

Analisis Sentimen Ulasan Pengguna Aplikasi Alfagift Menggunakan Random Forest

M. Bagus Prayogi ^{(1)*}, Gustina Masitoh ⁽²⁾

¹ Departemen Informatika, Universitas Nurul Huda, OKU Timur, Indonesia

² Departemen Pendidikan Ekonomi, Universitas Nurul Huda, OKU Timur, Indonesia

e-mail : mhdjesen212@gmail.com, gustina@unuha.ac.id.

* Penulis korespondensi.

Artikel ini diajukan 17 Maret 2024, direvisi 16 Desember 2024, diterima 16 Desember 2024, dan dipublikasikan 31 Mei 2025.

Abstract

Alfagift is a mobile application developed by Alfamart to support online ordering, featuring promotions, transactions, ordering, and delivery from the nearest point based on the consumer's address. User feedback on the Google Play Store reveals mixed sentiments, including both positive and negative responses, which developers can use as material to improve the application's quality. This study focuses on assessing the sentiment of Alfagift app user reviews using the Random Forest algorithm. A total of 4,379 review data points were collected from the Google Play Store and grouped into two categories: positive and negative sentiment. The research steps include data collection, data labeling, data preprocessing, word weighting, dividing the data into training and testing sets, implementing the Random Forest algorithm, and model evaluation. The test results show that the Random Forest algorithm achieves an accuracy of 97.6% and an AUC of 0.98, which falls into the category of excellent classification. This research is expected to contribute to application developers' understanding of user perceptions, enabling them to improve application quality and increase overall user convenience.

Keywords: Sentiment Analysis, Alfagift, Random Forest, Text Mining, Review

Abstrak

Alfagift adalah aplikasi *mobile* yang dikembangkan oleh Alfamart untuk mendukung pemesanan secara *online*, dengan fitur-fitur seperti promo, transaksi, pemesanan, dan pengiriman dari titik terdekat sesuai alamat konsumen. Masukan dari pengguna di Google Play Store menunjukkan sentimen yang beragam, meliputi tanggapan positif dan negatif, yang dapat dimanfaatkan oleh pengembang sebagai bahan untuk meningkatkan kualitas aplikasi. Studi ini berfokus pada pengkajian sentimen dari ulasan pengguna aplikasi Alfagift melalui penerapan algoritma Random Forest. Sebanyak 4.379 data ulasan dikumpulkan dari Google Play Store dan dikelompokkan menjadi dua kategori, yaitu sentimen positif dan negatif. Langkah-langkah penelitian mencakup pengumpulan data, pelabelan data, *preprocessing* data, pembobotan kata, pembagian data menjadi set pelatihan dan pengujian, implementasi algoritma Random Forest, serta evaluasi model. Hasil pengujian menunjukkan bahwa algoritma Random Forest mencapai akurasi sebesar 97,6% dan AUC 0,98, yang masuk dalam kategori klasifikasi sangat baik. Penelitian ini diharapkan dapat memberikan kontribusi bagi pengembang aplikasi dalam memahami persepsi pengguna, sehingga dapat meningkatkan kualitas aplikasi serta meningkatkan kenyamanan pengguna secara keseluruhan.

Kata Kunci: Analisis Sentimen, Alfagift, Random Forest, Text Mining, Ulasan

1. PENDAHULUAN

Trend berbelanja secara *online* meningkat secara signifikan pada masa pandemi COVID-19 (Hanifa et al., 2023). Berbelanja secara *online* kini telah menjadi aktivitas yang tidak terpisahkan dari kehidupan masyarakat. Hal ini disebabkan oleh kemudahan yang tersedia untuk memenuhi kebutuhan primer maupun tersier (Helmi et al., 2023). Belanja *online* dapat dilakukan di mana saja hanya dengan menggunakan *smartphone*. Hal ini menjadi alasan utama mengapa banyak individu cenderung memilih belanja secara *online*, karena pelanggan tidak perlu datang ke toko fisik, prosesnya menjadi lebih cepat dan efisien. Semua transaksi dapat dilakukan dengan mudah



Artikel ini didistribusikan mengikuti lisensi Atribusi-NonKomersial CC BY-NC sebagaimana tercantum pada <https://creativecommons.org/licenses/by-nc/4.0/>.

melalui *smartphone* (Perdana et al., 2022). Seiring dengan meningkatnya tren berbelanja secara *online*, banyak perusahaan bersaing untuk menawarkan layanan sistem penjualan *online* yang memudahkan pelanggan dalam membeli barang. Salah satu jenis layanan tersebut adalah aplikasi yang menghubungkan bisnis dengan pelanggan (Utami et al., 2022). Saat ini, Alfagift merupakan salah satu platform belanja *online* yang sedang berkembang, yang diperkenalkan oleh PT Sumber Alfaria Trijaya Tbk (Firdaus, 2024). Alfagift merupakan layanan digital yang disediakan oleh Alfamart. Aplikasi ini memungkinkan pengguna membeli kebutuhan sehari-hari secara online dan menawarkan berbagai promo menarik bagi pengguna (Fadli et al., 2023). Aplikasi ini menyediakan beragam opsi metode pembayaran, salah satunya adalah melalui transfer bank BCA, e-money, GoPay, dan lainnya, untuk mempermudah pengguna (Perdana et al., 2022).

Di zaman sekarang, terdapat berbagai metode untuk menganalisis sebuah aplikasi, salah satunya melalui analisis sentimen. Analisis sentimen merupakan metode berbasis komputasi yang bertujuan untuk mengidentifikasi opini, perasaan, serta emosi (baik positif maupun negatif) dari individu (Komarudin & Hilda, 2024). Pendekatan ini dilakukan dengan menganalisis elemen dan entitas yang diungkapkan melalui teks. Analisis sentimen menghasilkan opini yang diekstraksi dan diolah secara otomatis dari data teks (Jin et al., 2023). Secara umum, teknik ini memberikan beragam manfaat bagi organisasi, antara lain memahami persepsi masyarakat, meningkatkan pengalaman pelanggan, mendorong efektivitas penjualan dan pemasaran, memantau aktivitas media sosial, mendukung penelitian dan pengembangan, serta memperkuat aspek keamanan dan intelijen (Indrayanto et al., 2023). Analisis sentimen membagi opini atau pendapat menjadi tiga jenis: positif, negatif, dan netral. Namun, dalam proses klasifikasi, sentimen positif dan negatif biasanya lebih diperhatikan, karena sentimen netral sering dianggap kurang signifikan atau tidak memiliki arti yang relevan (Wicaksono, 2022). Model sentimen juga dapat memengaruhi faktor loyalitas, pengiriman, pemesanan, serta memberikan rekomendasi perbaikan pada aplikasi (Firdaus, 2024).

