ulasan, kemudian data difokuskan pada ulasan pengguna periode Januari 2021 sampai Januari 2026 akibatnya didapatkan sebanyak 17.212 data ulasan yang dipakai sebagai dataset riset. Jenis data yang digunakan merupakan data sekunder karena diperoleh dari platform digital yang telah tersedia dan bukan melalui pengumpulan data secara langsung.

Preprocessing adalah langkah awal untuk mengolah data yang bertujuan mengubah data mentah yang didapat dari

banyak sumber menjadi informasi yang lebih teratur dan siap untuk dianalisis lebih lanjut [11]. Pada tahap pemrosesan, ada beberapa langkah yang harus dilakukan sebagai berikut:

1. *Cleaning*: Menghilangkan karakter selain teks seperti alamat website, angka, tanda baca, simbol, emoji, spasi yang terlalu banyak, serta menghapus data yang sama atau data yang tidak ada.
2. *Case Folding*: Mengubah semua huruf menjadi huruf kecil agar bentuk kata menjadi seragam.
3. *Tokenizing*: Memisahkan kalimat atau rangkaian teks menjadi bagian-bagian kata satu per satu (*token*).
4. Normalisasi Kata: Mengubah kata-kata yang tidak baku, singkatan, istilah gaul di internet, dan kesalahan pengetikan menjadi kata-kata yang sesuai dengan bahasa Indonesia menggunakan pustaka *Indo-Normalizer* [12].
5. *Stopword Removal*: Menghapus kata-kata yang sering muncul namun tidak memberikan informasi penting (seperti *"dan"*, *"yang"*, *"di"*, *"ke"*) menggunakan pustaka Sastrawi.
6. *Stemming*: Menghilangkan imbuhan (awalan, sisipan, akhiran) pada kata untuk mengembalikannya ke bentuk kata dasar menggunakan algoritma *Enhanced Confix Stripping* (ECS) pada pustaka Sastrawi[13].

Pelabelan dilakukan menggunakan pendekatan berbasis rating (rating-based labeling). yaitu metode otomatis yang memanfaatkan nilai ulasan atau penilaian yang diberikan pengguna [14]. Pendekatan ini dipilih karena penilaian yang diberikan oleh pengguna di *Google Play Store* bisa menunjukkan bagaimana mereka menilai sebuah aplikasi. Selain itu, metode ini lebih efisien dibandingkan pelabelan manual terhadap seluruh data, terutama pada dataset yang berjumlah besar.

Ulasan dengan rating 1 dan 2 secara otomatis diberi label sentimen negatif (0) karena umumnya berisi keluhan, kritik, atau pengalaman yang kurang memuaskan. Di sisi lain, ulasan yang mendapat nilai 4 dan 5 dikategorikan sebagai positif karena biasanya menunjukkan kepuasan pengguna terhadap layanan yang diterima.

Sementara, ulasan dengan rating 3 tidak langsung diberikan label karena bersifat netral atau ambigu. Meskipun memiliki nilai rating yang sama, isi ulasan dapat mengandung sentimen positif maupun negatif. Oleh karena itu, dilakukan verifikasi manual oleh peneliti dengan membaca isi ulasan untuk menentukan kategori sentimen yang paling sesuai.

TF-IDF (*Term Frequency-Inverse Document Frequency*) adalah suatu hubungan kata (*term*) untuk memberi nilai pada kata dalam dokumen. Metode ini menggabungkan dua hal untuk menghitung nilai kata, yaitu seberapa sering kata muncul dalam dokumen tertentu yang disebut *term frequency* (TF) dan seberapa jarang dokumen lain menggunakan kata tersebut yang disebut *inverse document frequency* (IDF). Seberapa sering kata muncul dalam dokumen menunjukkan betapa pentingnya kata itu di dalam dokumen. Oleh karena itu, nilai hubungan antara kata dan dokumen akan tinggi jika kata itu sering muncul di dokumen dan dokumen lain yang menggunakan kata yang sama jumlahnya sedikit di seluruh koleksi dokumen. [15].

Dataset yang telah direpresentasikan menggunakan TF-IDF kemudian dipisahkan menjadi dua bagian, yaitu 80% untuk data pelatihan (*training*) dan 20% untuk data pengujian (*testing*). Pembagian data bertujuan agar model bisa dilatih menggunakan sebagian besar data, sedangkan sebagian data lainnya digunakan sebagai data pengujian yang benar-benar baru. Pembagian 80:20 ini adalah salah satu cara yang sering dipakai dalam penelitian machine learning karena memberikan keseimbangan yang baik antara pelatihan dan evaluasi model [16].

Setelah dilakukan pembagian data menjadi 80% data training dan 20% data testing, tahap berikutnya adalah menerapkan metode K-Fold *Cross*

Validation pada data *training*. Tujuan dari metode ini adalah untuk memperoleh model klasifikasi yang lebih stabil, mengurangi risiko *overfitting*, serta memastikan bahwa model memiliki kemampuan generalisasi yang baik terhadap data baru [17].

Dalam penelitian ini, menggunakan metode 5-*Fold Cross Validation* (K = 5). Data latih (*training*) dibagi menjadi lima bagian (*fold*) yang jumlahnya hampir sama. Di setiap iterasi, satu *fold* dipakai sebagai data validasi, sementara empat *fold* lainnya digunakan untuk data pelatihan. Proses ini dilakukan berulang hingga semua fold telah digunakan untuk data validasi sebanyak satu kali. Dengan menggunaan *K-Fold Cross Validatio*n, setiap data *training set* dapat digunakan baik sebagai data pelatihan ataupun data validasi, ehingga hasil dari penilaian model menjadi lebih mencerminkan keadaan sebenarnya dibandingkan jika hanya menggunakan satu pembagian data.

Distribusi data sentimen pada penelitian ini tidak seimbang karena terdapat lebih banyak ulasan positif dibandingkan negatif. Ketidakseimbangan ini bisa membuat model lebih fokus pada kelas yang lebih banyak, sehingga kesulitan dalam mengenali kelas yang lebih sedikit. Untuk mengatasi masalah ini, digunakan metode *Synthetic Minority Oversampling Technique* (SMOTE). Teknik ini menciptakan data sintetis untuk kelas yang lebih sedikit agar distribusi data jadi lebih seimbang [18]. SMOTE hanya digunakan pada data latihan di setiap *fold* agar tidak terjadi kebocoran data dan menjaga keadilan dalam evaluasi model.

Naive Bayes adalah metode sederhana untuk memprediksi kemungkinan berdasarkan teorema Bayes dengan asumsi bahwa semua fitur bersifat independen. Salah satu keuntungan dari teknik ini adalah hanya membutuhkan sedikit data latihan untuk menentukan parameter yang dibutuhkan saat mengklasifikasikan data [19]. Rumus teorema bayes adalah sebagai berikut:

$$P(C \mid X) = \frac{P(X|C)\cdot P(C)}{P(X)} \quad (1)$$

Keterangan:
1. P(C|X) : Kemungkinan kelas C muncul jika sudah memiliki bukti X.
2. P(X|C) : Kemungkinan bukti X terjadi jika kelas C benar.
3. P(C) : Kemungkinan awal kelas C tanpa mempertimbangkan buktinya.
4. P(X) : Kemungkinan semua bukti X secara umum.