Penelitian terdahulu oleh Anita Firdaus berjudul “*Analisis Sentimen pada Aplikasi Alfagift Menggunakan Metode Naïve Bayes Classifier*” menunjukkan tingkat akurasi sebesar 82%. Hasil tersebut dianggap cukup baik dalam menganalisis sentimen (Firdaus, 2024). Selanjutnya, penelitian berjudul “*Analisis Sentimen Kepuasan Pengguna terhadap Layanan Streaming Mola Menggunakan Algoritma Random Forest*” menggunakan 520 dataset dengan pembagian data 90:10 dan menghasilkan tingkat akurasi sebesar 98% (Nanda et al., 2022). Di sisi lain, penelitian oleh Natalya Br Sidauruk berjudul “*Penggunaan Metode SVM dan Random Forest untuk Analisis Sentimen Ulasan Pengguna terhadap KAI Access di Google Play Store*” menggunakan 655 ulasan. Dalam penelitian tersebut, algoritma SVM mencapai tingkat akurasi sebesar 97%, sedangkan algoritma Random Forest mendapatkan akurasi sebesar 93% (Sidauruk et al., 2023).

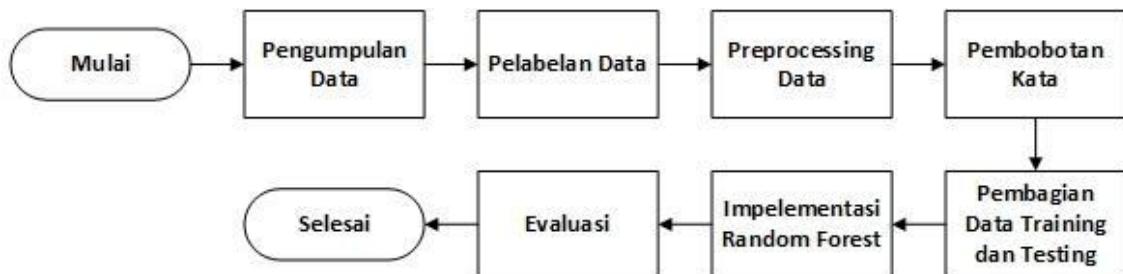
Penelitian terdahulu telah mengkaji berbagai pendekatan dalam analisis sentimen terhadap penggunaan aplikasi Alfagift menggunakan metode Naïve Bayes. Namun, faktor akurasi prediksi menjadi poin penting dalam memilih algoritma yang paling sesuai untuk tujuan ini. Dalam studi ini, peneliti menggunakan pendekatan lain untuk mendapatkan hasil prediksi yang maksimal dan pada hasil akhir studi ini peneliti akan membandingkan dengan penelitian sebelumnya.

Tujuan utama studi ini adalah untuk mengetahui seberapa akurat algoritma klasifikasi Random Forest dalam menganalisis sentimen pengguna aplikasi Alfagift dan menentukan klasifikasi sentimen positif atau negatif berdasarkan ulasan aplikasi tersebut. Random Forest dipilih karena, menurut Pahlevi et al. (2023) Random Forest merupakan algoritma dengan performa yang sangat baik. Pentingnya penelitian ini bagi *developer* atau perusahaan adalah dapat membantu mereka dalam meningkatkan produk dan layanan berdasarkan umpan balik yang diberikan oleh pelanggan secara tulus dan spesifik. Penelitian ini diharapkan dapat memberikan kemajuan pada aplikasi tersebut dan mempermudah masyarakat dalam berbelanja *online* dengan pelayanan yang maksimal.



2. METODE PENELITIAN

Tahapan penelitian adalah serangkaian langkah yang dijalankan oleh peneliti secara teratur dan terencana dalam melaksanakan proses penelitian (Hasibuan et al., 2024). Secara lebih rinci, tahapan yang kami gunakan meliputi pengumpulan data, pelabelan data, *preprocessing* data, pembobotan kata, pembagian data *training* dan *testing*, implementasi Random Forest, serta evaluasi. Pada Gambar 1 dipaparkan secara rinci proses penelitian.



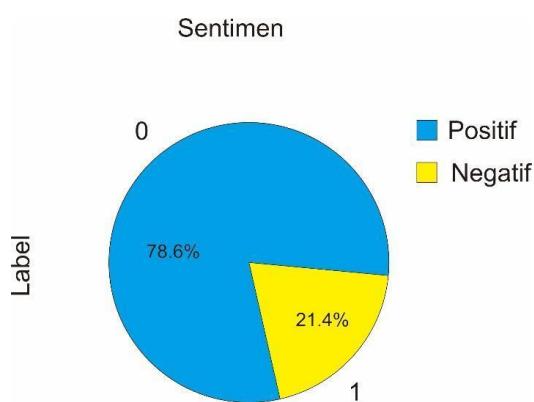
Gambar 1 Tahapan Penelitian

2.1 Pengumpulan Data

Langkah pertama dalam penelitian ini yaitu pengumpulan data, metode *web scraping* digunakan pada tahapan ini. *Web scraping* adalah proses mengambil data semi terstruktur dari suatu situs web atau aplikasi (Styawati et al., 2021). Pengumpulan data dilakukan menggunakan Google Colab dan bahasa pemrograman Python, di mana sebanyak 4.379 ulasan dari Google Play Store berhasil dikumpulkan dan disimpan dalam format CSV.

2.2 Pelabelan Data

Data hasil *scraping* tidak bisa langsung dilakukan analisis perlu di lakukan pelabelan terlebih dahulu pada data tersebut (Syakir & Hasan, 2023). Di mana hasil dari proses pelabelan yaitu di dapatkan kelas positif dan kelas negatif. Berdasarkan data ulasan yang ada, terdapat 3.443 ulasan dengan sentimen positif dan 935 ulasan dengan sentimen negatif. Persentase ulasan yang telah dikategorikan sebagai sentimen positif dan negatif divisualisasikan dalam Gambar 2 yang menunjukkan distribusi label tersebut.



Gambar 2 Diagram Sentimen

2.3 Preprocessing Data

Tahapan *preprocessing* adalah proses paling penting dikarenakan pada step inilah yang menentukan kualitas dari data yang digunakan. *Preprocessing* adalah metode untuk *cleaning* dan mengatur data yang semula acak acakan sehingga menjadi data yang rapi dan relevan yang



efektif (Arista et al., 2024). Adapun tahapan pada proses *preprocessing* meliputi *cleaning*, *normalization*, *tokenizing*, *stopwords removal*, dan *stemming*.

2.4 Pembobotan Kata

Selanjutnya, teknik pembobotan kata yang akan digunakan adalah Term Frequency-Inverse Document Frequency (TF-IDF), yang merupakan transformasi data teks menjadi angka. Kecerdasan komputasional dalam pembelajaran mesin dan pembelajaran mendalam secara umum bekerja dengan baik dan efektif ketika data yang diproses berbentuk angka. Dalam setiap studi di bidang *text mining*, khususnya yang berfokus pada analisis sentimen, perubahan data dari teks menjadi angka merupakan langkah yang wajib. Proses ini dikenal dengan istilah pembobotan kata. Beberapa metode yang dapat digunakan dalam pembobotan kata antara lain Bag of Words (BoW), N-gram, Word2Vec, dan TF-IDF (Kadhim, 2018). TF-IDF merupakan metode yang digunakan untuk mengelola teks dengan memberikan bobot pada kata-kata dalam sebuah dokumen. Pembobotan ini bertujuan untuk mengetahui kosa kata terpenting atau yang paling sering muncul dalam sebuah dokumen maupun kumpulan dokumen. Nilai frekuensi kata yang muncul dalam dokumen disebut Term Frequency (TF). Rumus untuk menghitung nilai TF ditunjukkan pada Pers. (1).

Kata penghubung serta kata-kata umum seringkali memiliki nilai TF yang tinggi, namun tidak mengandung makna yang signifikan. Terdapat metode alternatif yang dapat digunakan, yaitu Inverse Document Frequency (IDF). Metode ini bekerja dengan mencari kosa kata yang frekuensi kemunculannya jarang dalam sebuah dokumen, kemudian memberikan bobot pada kata-kata tersebut. Rumus untuk menghitung nilai IDF dijabarkan pada Pers. (2). Dalam rumus tersebut N adalah total jumlah dokumen, dan n adalah jumlah dokumen yang mengandung kata tersebut. Dengan mengalikan hasil TF dan IDF, maka diperoleh nilai TF-IDF.

$$TF = \frac{(jumlah Kemunculan Kata dalam dokumen)}{(jumlah kata dalam dokumen)} \quad (1)$$

$$IDF = \log \frac{N}{n} \quad (2)$$

2.5 Pembagian Data *Training* dan Data *Testing*

Perbandingan antara data pelatihan dan data pengujian dapat mempengaruhi hasil klasifikasi, karena pemilihan rasio yang tidak tepat bisa mengakibatkan penurunan akurasi pada model yang dibangun (Pradana, 2016). Proses pembagian data bertujuan untuk memisahkan data yang telah melalui tahap *preprocessing* menjadi data latih (*train*) dan data uji (*test*). Pembagian ini dilakukan dalam lima variasi persentase yang berbeda, guna mengevaluasi perbandingan mana yang menghasilkan kinerja terbaik. Setiap set data yang terpisah kemudian akan diterapkan pada algoritma *machine learning* dalam tahap pemodelan.

2.6 Implementasi Random Forest

Random Forest adalah metode komputasi yang melibatkan kumpulan pohon keputusan (*decision trees*) yang mengandalkan suara mayoritas untuk menentukan hasil akhir. Setiap pohon keputusan umumnya terdiri dari *root node*, *internal node*, dan *leaf node*. Metode seperti *entropy*, *information gain*, *gini index*, dan *gini split* digunakan untuk mencari akar pohon keputusan. Untuk membangun banyak pohon, teknik *bagging* diterapkan dengan mengambil sampel secara acak dari fitur dan baris data untuk proses pelatihan pohon keputusan. Biasanya, *entropy*, *information gain*, atau *gini split index* digunakan untuk mengevaluasi pohon keputusan dan prediksi yang didasarkan pada suara mayoritas (Anjani et al., 2023). Rumus yang digunakan untuk mendapatkan nilai *entropy*, *gain*, *gini index*, dan *gini split* dituliskan pada Pers. (3) sampai (6).



$$Entropy(S) = \sum_{i=1}^n -P_i \log_2 p_i \quad (3)$$

$$Gain(S, F) = Entropy(S) - \sum_{i=1}^n -\frac{|S_i|}{|S|} \times Entropy(S_i) \quad (4)$$

$$Gini(S) = 1 - \sum_{i=1}^n (P_i)^2 \quad (5)$$

$$Gini_{split} = \sum_{i=1}^n -\frac{|S_i|}{|S|} \times Gini(S_i) \quad (6)$$

Nilai entropi yang mengukur ketidakpastian atau impurity suatu dataset disebut $Entropy(S)$. Rumus untuk menghitung nilai Entropy ditunjukkan pada Pers. (3). Dalam rumus tersebut S adalah himpunan data yang dianalisis, P_i adalah probabilitas kelas ke- i dalam dataset, dan n adalah jumlah total kelas. Semakin tinggi nilai Entropy, semakin tidak teratur atau beragam data tersebut.

Nilai pengurangan entropi setelah pembagian dataset berdasarkan fitur disebut $Gain(S, F)$. Rumus untuk menghitung Gain ditunjukkan pada Pers. (4). Di dalamnya S adalah dataset utama sebelum split, F adalah fitur yang digunakan untuk split, $Entropy(S)$ adalah entropi sebelum split, $|S_i|$ adalah jumlah data pada subset hasil split, $|S|$ adalah jumlah total data, dan $Entropy(S_i)$ adalah entropi pada masing-masing subset. Nilai Gain yang tinggi menandakan fitur tersebut memberikan informasi signifikan untuk memisahkan data.