Logistic Regression adalah jenis analisis regresi yang menjelaskan hubungan antara variabel yang tergantung dan yang tidak tergantung. Ini menjelaskan hubungan antara satu atau lebih variabel yang bebas dan variabel yang terikat, jenisnya bisa berupa kategori tertentu, yang nilainya bisa antara 0 dan 1, serta bisa benar atau salah, besar atau kecil. Itulah sebabnya Regresi Logistik berbeda dari regresi berganda atau regresi linear [20]. Rumus fungsi *sigmoid* pada Regresi Logistik dapat ditulis seperti berikut:

$$P(y = 1 \mid x) = \frac{1}{1+e^{-z}} \quad (2)$$

Dengan nilai:

$$z = w_0 + w_1x_1 + w_2x_2 + \cdots + w_nx_n \quad (3)$$

Keterangan:
1. P(y=1|x) : kemungkinan data masuk ke kelas positif.
2. e : angka eksponensial.
3. z : kombinasi linear dari fitur yang dimasukkan.
4. w : bobot model parameter.
5. x : fitur data yang dihasilkan dari pembobotan TF-IDF.

Setelah pelatihan selesai, model yang terbaik dari setiap algoritma akan diuji dengan menggunakan data uji (*testing*) sebesar 20% yang tidak pernah dipakai sebelumnya dalam pelatihan atau validasi. Tujuan penggunaan data uji yang terpisah adalah untuk melihat seberapa baik model dapat mengklasifikasikan data baru. Selain itu, memisahkan data uji juga bertujuan

untuk mencegah terjadinya kebocoran data sehingga hasil evaluasi model lebih adil [21].

Tahap evaluasi dilakukan dengan menggunakan *confusion matrix* yang berfungsi untuk menilai seberapa baik kinerja model klasifikasi, dengan membandingkan hasil prediksi model dengan data sebenarnya dalam dataset pengujian menggunakan metrik *accuracy, precision, recall*, dan F1-score.

1. *Accuracy* digunakan untuk menilai seberapa akurat klasifikasi secara keseluruhan.
2. *Precision* mengukur seberapa tepat model dalam memprediksi suatu kategori.
3. Recall mengukur seberapa baik model dalam menemukan semua data yang termasuk dalam kategori tertentu.
4. *F1-score* adalah nilai rata-rata antara precision dan recall yang memberikan gambaran kinerja model secara lebih seimbang.

3. Hasil dan Pembahasan

Pengumpulan dan Penyaringan Data

Pengumpulan data ulasan pengguna aplikasi Kredivo dilakukan secara otomatis melalui teknik *web scraping* menggunakan pustaka *google-play-scraper* pada Google Colaboratory. Data hasil *scraping* selanjutnya disimpan dalam bentuk *DataFrame* menggunakan pustaka pandas untuk memudahkan proses pengelolaan dan analisis data. Sebanyak 50.000 ulasan berhasil dikumpulkan pada tahap awal. Atribut yang digunakan dalam analisis meliputi *content* sebagai isi ulasan, *score* sebagai rating pengguna dengan rentang 1-5, serta *at* sebagai informasi waktu publikasi ulasan.

Tahap berikutnya dilakukan penyaringan data berdasarkan atribut *at* untuk memperoleh ulasan yang sesuai dengan periode penelitian. Data difokuskan pada ulasan yang dipublikasikan dalam rentang 1 Januari 2021 hingga 31 Januari 2026. Hasil penyaringan menunjukkan bahwa sebanyak 17.212 ulasan memenuhi kriteria periode penelitian dan selanjutnya digunakan sebagai dataset awal untuk proses *preprocessing*. Penyaringan berdasarkan periode tersebut dilakukan untuk memastikan seluruh data yang dianalisis berada dalam rentang waktu yang telah ditetapkan.

Tabel 1. Ulasan Pengguna Kredivo pada Google Play Store

Ulasan	Score	At
Alhamdulillah mantap sangat kredivo, moga sukses selalu	5	2026-01-30 18:13:37
Kecewa, limit masih banyak tapi gabisa pinjam aneh	2	2026-01-30 12:19:03
Saya merasa terbantukan dan dipermudah dengan memakai aplikasi Kredivo	4	2026-01-23 03:45:15
Sangat mengecewakan! Baru daftar sudah ditolak, padahal belum pernah pakai kredivo!	1	2026-01-14 06:09:36
Transaksi sering ditolak, padahal tidak pernah telat	3	2026-01-14 00:31:21

Tabel 1 menunjukkan contoh data ulasan yang diperoleh dari Google Play Store beserta rating dan waktu publikasinya. Ulasan yang dikumpulkan memiliki karakteristik yang beragam, mulai dari apresiasi terhadap layanan hingga keluhan mengenai penolakan transaksi dan penggunaan limit. Keberagaman karakteristik tersebut menjadi dasar untuk melakukan pengolahan teks dan klasifikasi sentimen pada tahap berikutnya.

Preprocessing Data

Data hasil *scraping* masih mengandung berbagai bentuk ketidakteraturan, seperti penggunaan huruf kapital, tanda baca, karakter khusus, kata tidak baku, serta data duplikat dan ulasan yang tidak memiliki informasi tekstual. Oleh karena itu, dilakukan *preprocessing* untuk mengurangi *noise* dan menyeragamkan bentuk data sebelum digunakan pada proses ekstraksi fitur. Tahap ini diperlukan agar data teks memiliki format yang lebih terstruktur sehingga kata-kata yang memiliki makna

serupa dapat diproses secara konsisten. Selain itu, pembersihan data membantu mengurangi informasi yang tidak relevan dan berpotensi memengaruhi hasil klasifikasi sentimen.

Tahapan *preprocessing* terdiri atas *cleansing*, *case folding*, *tokenizing*, normalisasi kata, *stopword removal*, dan *stemming*. Setiap tahapan memiliki fungsi yang berbeda dalam mempersiapkan teks, mulai dari menghilangkan karakter yang tidak diperlukan hingga mengubah kata berimbuhan menjadi bentuk dasarnya.