Nilai *impurity* dataset dengan pendekatan indeks Gini disebut $Gini(S)$. Rumus untuk menghitung Gini ditunjukkan pada Pers. (5). Dalam rumus ini S adalah himpunan data yang dianalisis dan P_i adalah probabilitas kelas ke- i dalam dataset. Semakin tinggi nilai Gini, semakin beragam data, sedangkan nilai rendah menunjukkan data lebih homogen.

Rata-rata bobot *impurity* Gini setelah dataset dibagi ke dalam subset disebut $Gini_{split}$. Rumus untuk menghitung $Gini_{split}$ ditunjukkan pada Pers. (6). Di dalamnya S adalah dataset utama sebelum split, S_i adalah subset hasil pembagian, $|S_i|$ adalah jumlah data pada subset, $|S|$ adalah jumlah total data, dan $Gini(S_i)$ adalah nilai Gini *impurity* pada setiap subset. Nilai $Gini_{split}$ ini digunakan untuk mengevaluasi kualitas pembagian dataset.

2.7 Evaluasi

Pada penghujung studi ini dilakukan evaluasi performa algoritma Random Forest dalam pengklasifikasian teks sentimen. Evaluasi ini bertujuan untuk menilai sejauh mana model mampu memberikan hasil klasifikasi yang baik. Evaluasi ini akan berpatokan pada hasil visualisasi *confusion matrix* dan grafik ROC.

3. HASIL DAN PEMBAHASAN

3.1 Pengumpulan Data

Proses pengumpulan ulasan aplikasi Alfagift dari Google Play Store dilakukan melalui penerapan metode *web scraping*, menggunakan bahasa pemrograman Python yang diintegrasikan dengan alat Google Colab untuk memudahkan prosesnya. Data yang diproleh dalam proses ini berjumlah 4.379 ulasan dengan beberapa atribut yaitu *review Id*, *userName*, *userImage content*, *score*, dan lain-lain. Kemudian dilakukan seleksi pada atribut tersebut dengan menyisakan *content* dan *score* saja. Detail hasil proses ditampilkan pada Tabel 1.



Artikel ini didistribusikan mengikuti lisensi Atribusi-NonKomersial CC BY-NC sebagaimana tercantum pada <https://creativecommons.org/licenses/by-nc/4.0/>.

Tabel 1 Hasil Pengumpulan Data

No.	Score	Content	
1	1	Respon pelayanannya lama banget, masa sampe 2 hari chat pelanggan baru di bales. Heran deh, malah tercepat admin IG alfamart.	
2	1	Stok dikranjang masih ada giliran mau belanja diklik berubah jadi stok kosong 😊	
...	
4379	5	Keren belanja jdi mudah dan dapat point	

3.2 Pelabelan Data

Proses pemberian label dilakukan dengan menggunakan *rating* ulasan sebagai tolak ukur dalam pelabelan sentimen. Di mana *rating* 1, 2, 3 dikelompokkan ke dalam sentimen negatif (1) kemudian *rating* 4 dan 5 dikelompokkan ke dalam sentimen positif (0). Dari proses pelabelan ini, diperoleh 3.443 ulasan dengan sentimen positif dan 935 ulasan dengan sentimen negatif. Rincian hasil dari proses ini disajikan dalam Tabel 2.

Tabel 2 Hasil Pelabelan

No.	Score	Content	Label
1	1	Respon pelayanannya lama banget, masa sampe 2 hari chat pelanggan baru di bales. Heran deh, malah tercepat admin IG alfamart.	1
2	1	Stok dikranjang masih ada giliran mau belanja diklik berubah jadi stok kosong 😊	1
...
4379	5	Keren belanja jdi mudah dan dapat point	0

3.3 Preprocessing Data

Setelah proses pelabelan data selesai, tahapan selanjutnya yang dilakukan adalah *preprocessing* data sebelum masuk ke tahap pemodelan. Tahapan ini mencakup pembersihan data dari nilai-nilai yang hilang, duplikat, atau tidak relevan, serta penyesuaian format agar data dapat diproses dengan optimal oleh algoritma pembelajaran mesin. Tujuan dari *preprocessing* ini adalah untuk memastikan bahwa data berada dalam kondisi yang bersih dan terstruktur, sehingga dapat meningkatkan akurasi dan efisiensi model yang akan dibangun.

Pada proses *cleaning*, karakter selain huruf akan dihilangkan. Ini termasuk simbol pagar (#), *link*, dan tanda baca contohnya garis miring (/), titik koma (;), titik dua (:), dan titik (.). Seluruh teks juga dikonversi menjadi huruf kecil, menghapus mention akun (@), menstandarisasi kata dengan menghilangkan apostrof ('') dan karakter alfanumerik yang menempel, serta menghapus spasi ganda agar diperoleh data teks yang lebih bersih untuk analisis selanjutnya. Detail hasil proses ditampilkan pada Tabel 3.

Tabel 3 Hasil Cleaning Data

No.	Score	Content	Label
1	1	respon pelayanannya lama banget masa sampe hari chat pelanggan baru di bales heran deh malah tercepat admin ig alfamart	1
2	1	Stok dikranjang masih ada giliran mau belanja diklik berubah jadi stok kosong	1
...
4379	5	keren belanja jdi mudah dan dapat point	0

Dalam proses *normalization*, dilakukan standarisasi ejaan kata-kata yang dalam penulisannya sering kali di singkat dan ejaan yang tidak baku menjadi baku. Dengan contoh kata “syia”



kemudian dinormalisasi menjadi kata “saya”, kata “dgn” menjadi “dengan”, kemudian kata “kpn” yang diubah menjadi “kapan” dan seterusnya. Tujuan dari proses normalisasi ini adalah untuk menseragamkan supaya nanti pada saat proses pengujian didapatkan hasil yang akurat. Detail hasil proses ditampilkan pada Tabel 4.

Tabel 4 Hasil Normalisasi

No.	Score	Content	Label
1	1	respon pelayanannya lama banget masa sampe hari chat pelanggan baru di bales heran deh malah tercepat admin ig alfamart	1
2	1	Stok dikranjang masih ada giliran mau belanja diklik berubah jadi stok kosong	1
...
4379	5	Keren belanja jadi mudah dan dapat point	0

Dalam proses *tokenizing*, teks dipotong menjadi token berupa kata-kata individual menggunakan RegexpTokenizer. Teks dipisahkan pada setiap kata yang terdiri dari karakter alfanumerik (\w+). Dengan *tokenizing* ini, teks yang semula berupa kalimat utuh dapat dipecah menjadi kata per kata, memudahkan analisis leksikal. Hasil *tokenizing* ditambahkan ke dalam kolom baru ‘text_token’ pada *data frame* untuk digunakan pada tahap analisis selanjutnya. Detail hasil proses ditampilkan pada Tabel 5.

Tabel 5 Hasil Tokenizing

No.	Score	Content	Label
1	1	[‘respon’, ‘pelayanannya’, ‘lama’, ‘banget’, ‘masa’, ‘sampe’, ‘hari’, ‘chat’, ‘pelanggan’, ‘baru’, ‘di’, ‘bales’, ‘heran’, ‘deh’, ‘malah’, ‘tercepat’, ‘admin’, ‘ig’, ‘alfamart’]	1
2	1	[‘stok’, ‘dikranjang’, ‘masih’, ‘ada’, ‘giliran’, ‘mau’, ‘belanja’, ‘diklik’, ‘berubah’, ‘jadi’, ‘stok’, ‘kosong’]	1
...
4379	5	[‘keren’, ‘belanja’, ‘jadi’, ‘mudah’, ‘dan’, ‘dapat’, ‘point’]	0

Dalam proses penghapusan *stopwords*, kata-kata yang termasuk dalam daftar *stopwords* dalam bahasa Indonesia dihapus dari hasil tokenisasi agar tidak mempengaruhi proses analisis selanjutnya. *Stopwords* adalah kata-kata yang sangat umum seperti ‘yang’, ‘dan’, ‘di’, ‘dll’ yang kurang memiliki makna penting secara leksikal. Dengan menghapus *stopwords*, diharapkan kata-kata yang tersisa lebih mewakili substansi dan topik teks sebenarnya. Daftar *stopwords* bahasa Indonesia diunduh dari corpus NLTK. Selanjutnya, kata-kata hasil tokenisasi disaring dengan menghapus *stopwords*, menghasilkan kumpulan kata yang lebih relevan dan bermakna untuk analisis. Rincian hasil dari proses ini disajikan dalam Tabel 6.

Tabel 6 Hasil Stopwords Removal

No.	Score	Content	Label
1	1	[‘respon’, ‘pelayanannya’, ‘banget’, ‘sampe’, ‘chat’, ‘pelanggan’, ‘bales’, ‘heran’, ‘deh’, ‘tercepat’, ‘admin’, ‘ig’, ‘alfamart’]	1
2	1	[‘stok’, ‘dikranjang’, ‘giliran’, ‘belanja’, ‘diklik’, ‘berubah’, ‘stok’, ‘kosong’]	1
...
4379	5	[‘keren’, ‘belanja’, ‘jadi’, ‘mudah’, ‘point’]	0

Dalam proses *stemming*, kata-kata hasil tokenisasi diubah ke bentuk kata dasarnya menggunakan *stemmer* bahasa Indonesia dari library Sastrawi. Proses *stemming* diterapkan pada setiap kata untuk memperoleh bentuk dasarnya dengan menghapus imbuhan seperti awalan, akhiran, dan sisipan. Sebagai contoh, kata “bermain” akan diubah menjadi “main” setelah melalui proses *stemming*. Dengan *stemming* ini, kata-kata dengan akar yang sama dapat



dianggap sebagai kata yang sama meskipun bentuk infleksinya berbeda. Hasil *stemming* kemudian diubah kembali dari kumpulan token menjadi kalimat utuh agar siap digunakan pada proses analisis selanjutnya, seperti pembentukan model *topic modeling*. Detail hasil proses ditampilkan pada Tabel 7.

Tabel 7 Hasil Stemming

No.	Score	Content	Label
1	1	['respon', 'layan', 'banget', 'sampe', 'chat', 'pelanggan', 'bales', 'heran', 'deh', 'cepat', 'admin', 'ig', 'alfamart']	1
2	1	['stok', 'dikranjang', 'gilir', 'belanja', 'klik', 'ubah', 'stok', 'kosong']	1
...
4379	5	['keren', 'belanja', 'jadi', 'mudah', 'point']	0

3.4 Pembobotan Kata

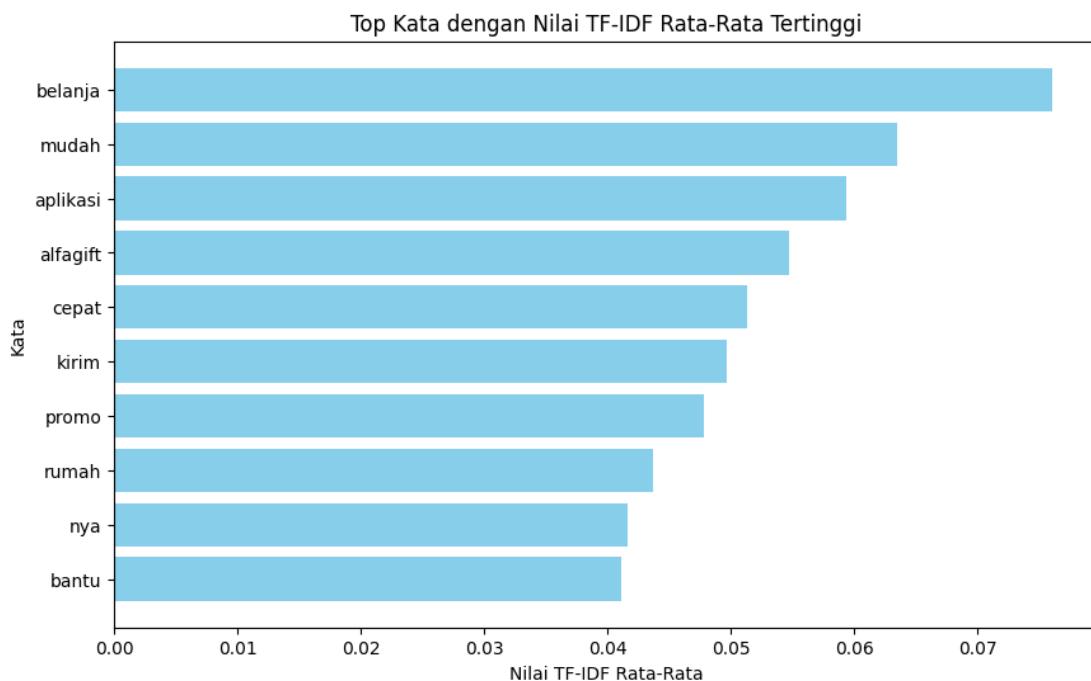
Kata kata dengan dinilai paling tinggi dianggap paling penting dalam menentukan topik sebuah dokumen. Detail hasil proses pembobotan ditampilkan pada Gambar 3. Pada Gambar 3, kode 1 menunjukkan penomoran dari baris yang sedang dilakukan pengolahan, kode 2 merupakan nomor unik yang diberikan kepada setiap kata dalam baris, dan kode 3 menunjukkan skor, yang merupakan hasil dari pembobotan kata yang merupakan hasil dari perhitungan TF-IDF. Detail kosa kata yang sering muncul dalam bentuk diagram *plot* ditampilkan dalam Gambar 4.