Tabel 2. Hasil Preprocessing Data

Tahapan	Hasil
Content	Aplikasinya bermanfaat banget, terimakasih yah semoga sukses terus, ku kasi bintang 5
Casefolding	Aplikasinya bermanfaat banget terimakasih yah semoga sukses terus ku kasi bintang
Cleansing	Aplikasinya bermanfaat banget terimakasih yah semoga sukses terus ku kasi bintang
Tokenizing	[aplikasinya, bermanfaat, banget, terimakasih, yah, semoga, sukses, terus, ku, kasi, bintang]
Normalisasi	[aplikasinya, bermanfaat, banget, terima kasih, ya, semoga, sukses, terus, ku, kasih, bintang]
Stopword	[aplikasinya, bermanfaat, banget, terima kasih, semoga, sukses, kasih, bintang]
Stemming	[aplikasi, manfaat, banget, terima kasih, semoga, sukses, kasih, bintang]

Tabel 2 menunjukkan bahwa *preprocessing* mengubah teks mentah menjadi bentuk yang lebih seragam dan terstruktur. *Cleansing* digunakan untuk menghilangkan tanda baca dan karakter yang tidak diperlukan, sedangkan *case folding* menyeragamkan huruf menjadi huruf kecil. Tahap *tokenizing* memecah kalimat menjadi unit kata, kemudian normalisasi digunakan untuk mengubah kata tidak baku menjadi bentuk yang lebih sesuai. Selanjutnya, *stopword removal* mengurangi kata yang kurang memberikan informasi terhadap proses klasifikasi, sedangkan *stemming* mengubah kata berimbuhan menjadi bentuk dasarnya.

Setelah seluruh tahapan *preprocessing* dilakukan, termasuk penghapusan baris kosong, penghilangan data duplikat, serta penanganan ulasan yang hanya terdiri atas simbol atau emoji, jumlah data berkurang dari 17.212 menjadi 10.798 ulasan bersih. Dengan demikian, sebanyak 6.414 data tidak digunakan dalam tahap klasifikasi karena tidak memenuhi kriteria data teks yang dapat diproses.

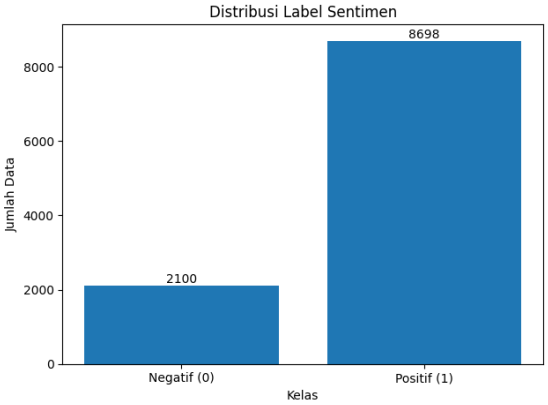
Pelabelan Sentimen

Pelabelan awal dilakukan berdasarkan rating pengguna. Ulasan dengan rating 4 dan 5 dikategorikan sebagai sentimen positif, sedangkan rating 1 dan 2 dikategorikan sebagai sentimen negatif. Ulasan dengan rating 3 tidak langsung dikategorikan karena rating tersebut dapat merepresentasikan pengalaman positif maupun negatif. Oleh sebab itu, ulasan rating 3 diverifikasi berdasarkan konteks teks untuk memastikan kesesuaian antara nilai rating dan makna yang disampaikan pengguna. Pendekatan ini digunakan agar pelabelan tidak hanya bergantung pada nilai rating, tetapi juga mempertimbangkan informasi yang terkandung dalam teks ulasan.

Hasil pelabelan awal menghasilkan 8.659 ulasan positif, 1.903 ulasan negatif, dan 236 ulasan dengan rating 3. Dari 236 ulasan rating 3, sebanyak 197 ulasan dikategorikan sebagai negatif karena didominasi oleh keluhan atau kendala penggunaan, sedangkan 39 ulasan dikategorikan sebagai positif karena mengandung apresiasi terhadap layanan. Dengan demikian, seluruh data dapat dikelompokkan ke dalam dua kelas sentimen berdasarkan hasil pelabelan dan verifikasi.

Berdasarkan hasil tersebut, distribusi akhir label sentimen terdiri atas 8.698 ulasan positif, yang diperoleh dari 8.659 ulasan dengan rating 4-5 dan 39 ulasan

rating 3 yang telah diverifikasi sebagai positif. Sementara itu, 2.100 ulasan termasuk dalam kelas negatif, yang terdiri atas 1.903 ulasan dengan rating 1-2 dan 197 ulasan rating 3 yang teridentifikasi mengandung keluhan. Jumlah keseluruhan data yang digunakan dalam proses klasifikasi adalah 10.798 ulasan. Untuk kebutuhan pemodelan, kelas positif dipetakan menjadi label 1, sedangkan kelas negatif dipetakan menjadi label 0.



Gambar 2. Distribusi Label Sentimen

Gambar 2 menunjukkan bahwa data penelitian memiliki distribusi kelas yang tidak seimbang. Sebanyak 8.698 ulasan (80,57%) termasuk kelas positif, sedangkan 2.100 ulasan (19,43%) termasuk kelas negatif. Perbedaan jumlah tersebut menunjukkan bahwa kelas positif memiliki proporsi lebih besar dibandingkan kelas negatif. Kondisi *class imbalance* tersebut perlu diperhatikan pada proses pelatihan karena model berpotensi lebih dominan mempelajari karakteristik kelas mayoritas. Oleh karena itu, SMOTE diterapkan pada data pelatihan untuk meningkatkan keterwakilan kelas minoritas tanpa mengubah data validasi dan data uji.

Pembobotan Fitur dengan TF-IDF

Setelah proses pelabelan, teks ulasan diubah menjadi representasi numerik menggunakan metode *Term Frequency-Inverse Document Frequency* (TF-IDF). Proses pembobotan dilakukan menggunakan *TfidfVectorizer* dari pustaka *scikit-learn*. Hasil ekstraksi fitur menghasilkan 4.558 kosakata unik, sehingga diperoleh matriks TF-IDF berukuran 10.798 × 4.558. Pembobotan ini memungkinkan setiap kata direpresentasikan berdasarkan tingkat kepentingannya dalam suatu ulasan dibandingkan dengan keseluruhan dokumen. Fitur yang dihasilkan selanjutnya digunakan sebagai masukan bagi algoritma *Multinomial Naive Bayes* dan *Logistic Regression* dalam proses klasifikasi sentimen. Dengan demikian, teks ulasan yang semula berbentuk data tidak terstruktur dapat diolah menjadi representasi numerik yang sesuai untuk pemodelan.

Tabel 3. Hasil Frekuensi Kemunculan Kata TF-IDF

Dokumen	Kata	TF-IDF
0	Aman	0.415538
1	Bagus	0.809331
2	Mantap	1
3	Manfaat	0.389027
3	Sukses	0.377183

Tabel 3 memperlihatkan contoh nilai bobot TF-IDF pada beberapa dokumen dan kata. Nilai TF-IDF menunjukkan tingkat kepentingan suatu kata dalam dokumen dengan mempertimbangkan kemunculannya pada keseluruhan kumpulan dokumen. Kata yang memiliki karakteristik lebih spesifik terhadap dokumen memperoleh bobot yang relatif lebih tinggi. Hasil pembobotan tersebut selanjutnya digunakan sebagai fitur masukan bagi algoritma *Multinomial Naive Bayes* dan *Logistic Regression*.

Pembagian Data

Dataset kemudian dibagi menjadi data latih dan data uji dengan proporsi 80:20 menggunakan *stratified split*. Pendekatan tersebut digunakan untuk mempertahankan proporsi kelas sentimen pada kedua subset.

Tabel 4. Hasil Splitting Data

Kelompok Data	Proporsi %	Jumlah
Data latih (Training)	80%	8.638
Data uji (Testing)	20%	2.160
Total data	100%	10.798

Tabel 4 menunjukkan bahwa sebanyak 8.638 ulasan digunakan sebagai data latih dan 2.160 ulasan digunakan sebagai data uji. Berdasarkan *stratified split*, data latih terdiri atas 6.958 ulasan positif dan 1.680 ulasan negatif, sedangkan data uji terdiri atas 1.740 ulasan positif dan 420 ulasan negatif. Data uji tidak digunakan dalam proses pelatihan maupun SMOTE sehingga dapat digunakan sebagai data yang belum pernah dilihat model untuk mengukur kemampuan generalisasi.

5-Fold Cross Validation

Evaluasi pada data latih dilakukan menggunakan metode 5-Fold *Cross Validation*. Sebanyak 8.638 data latih dibagi menjadi lima *fold*. Pada setiap iterasi, empat *fold* digunakan sebagai data pelatihan dan satu *fold* digunakan sebagai data validasi. Proses tersebut diulang sebanyak lima kali sehingga setiap *fold* memperoleh kesempatan menjadi data validasi.

Tabel 5. Pembagian Data pada 5-Fold Cross Validation

Iterasi	Data Pelatihan	Data Validasi	Total Data
Fold 1	6.910	1.728	8.638
Fold 2	6.910	1.728	8.638
Fold 3	6.910	1.728	8.638
Fold 4	6.911	1.727	8.638
Fold 5	6.911	1.727	8.638

Tabel 5 menunjukkan bahwa pembagian data pada lima *fold* menghasilkan jumlah data yang hampir sama pada setiap iterasi. Perbedaan satu data pada beberapa *fold* terjadi karena jumlah 8.638 tidak dapat dibagi secara tepat menjadi lima bagian yang sama besar. Pada setiap iterasi, data validasi tidak digunakan untuk membentuk data sintetis maupun melatih model pada *fold* tersebut sehingga proses evaluasi dapat dilakukan secara lebih objektif.

Penerapan SMOTE

Ketidakseimbangan kelas pada data latih ditangani menggunakan SMOTE (*Synthetic Minority Oversampling Technique*). Penerapan SMOTE dilakukan hanya pada data pelatihan di setiap *fold*, sedangkan data validasi dan data uji tetap dalam kondisi asli. Strategi tersebut digunakan untuk mencegah *data leakage* karena informasi dari data validasi atau data uji tidak digunakan dalam pembentukan sampel sintetis.

Tabel 6. Distribusi Jumlah Kelas Sebelum dan Sesudah SMOTE

Iterasi	Label	Sebelum SMOTE	Sesudah SMOTE	Sampel Sintesis	Total
Fold 1	Positif (1)	5.551	5.551		11.102
	Negatif (0)	1.359	5.551	4.192	
Fold 2	Positif (1)	5.577	5.577		11.154
	Negatif (0)	1.333	5.577	4.244	
Fold 3	Positif (1)	5.543	5.543		11.086
	Negatif (0)	1.367	5.543	4.176	
Fold 4	Positif (1)	5.587	5.587		11.174
	Negatif (0)	1.324	5.587	4.263	
Fold 5	Positif (1)	5.574	5.574		11.148
	Negatif (0)	1.337	5.574	4.237	

Tabel 6 menunjukkan bahwa SMOTE berhasil menyetarakan jumlah sampel kelas negatif dengan kelas positif pada data pelatihan setiap *fold*. Sebelum SMOTE, jumlah kelas negatif lebih rendah dibandingkan kelas positif. Setelah SMOTE, kedua kelas memiliki jumlah sampel yang sama, yaitu 5.551 pada Fold 1, 5.577 pada Fold 2, 5.543 pada Fold 3, 5.587 pada Fold 4, dan 5.574 pada Fold 5. Dengan demikian, proses pelatihan dilakukan pada distribusi kelas yang seimbang, sedangkan

data validasi tetap mempertahankan distribusi aslinya.

Evaluasi Multinomial Naive Bayes

Model *Multinomial Naive Bayes* dilatih menggunakan fitur TF-IDF dengan SMOTE yang diterapkan pada data pelatihan setiap *fold*. Evaluasi dilakukan menggunakan lima *fold* untuk mengetahui kestabilan performa model. Penerapan SMOTE pada masing-masing *fold* dilakukan untuk mengatasi ketidakseimbangan jumlah kelas tanpa memengaruhi data validasi. Selanjutnya, hasil evaluasi dari setiap *fold* dibandingkan untuk melihat konsistensi kemampuan model dalam mengklasifikasikan sentimen positif dan negatif. Pendekatan ini memberikan gambaran yang lebih menyeluruh mengenai performa model dibandingkan evaluasi yang hanya menggunakan satu pembagian data.

Tabel 7. Evaluasi Performa Naive Bayes per Fold

Iterasi	Accuracy	Precision	Recall	F1-Score
Fold 1	0,8912	0,9160	0,8912	0,8977
Fold 2	0,8883	0,9098	0,8883	0,8940
Fold 3	0,8785	0,9178	0,8785	0,8878
Fold 4	0,8674	0,9010	0,8674	0,8756
Fold 5	0,8906	0,9139	0,8906	0,8965
Rata-rata	0,8832	0,9117	0,8832	0,8904

Tabel 7 menunjukkan bahwa *Naive Bayes* memperoleh rata-rata akurasi sebesar 88,32% dan F1-Score sebesar 89,04%. Akurasi tertinggi diperoleh pada Fold 1 sebesar 89,12%, sedangkan akurasi terendah terdapat pada Fold 4 sebesar 86,74%. Rentang performa tersebut menunjukkan adanya variasi performa antarfold, tetapi model tetap mempertahankan akurasi di atas 86% pada seluruh iterasi. Perbedaan nilai akurasi antarfold menunjukkan bahwa karakteristik data pada masing-masing subset dapat memengaruhi hasil klasifikasi, meskipun perbedaannya masih berada pada rentang yang relatif terbatas.