1	2	3
(0, 2172)		0.09594938526700864
(0, 4962)		0.07874241691835011
(0, 3406)		0.06631816875103022
(0, 3627)		0.1245389886334698
(0, 4176)		0.3531106263389961
(0, 310)		0.16370596653203248
(0, 3371)		0.14010975643633627
(0, 1103)		0.1866114148507248
(0, 871)		0.09872429561247213
(0, 592)		0.030256407576428574
(0, 5060)		0.29185735493308645
(0, 3991)		0.15606889541962135
(0, 2805)		0.1506503072325116
(0, 1095)		0.1016277752962363
(0, 905)		0.1375946479329907
(0, 5006)		0.1133374810756203
(0, 223)		0.07483145993873176
(0, 917)		0.12339710357673728
(0, 2052)		0.06829078210914138
(0, 375)		0.14644732370671837
(0, 4562)		0.09933193975738468
(0, 2373)		0.20056364355219883
(0, 1606)		0.09446161208447418
(0, 5064)		0.20970804583648062
(0, 1232)		0.16370596653203248

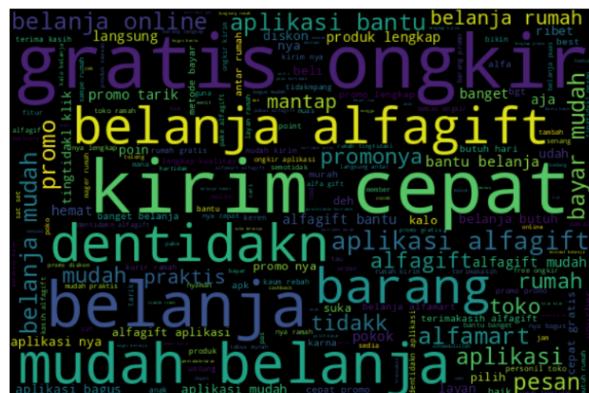
Gambar 3 Hasil TF-IDF

Selain menggunakan diagram plot, terdapat visualisasi yang menggunakan *wordcloud* di mana visualisasi dibagi menjadi kelas positif dan kelas negatif. Di dalam Gambar 5, ditampilkan visualisasi dari ulasan yang bersifat positif dan pada Gambar 6, terdapat visualisasi dari ulasan yang memiliki sifat negatif. Berdasarkan Gambar 5 dan 6, kata-kata seperti “gratis ongkir”, “cepat”, “belanja”, “Mudah”, dan “promo” banyak muncul pada ulasan positif. Sedangkan kata-kata seperti “tolong”, “kecewa”, “kirim”, dan “aplikasi” sering muncul pada ulasan negatif.





Gambar 4 Diagram Kosa Kata



Gambar 5 Wordcloud Positif



Gambar 6 Wordcloud Negatif



3.5 Pembagian Data *Training* Dan Data *Testing*

Setelah tahap *preprocessing* data selesai, langkah selanjutnya adalah memisahkan data menjadi dua bagian, yakni data pelatihan (*train*) dan data pengujian (*test*), yang akan dipisah sebanyak lima kali dan akan dilakukan eksperimen dengan algoritma Random Forest. Rasio perbandingan data dapat dilihat pada Tabel 8. Berdasarkan Tabel 8, terlihat bahwa 4.379 data dibagi antara data pelatihan dan pengujian dengan berbagai perbandingan, mulai dari 90% untuk data pelatihan dan 10% untuk data pengujian, hingga 50% untuk masing-masing data pelatihan dan pengujian.

Tabel 8 Persentase Perbandingan Data *Train* dan Data *Test*

Jumlah Data	Data <i>Train</i>	Data <i>Test</i>	Persentase
4.379	3.941	438	90:10
4.379	3.503	876	80:20
4.379	3.065	1.314	70:30
4.379	2.627	1.752	60:40
4.379	2.189	2.190	50:50

3.6 Implementasi Random Forest dan Evaluasi

Setelah melalui beberapa proses dari pengumpulan data, pelabelan, *preprocessing*, pembobotan kata serta pembagian data *training* dan *testing*, tahap selanjutnya yaitu penerapan algoritma Random Forest. Proses evaluasi dilakukan dengan menggunakan *confusion matrix* yang berfungsi memberikan sebuah visualisasi mengenai hasil kerja yang dilakukan oleh algoritma tersebut. Sebelum melihat hasil dari *confusion matrix* model terbaik, mari kita lihat terlebih dulu hasil akurasi dari lima rasio *train* dan *test* sebelumnya untuk melihat model terbaik menggunakan pembagian data di rasio berapa, yang dapat dilihat pada Tabel 9.