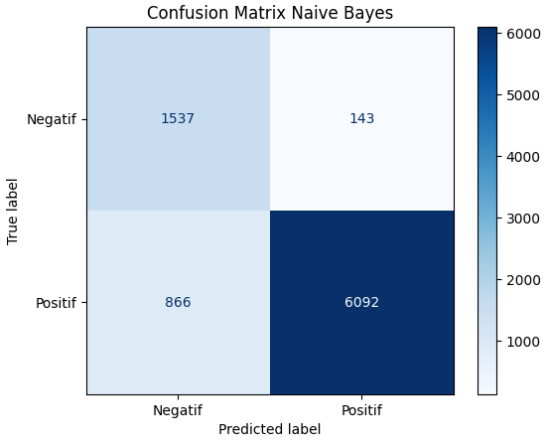
Nilai F1-Score yang mencapai 89,04% menunjukkan bahwa model mampu mempertahankan keseimbangan antara *precision* dan *recall* dalam proses klasifikasi. Konsistensi performa pada kelima *fold* juga menunjukkan bahwa hasil model tidak hanya bergantung pada satu pembagian data tertentu. Hasil tersebut menjadi dasar untuk membandingkan performa *Naive Bayes* dengan *Logistic Regression* pada tahap evaluasi berikutnya.

Tabel 8. Performa Naive Bayes berdasarkan Kelas Sentimen

Kelas Sentimen	Precision	Recall	F1-Score	Karakteristik Klasifikasi
Kelas negatif (0)	0,6404	0,9155	0,7531	Memiliki nilai *recall* yang sangat tinggi, sehingga model mampu mendeteksi sebagian besar ulasan negatif dengan baik, meskipun nilai *precision* masih relatif rendah.
Kelas positif (1)	0,9771	0,8755	0,9234	Memiliki nilai *precision* yang sangt tingggi, menunjukkan bahwa sebagian besar ulasan yang diprediksi sebagai positif benar-benar termasuk kedalam kelas positif.

Tabel 8 menunjukkan bahwa *Naive* Bayes memiliki karakteristik performa yang berbeda pada kedua kelas. Pada kelas negatif, recall mencapai 91,55%, sedangkan precision sebesar 64,04%. Kondisi tersebut menunjukkan bahwa model mampu menemukan sebagian besar ulasan negatif, tetapi sebagian prediksi negatif masih berasal dari ulasan yang sebenarnya positif. Sebaliknya, kelas

positif memiliki precision sebesar 97,71% dan recall sebesar 87,55%, sehingga prediksi positif relatif lebih tepat meskipun masih terdapat sebagian ulasan positif yang dikategorikan sebagai negatif.



Gambar 3. Confusion Matrix Naive Bayes

Gambar 3 memperlihatkan bahwa model menghasilkan 1.537 True Negative (TN), yaitu ulasan negatif yang berhasil diklasifikasikan sebagai negatif, serta 143 False Positive (FP), yaitu ulasan negatif yang diprediksi sebagai positif. Pada kelas positif terdapat 866 False Negative (FN), yaitu ulasan positif yang diprediksi sebagai negatif, dan 6.092 True Positive (TP), yaitu ulasan positif yang berhasil diprediksi dengan benar.

Hasil tersebut menunjukkan bahwa kesalahan klasifikasi terbesar terjadi pada 866 ulasan positif yang diprediksi sebagai negatif. Sementara itu, tingginya nilai *recall* kelas negatif menunjukkan bahwa model relatif sensitif dalam mendeteksi ulasan yang mengandung keluhan. Karakteristik tersebut berkaitan dengan kemampuan *Multinomial Naive Bayes* dalam menghitung probabilitas kelas berdasarkan distribusi fitur teks, meskipun model memiliki asumsi bahwa fitur bersifat independen satu sama lain.

Evaluasi Logistic Regression

Model *Logistic Regression* dengan *max_iter* = 1000 kemudian dilatih menggunakan skema 5-Fold *Cross Validation* dan SMOTE. Evaluasi dilakukan untuk membandingkan kemampuan model dengan *Naive Bayes* dalam mengklasifikasikan sentimen ulasan pengguna. Penggunaan *max_iter* = 1000 bertujuan memberikan jumlah iterasi yang memadai agar proses optimasi model dapat mencapai kondisi konvergen pada data dengan jumlah fitur yang cukup besar. Sementara itu, penerapan SMOTE pada setiap data pelatihan *fold* dilakukan untuk mengurangi pengaruh ketidakseimbangan kelas tanpa mengubah distribusi data validasi.

Melalui 5-Fold *Cross Validation*, performa *Logistic Regression* dapat diamati pada beberapa pembagian data sehingga kestabilan model dapat dievaluasi secara lebih menyeluruh. Nilai *accuracy*, *precision*, *recall*, dan F1-Score dari setiap *fold* kemudian digunakan untuk memperoleh gambaran rata-rata performa model. Hasil tersebut selanjutnya dibandingkan dengan *Naive Bayes* untuk mengetahui algoritma yang memberikan kinerja klasifikasi lebih baik pada dataset ulasan pengguna Kredivo.

Tabel 9. Evaluasi Performa Logistic Regression per Fold

Iterasi	Accuracy	Precision	Recall	F1-Score
Fold 1	0,9051	0,9173	0,9051	0,9089
Fold 2	0,8947	0,9045	0,8947	0,8980
Fold 3	0,8999	0,9222	0,8999	0,9058
Fold 4	0,9010	0,9129	0,9010	0,9045
Fold 5	0,9062	0,9144	0,9062	0,9089
Rata-rata	0,9014	0,9143	0,9014	0,9052

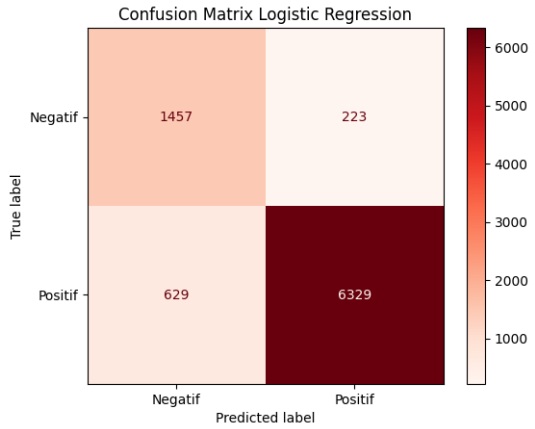
Tabel 9 menunjukkan bahwa *Logistic Regression* memperoleh rata-rata akurasi sebesar 90,14% dan F1-Score sebesar 90,52%. Akurasi tertinggi diperoleh pada

Fold 5 sebesar 90,62%, sedangkan nilai terendah terdapat pada Fold 2 sebesar 89,47%. Dibandingkan dengan *Naive Bayes*, *Logistic Regression* menghasilkan rata-rata akurasi dan F1-Score yang lebih tinggi. Sama seperti Tabel 7, nilai precision dan recall pada tabel ini perlu diverifikasi kembali dengan keluaran kode untuk memastikan metrik dihitung menggunakan parameter evaluasi yang sesuai.