Tabel 9 Hasil Evaluasi Agoritma Random Forest

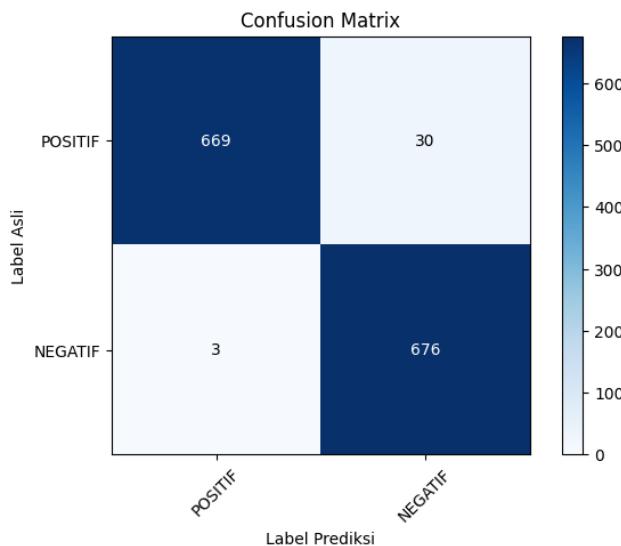
Split	Akurasi	%
90:10	0,965	96,5%
80:20	0,976	97,6%
70:30	0,969	96,9%
60:40	0,967	96,7%
50:50	0,961	96,1%

Dari Tabel 9 didapatkan bahwa rasio pembagian 80:20 menghasilkan akurasi terbaik dari rasio lainnya. Setelah dilakukan pengujian didapatkan nilai akurasi terbaik sebesar 0,976 dengan nilai 669 True Positive (TP), dan 676 True Negative (TN). Di sisi lain, terdapat 30 False Positive (FP) dan 3 False Negative (FN). Detail hasil ditampilkan dalam Gambar 7. Dikarenakan nilai dari *True Negatif* dan *True Positif* lebih besar dari nilai *False Positif* dan *False Negatif*, maka *confusion matrix* tersebut menunjukkan tingkat akurasi yang tinggi sebesar 0,976. Penjabaran perhitungan dari nilai akurasi ditunjukkan pada Pers. (7).

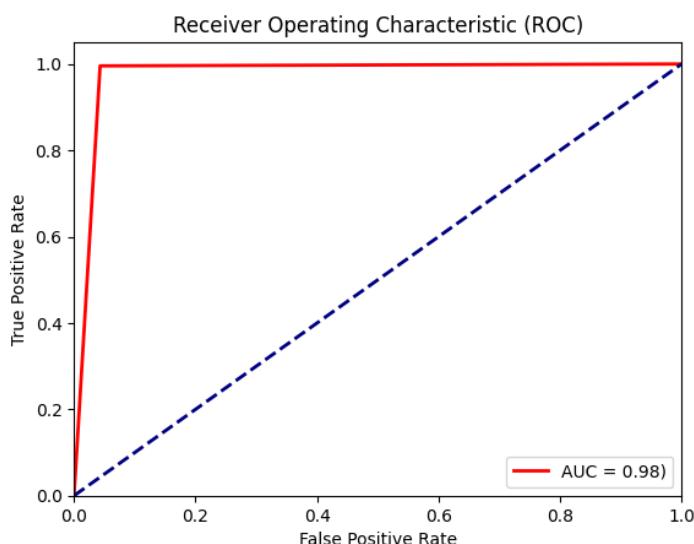
Untuk mengevaluasi hasil prediksi, Grafik Receiver Operating Characteristic (ROC) umumnya digunakan untuk menilai seberapa efektif model dalam membedakan antara kelas positif dan negatif. Kurva ROC dihasilkan dengan membandingkan nilai True Positive Rate (TPR) dan False Positive Rate (FPR) yang dihitung dari *confusion matrix*. Area di bawah kurva, juga dikenal sebagai Area Under Curve (AUC), menunjukkan kinerja model yang digunakan. Detail hasil ditampilkan dalam Gambar 8. Berdasarkan grafik ROC didapatkan nilai AUC sebesar 0,98. Menurut Wilujeng et al. (2023) kategori *excellent* memiliki nilai AUC 0,90–1. Berdasarkan tingkat nilai diagnosa pada kurva ROC pada penelitian ini dapat dikelompokkan ke dalam kelompok *excellent classification*.



$$\begin{aligned}
 Accuracy &= \frac{TP + TN}{TP + TN + FP + FN} \times 100 \\
 &= \frac{669 + 676}{669 + 676 + 30 + 3} \times 100\% \\
 &= \frac{1345}{1378} \times 100\% \\
 &= 97,6\%.
 \end{aligned} \tag{7}$$



Gambar 7 Confusion Matrix



Gambar 8 Grafik ROC

4. KESIMPULAN

Berdasarkan hasil pengujian model dan analisis sentimen terhadap ulasan pengguna aplikasi Alfagift, model *machine learning* Random Forest terbukti mampu mengklasifikasikan teks dengan sangat baik dan efektif, dengan nilai keakuratan sebesar 97,6% dan nilai AUC sebesar 0,98, yang masuk ke dalam kategori *excellent classification*. Dibandingkan dengan penelitian sebelumnya yang juga membahas topik yang sama menggunakan model Naïve Bayes, model Random Forest



terbukti lebih baik dari pada model Naïve Bayes. Secara keseluruhan, mayoritas ulasan ditemukan bersifat positif, mencapai persentase sebesar 78,6%, sementara ulasan negatif sebesar 21,4%.

Ulasan negatif umumnya berisi keluhan pengguna terkait pelayanan yang lambat, stok yang tidak sesuai, gagal login, dan masalah lain yang mengganggu pengalaman pengguna dalam menggunakan aplikasi Alfagift. Oleh karena itu, perhatian khusus perlu diberikan oleh pengembang Alfagift untuk memperbaiki dan meningkatkan kualitas aplikasi tersebut. Dengan memanfaatkan hasil analisis sentimen, pengembang dapat lebih baik dalam menargetkan perbaikan pada area yang diperlukan, sehingga dapat meningkatkan kepuasan pengguna. Hasil penelitian ini diharapkan dapat meningkatkan kenyamanan pengguna dalam berbelanja *online* di Alfagift. Dalam studi selanjutnya, diharapkan dapat memberikan peningkatan akurasi yang lebih baik lagi, terutama dengan penggunaan algoritma optimasi seperti Particle Swarm Optimization.