Tabel 10. Performa Logistic Regression berdasarkan Kelas Sentimen

Kelas Sentimen	Precision	Recall	F1-Score	Karakteristik Klasifikasi
Kelas negatif (0)	0,6988	0,8681	0,7737	Mengalami peningkatan nilai *precision* yang signifikan dibandingkan model Naive Bayes, sehingga kemampuan model ini lebih selektif.
Kelas positif (1)	0,9660	0,9096	0,9369	Memiliki nilai *precision* dan *recall* yang sama-sama tinggi, menunjukkan kemampuan klasifikasi yang seimbang dan akurat dalam mengenali ulasan.

Tabel 10 menunjukkan bahwa *Logistic Regression* menghasilkan *precision* kelas negatif sebesar 69,88% dan *recall* sebesar 86,81%. Dibandingkan dengan *Naive Bayes*, nilai *precision* kelas negatif meningkat sehingga jumlah prediksi negatif yang keliru dapat ditekan. Pada kelas positif, model memperoleh *precision* 96,60% dan *recall* 90,96%, yang menunjukkan kemampuan lebih baik dalam mengenali ulasan positif dibandingkan *Naive Bayes*.



Gambar 2. Confusion Matrix Logistic Regression

Gambar 4 menunjukkan bahwa *Logistic Regression* menghasilkan 1.457 True Negative (TN) dan 223 False Positive (FP) pada kelas negatif. Sementara itu, pada kelas positif terdapat 629 False Negative (FN) dan 6.329 True Positive (TP). Dibandingkan *Naive Bayes*, jumlah *False Negative* berkurang dari 866 menjadi 629 ulasan.

Berdasarkan hasil tersebut, *Logistic Regression* menunjukkan performa yang lebih baik dibandingkan *Naive Bayes*. Model memperoleh rata-rata akurasi sebesar 90,14% dan F1-Score sebesar 90,52%, lebih tinggi dibandingkan masing-masing 88,32% dan 89,04% pada *Naive Bayes*. Penurunan jumlah *False Negative* menunjukkan bahwa *Logistic Regression* lebih mampu mengenali ulasan positif yang sebelumnya berpotensi diklasifikasikan sebagai negatif. Perbedaan tersebut dapat berkaitan dengan kemampuan *Logistic Regression* dalam mempelajari hubungan berbobot antara fitur TF-IDF dan kelas sentimen tanpa menggunakan asumsi independensi fitur seperti pada *Naive Bayes*.

Pengujian pada Data Testing

Setelah proses *Cross Validation*, pengujian akhir dilakukan menggunakan 20% data yang telah dipisahkan sejak awal. Data tersebut terdiri atas 2.160 ulasan, dengan 1.740 ulasan positif dan 420 ulasan negatif. Data uji tidak digunakan dalam proses pelatihan, SMOTE, maupun *Cross Validation*, sehingga digunakan untuk mengukur kemampuan generalisasi model terhadap data yang belum pernah digunakan sebelumnya.

Tabel 11. Hasil Pengujian Model dengan Data Testing 20%

Model	Accuracy	Precision	Recall	F1-Score
Naive Bayes	0,8884	0,9139	0,8884	0,8949
Logistic Regression	0,9042	0,9141	0,9042	0,9074

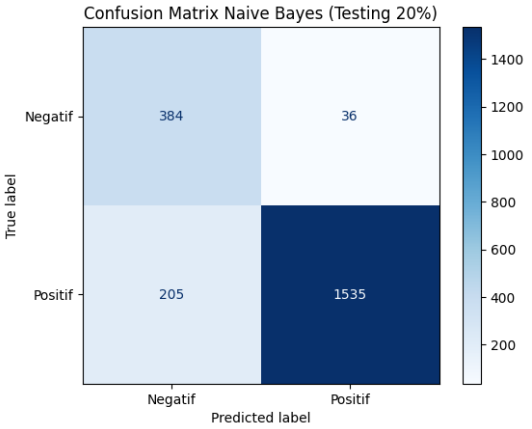
Tabel 11 menunjukkan bahwa *Logistic Regression* kembali menghasilkan performa yang lebih tinggi dibandingkan *Naive Bayes* pada data uji. *Logistic Regression* memperoleh akurasi sebesar 90,42% dan F1-Score sebesar 90,74%, sedangkan *Naive Bayes* memperoleh akurasi 88,84% dan F1-Score 89,49%. Perbedaan tersebut menunjukkan bahwa keunggulan *Logistic Regression* tidak hanya terlihat pada proses *Cross Validation*, tetapi juga tetap terlihat ketika model diuji menggunakan data yang tidak digunakan selama pelatihan.

Tabel 12. Performa Kelas Sentimen pada Data Testing

Model	Kelas Sentimen	Precision	Recall	F1-Score	Support
Naive Bayes	Negatif (0)	0,65	0,91	0,76	420
	Positif (1)	0,98	0,88	0,93	1.740
	Weighted Average	0,91	0,89	0,89	2.160
Logistic Regression	Negatif (0)	0,71	0,85	0,78	420
	Positif (1)	0,96	0,92	0,94	1.740
	Weighted Average	0,91	0,90	0,91	2.160

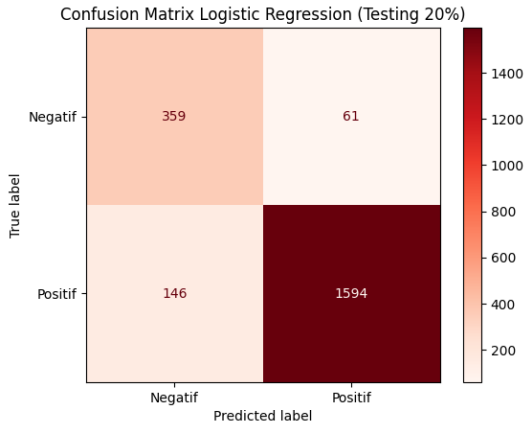
Tabel 12 menunjukkan bahwa *Naive Bayes* memiliki *recall* kelas negatif sebesar 91%, sedangkan *Logistic Regression* memperoleh 85%. Dengan demikian, *Naive Bayes* lebih sensitif dalam mendeteksi ulasan negatif, tetapi memiliki *precision* yang lebih rendah, yaitu 65%. Sebaliknya, *Logistic Regression* memperoleh *precision* kelas negatif sebesar 71%, yang menunjukkan bahwa prediksi negatif yang dihasilkan lebih selektif. Pada kelas positif, *Logistic Regression* memperoleh *recall* sebesar 92%, lebih tinggi dibandingkan *Naive Bayes* sebesar 88%.

Hasil pengujian memperlihatkan bahwa *Logistic Regression* memperoleh F1-Score *weighted average* sebesar 90,74%, sedangkan *Naive Bayes* sebesar 89,49%. Perbedaan tersebut memperlihatkan bahwa *Logistic Regression* memberikan keseimbangan performa yang lebih baik pada kedua kelas, terutama karena mampu mempertahankan *recall* kelas positif yang lebih tinggi sekaligus meningkatkan *precision* kelas negatif.



Gambar 5. Confusion Matrix Naive Bayes Data Testing 20%

Gambar 5 menunjukkan bahwa *Naive Bayes* menghasilkan 384 True Negative (TN) dan 36 False Positive (FP) pada kelas negatif. Pada kelas positif terdapat 205 False Negative (FN) dan 1.535 True Positive (TP). Dengan demikian, sebagian besar data uji berhasil diklasifikasikan dengan benar, tetapi masih terdapat 205 ulasan positif yang diprediksi sebagai negatif.



Gambar 6. Confusion Matrix Logistic Regression Data Testing 20%

Gambar 6 menunjukkan bahwa *Logistic Regression* menghasilkan 359 True Negative (TN), 61 False Positive (FP), 146 False Negative (FN), dan 1.594 True Positive (TP). Jumlah *False Negative* turun dari 205 pada *Naive Bayes* menjadi 146 pada *Logistic Regression*. Penurunan tersebut menunjukkan bahwa *Logistic Regression* lebih mampu mengenali ulasan positif pada data uji. Meskipun jumlah *False Positive* meningkat dari 36 menjadi 61, peningkatan jumlah *True Positive* dari 1.535 menjadi 1.594 menghasilkan akurasi keseluruhan yang lebih tinggi.

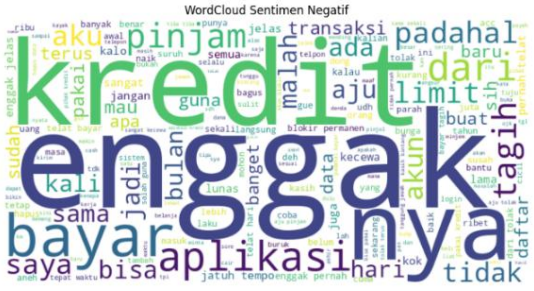
Berdasarkan keseluruhan pengujian, *Logistic Regression* memberikan performa yang lebih baik dibandingkan *Naive Bayes*. Performa tersebut terlihat dari akurasi 90,42% dan F1-Score 90,74% pada data uji. Namun, hasil ini tidak berarti bahwa *Naive Bayes* lebih buruk pada seluruh aspek, karena *Naive Bayes* justru memiliki *recall* kelas negatif yang lebih tinggi. Dengan demikian, pemilihan model dapat disesuaikan dengan tujuan klasifikasi, yaitu *Logistic Regression* lebih unggul secara keseluruhan, sedangkan *Naive Bayes* lebih sensitif dalam mendeteksi kelas negatif.

Analisis Word Cloud

Untuk mengidentifikasi kata-kata yang paling dominan pada masing-masing kelompok sentimen, digunakan visualisasi *Word Cloud*. Ukuran kata pada visualisasi menunjukkan tingkat kemunculan kata dalam kumpulan ulasan yang dianalisis. Analisis ini digunakan sebagai pelengkap hasil klasifikasi untuk memberikan gambaran mengenai topik atau keluhan yang sering muncul dalam ulasan pengguna.

Gambar 7. Word Cloud Sentimen Positif

Gambar 7 memperlihatkan sejumlah kata yang dominan pada ulasan positif, seperti cepat, cair, mudah, bantu, mantap, percaya, baik, dan sukses. Kemunculan kata-kata tersebut menunjukkan bahwa ulasan positif banyak berkaitan dengan kemudahan penggunaan aplikasi, proses layanan, serta pengalaman pengguna yang dianggap membantu. Kata seperti *cepat*, *cair*, dan *mudah* dapat mengindikasikan adanya apresiasi terhadap proses layanan, sedangkan kata *mantap*, *baik*, dan *sukses* menunjukkan bentuk apresiasi atau kepuasan pengguna. Interpretasi tersebut perlu dipahami sebagai kecenderungan topik berdasarkan frekuensi kata, bukan sebagai ukuran langsung tingkat kepuasan pengguna.



Gambar 8. *Word Cloud* Sentimen Negatif

Gambar 8 menunjukkan bahwa kata-kata yang dominan pada ulasan negatif antara lain limit, tolak, blokir, tagih, denda, jatuh tempo, dan telat. Kemunculan kata tersebut menunjukkan bahwa keluhan pengguna banyak berkaitan dengan penggunaan limit, penolakan transaksi,

pembatasan akun, serta persoalan pembayaran. Selain itu, kata seperti *aplikasi*, *transaksi*, dan *error* mengindikasikan adanya keluhan yang berkaitan dengan kendala penggunaan aplikasi atau proses transaksi. Temuan *Word Cloud* tersebut melengkapi hasil klasifikasi dengan menunjukkan tema yang paling sering muncul pada ulasan negatif pengguna.

4. Kesimpulan

Berdasarkan hasil analisis sentimen terhadap ulasan pengguna aplikasi Kredivo pada Google Play Store periode Januari 2021 hingga Januari 2026, diperoleh 10.798 ulasan bersih yang digunakan dalam proses klasifikasi. Distribusi sentimen menunjukkan bahwa 8.698 ulasan (80,57%) termasuk sentimen positif dan 2.100 ulasan (19,43%) termasuk sentimen negatif. Hasil tersebut menunjukkan bahwa ulasan pengguna didominasi oleh sentimen positif, meskipun masih terdapat sejumlah keluhan yang berkaitan dengan penggunaan layanan.

Perbandingan model menunjukkan bahwa Logistic Regression memberikan performa lebih baik dibandingkan *Multinomial Naive Bayes*. Pada 5-Fold *Cross Validation* dengan SMOTE, *Logistic Regression* memperoleh rata-rata akurasi 90,14% dan F1-Score 90,52%, sedangkan *Naive Bayes* memperoleh akurasi 88,32% dan F1-Score 89,04%. Pada pengujian data *testing* sebanyak 2.160 ulasan, *Logistic Regression* kembali menunjukkan performa lebih tinggi dengan akurasi 90,42% dan F1-Score 90,74%.

Penerapan *preprocessing*, pembobotan TF-IDF dengan 4.558 fitur, serta SMOTE pada data pelatihan membantu mempersiapkan data untuk proses klasifikasi dan menangani ketidakseimbangan kelas. Analisis *Word Cloud* menunjukkan bahwa sentimen positif banyak berkaitan dengan kemudahan, kecepatan, dan manfaat layanan, sedangkan sentimen negatif terutama berkaitan dengan penolakan transaksi, limit, pembekuan akun, penagihan, denda, dan kendala aplikasi.

Penelitian selanjutnya dapat mengembangkan penggunaan fitur *N-Gram* untuk menangkap konteks antarkata serta membandingkan *Logistic Regression* dengan metode *Deep Learning* atau model Transformer bahasa Indonesia untuk memperoleh hasil klasifikasi yang lebih mendalam.