DAFTAR PUSTAKA

- Anjani, A. F., Anggraeni, D., & Tirta, I. M. (2023). Implementasi Random Forest Menggunakan SMOTE untuk Analisis Sentimen Ulasan Aplikasi Sister for Students UNEJ. *Jurnal Nasional Teknologi dan Sistem Informasi*, 9(2), 163–172. <https://doi.org/10.25077/TEKNOSI.v9i2.2023.163-172>
- Arista, D., Sibaroni, Y., & Prasetyo, S. S. (2024). Sentiment Analysis on Twitter (X) Related to Relocating the National Capital Using the IndoBERT Method Using Extraction Features of Chi-Square. *Jurnal Media Informatika Budidarma*, 8(1), 403–411. <https://doi.org/10.30865/mib.v8i1.7198>
- Helmi, A., Komaladewi, R., Sarasi, V., & Yolanda, L. (2023). Characterizing Young Consumer Online Shopping Style: Indonesian Evidence. *Sustainability*, 15(5), Article ID: 3988. <https://doi.org/10.3390/su15053988>
- Fadli, M. N., SJ, A. S., & Nafiah, N. (2023). Mekanisme Penggunaan Member Card Alfagift dalam Jual Beli Perspektif Hukum Islam (Studi Kasus di Alfamart Ponorogo). *Social Science Academic*, 1(2), 29–36. <https://doi.org/10.37680/ssa.v1i2.3173>
- Firdaus, A. (2024). *Analisis Sentimen pada Aplikasi Alfagift Menggunakan Metode Naïve Bayes Classifier* [UIN Syarif Hidayatullah]. <https://repository.uinjkt.ac.id/dspace/handle/123456789/76766>
- Hanifa, F., Putri, C. H., N, A. F., & Wulansari, A. (2023). Analisis Penerimaan Aplikasi Alfagift di Kota Surabaya Menggunakan Metode Technology Acceptance Model. *Jurnal Sains dan Teknologi (JSIT)*, 3(2), 233–244. <https://doi.org/10.47233/jsit.v3i2.835>
- Hasibuan, S. S., Angraini, A., Saputra, E., & Megawati, M. (2024). Sentimen Analisis Terhadap Fitur Tiktok Shop Menggunakan Naïve Bayes dan K-Nearest Neighbor. *Jurnal Media Informatika Budidarma*, 8(1), 303–311. <https://doi.org/10.30865/mib.v8i1.7238>
- Jin, Y., Cheng, K., Wang, X., & Cai, L. (2023). A Review of Text Sentiment Analysis Methods and Applications. *Frontiers in Business, Economics and Management*, 10(1), 58–64. <https://doi.org/10.54097/fbem.v10i1.10171>
- Kadhim, A. (2018). An Evaluation of Preprocessing Techniques for Text Classification. *International Journal of Computer Science and Information Security (IJCSIS)*, 16(6), 22–32. <https://sites.google.com/site/ijcsis/all-volumes-issues/vol-16-no-6-june-2018>
- Komarudin, A., & Hilda, A. M. (2024). Analisis Sentimen Ulasan Aplikasi Identitas Kependudukan Digital pada Play Store Menggunakan Metode Naïve Bayes. *Computer Science (COSCIENCE)*, 4(1), 28–36. <https://doi.org/10.31294/coscience.v4i1.2955>
- Nanda, S., Mualfah, D., & Fitri, D. A. (2022). Analisis Sentimen Kepuasan Pengguna Terhadap Layanan Streaming Mola Menggunakan Algoritma Random Forest. *Jurnal Aplikasi Teknologi Informasi dan Manajemen (JATIM)*, 3(2), 210–219. <https://doi.org/10.31102/jatim.v3i2.1592>
- Pahlevi, O., Amrin, A., & Handrianto, Y. (2023). Implementasi Algoritma Klasifikasi Random Forest untuk Penilaian Kelayakan Kredit. *Jurnal Infortech*, 5(1), 71–76. <https://doi.org/10.31294/infortech.v5i1.15829>



- Perdana, S. A., Florentin, S. F., & Santoso, A. (2022). Analisis Segmentasi Pelanggan Menggunakan K-Means Clustering Studi Kasus Aplikasi Alfagift. *Sebatik*, 26(2), 446–457. <https://doi.org/10.46984/sebatik.v26i2.1991>
- Pradana, M. (2016). Klasifikasi Bisnis E-Commerce di Indonesia. *MODUS*, 27(2), 163–174. <https://doi.org/10.24002/modus.v27i2.554>
- Sidauruk, N., Riza, N., & Fatonah, Rd. N. S. (2023). Penggunaan Metode SVM dan Random Forest untuk Analisis Sentimen Ulasan Pengguna Terhadap KAI Access di Google Play Store. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 7(3), 1901–1906. <https://doi.org/10.36040/jati.v7i3.6899>
- Styawati, S., Hendrastuty, N., & Isnain, A. R. (2021). Analisis Sentimen Masyarakat Terhadap Program Kartu Prakerja pada Twitter dengan Metode Support Vector Machine. *Jurnal Informatika: Jurnal Pengembangan IT*, 6(3), 150–155. <https://doi.org/10.30591/jpit.v6i3.2870>
- Syakir, A., & Hasan, F. N. (2023). Analisis Sentimen Masyarakat Terhadap Perilaku Korupsi Pejabat Pemerintah Berdasarkan Tweet Menggunakan Naive Bayes Classifier. *Jurnal Media Informatika Budidarma*, 7(4), 1796–1805. <https://doi.org/10.30865/mib.v7i4.6648>
- Utami, A. F., Ekaputra, I. A., Japutra, A., & Van Doorn, S. (2022). The Role of Interactivity on Customer Engagement in Mobile E-Commerce Applications. *International Journal of Market Research*, 64(2), 269–291. <https://doi.org/10.1177/14707853211027483>
- Wicaksono, N. B. A. (2022). *Analisis Sentimen Tingkat Kepuasan Pengguna BPJS dengan Metode SentiStrength* [Universitas Atma Jaya Yogyakarta]. <https://repository.uajy.ac.id/id/eprint/27165/>
- Wilujeng, D. T., Fatekurohman, M., & Tirta, I. M. (2023). Analisis Risiko Kredit Perbankan Menggunakan Algoritma K-Nearest Neighbor dan Nearest Weighted K-Nearest Neighbor. *Indonesian Journal of Applied Statistics*, 5(2), 142–148. <https://doi.org/10.13057/ijas.v5i2.58426>