Daftar Pustaka

[1] OJK, "Direktori Layanan Pendanaan Bersama Berbasis Teknologi Informasi," Otoritas Jasa Keuangan. [Daring]. Tersedia pada: https://ojk.go.id/id/kanal/iknb/data-dan-statistik/direktori/fintech/Pages/Direktori-LPBBTI-28-Februari-2026.aspx

[2] APJII, "Survei Internet APJII 2026 - Jumlah Pengguna Internet di Indonesia," Asosiasi Penyelenggara Jasa Internet Indonesia. [Daring]. Tersedia pada: https://survei.apjii.or.id

[3] Kredivo, "Kredivo - Paylater & Pinjaman," PT Kredivo Finance Indonesia. [Daring]. Tersedia pada: https://play.google.com/store/apps/details?id=com.finaccel.android&pcampaignid=web_share

[4] A. Suwendi, N. Indriani, N. S. Sahira, and M. L. Wardiyah, "Pengaruh Ulasan dan Rating Produk terhadap Minat Beli Berdasarkan Kelompok Usia pada Platform E-commerce," *Moneter: Jurnal Ekonomi dan Keuangan*, vol. 3, no. 3, pp. 144–153, 2025, https://doi.org/10.61132/moneter.v3i3.1498

[5] L. O. Lukmana, D. R. L., and P. Elisya, "Analisis sentimen terhadap ulasan pengguna aplikasi Threads Instagram di PlayStore menggunakan algoritma Naive Bayes," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 13, no. 2, pp. 496–504,

2025, https://doi.org/10.23960/jitet.v13i2.6250

[6] N. Nurwanda, N. Suarna, and W. Prihartono, "Penerapan NLP (Natural Language Processing) dalam analisis sentimen pengguna Telegram di PlayStore," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 8, no. 2, pp. 1841–1846, 2024, https://doi.org/10.36040/jati.v8i2.8469.

[7] M. Afdal and L. R. Elita, "Penerapan text mining pada aplikasi Tokopedia menggunakan algoritma K-nearest neighbor," *Jurnal Ilmiah Rekayasa dan Manajemen Sistem Informasi*, vol. 8, no. 1, pp. 78–87, 2022, https://doi.org/10.24014/rmsi.v8i1.16595.

[8] A. Aryanusa, R. M. Akbar, dan Y. N. S, "Evaluasi Kinerja Logistic Regression Dan Multinomial Naïve Bayes Dalam Klasifikasi Sentimen Mobile Banking," *J. Inform. dan Tek. Elektro Terap.*, vol. 13, no. 3S1, hal. 463–474, 2025, [Daring]. Tersedia pada: https://journal.eng.unila.ac.id/index.php/jitet/article/view/7687

[9] I. Dikusuma and Y. Widjaja, "Perbandingan Naïve Bayes dan Logistic Regression untuk analisis sentimen ulasan produk skincare di Tokopedia," *Jurnal Informatika dan Teknologi Komputer*, vol. 5, no. 3, pp. 242–251, 2025, https://doi.org/10.55606/jitek.v5i3.8420.

[10] A. Bachtiar, M. J. Hawali, N. C. Simbolon, D. A. Maulana, and R. A. Anggraini, "Analisis sentimen ulasan Free Fire menggunakan Naive Bayes Logistic Regression," *Journal of Information System and Management*, vol. 7, no. 2, 2026, https://doi.org/10.24076/joism.2026v7i2.2433.

[11] R. R. Salam, M. F. Jamil, Y. Ibrahim, Rahmaddeni, Soni, and Herianto, "Analisis sentimen terhadap Bantuan Langsung Tunai (BLT) Bahan Bakar Minyak (BBM) menggunakan Support Vector Machine," *Indonesian Journal of Machine Learning and Computer Science*, vol. 3, no. 1, pp. 27–35, 2023.

[12] D. M. Dyasputro, "Rule-based text normalization library for Bahasa Indonesia," in *Proc. Int. Conf. Inf. Technol. Digit. Appl.*, pp. 1–8, 2025, https://doi.org/10.1109/ICITDA68167.2025.11332325.

[13] J. Pardede and D. Darmawan, "Perbandingan algoritma stemming Porter, Sastrawi, Idris, dan Arifin Setiono pada dokumen teks Indonesia," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 12, no. 1, pp. 69–76, 2025, https://doi.org/10.25126/jtiik.2025128860.

[14] P. S. Silitonga, A. R. Tatipang, E. Salsabilla, and A. P. Sari, "Analisis sentimen ulasan pengguna aplikasi Tiket.com dengan K-Nearest Neighbor," *JITET (Jurnal Informatika dan Teknik Elektro Terapan)*, vol. 13, no. 3, pp. 818–827, 2025, https://doi.org/10.23960/jitet.v13i3.7012.

[15] HSari, G. L. Ginting, T. Zebua, and Mesran, "Penerapan algoritma text mining dan TF-IDF untuk pengelompokan topik skripsi pada aplikasi repository STMIK Budi Darma," *TIN: Terapan Informatika Nusantara*, vol. 2, no. 7, pp. 414–432, 2021. [Online]. Available: https://ejurnal.seminar-id.com/index.php/tin

[16] A. Pranata, Rudirman, dan N. Azmi Verdikha, "Jurnal Informatika Terpadu Twitter Menggunakan Metode TF-IDF dan Naive Bayes," *J. Inform. terpadu*, vol. 10, no. 2, hal. 93–100, 2024.

[17] Wijiyanto, A. I. Pradana, Sopingi, and V. Atina, "Teknik K-Fold Cross

Validation untuk mengevaluasi kinerja mahasiswa," *J. Algoritm.*, vol. 21, no. 1, pp. 239–248, 2024, https://doi.org/10.33364/algoritma/v.21-1.1618

[18] A. F. Anjani, D. Anggraeni, and I. M. Tirta, "Implementasi Random Forest menggunakan SMOTE untuk analisis sentimen ulasan aplikasi Sister for Students UNEJ," *J. Nas. Teknol. dan Sist. Inf.*, vol. 9, no. 2, pp. 163–172, 2023, https://doi.org/10.25077/teknosi.v9i2.2023.163-172

[19] A. Pebdika, R. Herdiana, and D. Solihudin, "Klasifikasi menggunakan metode Naive Bayes untuk menentukan calon penerima PIP," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 7, no. 1, pp. 452–458, 2023, https://doi.org/10.36040/jati.v7i1.6303.

[20] A. A. Putri, S. Agustian, R. Abdillah, dan Pizaini, "Penerapan Metode Logistic Regression Untuk," *J. Sist. Inf.*, vol. 7, no. 1, hal. 95–107, 2025.

[21] I. Irmayani and B. Asrun, "Klasifikasi sosial ekonomi menggunakan Naïve Bayes Classifier," *Dewantara Journal of Technology*, vol. 2, no. 1, pp. 71–75, Nov. 2021, https://doi.org/10.59563/djtech.v2i1.138.

[22] D. A. Nggego, "Optimalisasi execution time Firefly Algorithm," *Dewantara Journal of Technology*, vol. 1, no. 2, pp. 112–117, May 2021, doi: https://doi.org/10.59563/djtech.v1i2.93