

Analisis Sentimen pada Ulasan Aplikasi Wondr di Play Store dengan Metode *Naïve Bayes*

Suci Nurhikmah^{1*}, Romi Ramadani², Gandung Triyono³

^{1,2,3}Universitas Budi Luhur, Indonesia

*email: romiramadhaan@gmail.com

Info Artikel

Dikirim: 14 Juli 2025

Diterima: 11 Agustus 2025

Diterbitkan: 30 November 2025

Kata kunci:

Analisis Sentimen;

Aplikasi;

Machine Learning;

Mobile Banking;

Naive Bayes Classifier.

ABSTRAK

Perkembangan teknologi digital mendorong sektor perbankan untuk terus berinovasi, termasuk dalam pengembangan layanan *mobile banking* yang lebih responsif terhadap kebutuhan nasabah. Bank Negara Indonesia (BNI) merespons tuntutan tersebut dengan meluncurkan aplikasi Wondr sebagai pengganti BNI *Mobile Banking* sebelumnya, yang menuai beragam tanggapan dari pengguna di *Google Play Store*. Penelitian ini dilakukan untuk memahami opini atau persepsi pengguna mengenai aplikasi Wondr, guna mengevaluasi opini pengguna yang bisa dijadikan dasar untuk mengambil langkah strategis dalam pengembangan layanan digital BNI. Pendekatan yang digunakan melibatkan analisis sentimen menggunakan *Naive Bayes Classifier*, yang dieksekusi melalui bahasa pemrograman *Python*. Data yang dianalisis terdiri dari 27.124 ulasan pengguna. Hasil klasifikasi menunjukkan bahwa 52,9% ulasan bersifat positif, 39,9% negatif, dan 7,2% netral. Model *Naive Bayes* memberikan akurasi sebesar 82%, namun performanya masih lemah dalam mengidentifikasi sentimen netral, yang dievaluasi berdasarkan metrik presisi, *recall*, dan *F1-Score*. Temuan ini menunjukkan bahwa secara umum aplikasi Wondr diterima dengan baik oleh pengguna, meskipun masih terdapat aspek-aspek yang perlu ditingkatkan. Penelitian ini merekomendasikan eksplorasi algoritma terhadap algoritma klasifikasi alternatif, seperti *Random Forest*, *SVM* dan metodologi *Deep Learning*, serta penerapan teknik *SMOTE* untuk mengatasi ketidakseimbangan data seperti ulasan netral dalam klasifikasi sentiment penelitian ini.

1. PENDAHULUAN

Kemajuan teknologi informasi dan komunikasi telah memberikan pengaruh yang signifikan terhadap berbagai aspek kehidupan, seperti sektor perbankan [1]. Transformasi digital yang semakin pesat menuntut industri perbankan untuk terus berinovasi guna menambah kualitas layanan dan memenuhi kebutuhan nasabah. Salah satu inovasi yang menjadi strategi utama dalam industri ini adalah pengembangan layanan digital, seperti *mobile banking*. *Mobile banking* hadir sebagai solusi memungkinkan nasabah untuk memanfaatkan layanan perbankan tanpa perlu mengunjungi kantor cabang, layanan tersedia setiap saat dan dari lokasi mana pun. Dengan fitur-fitur yang semakin beragam seperti transfer uang, pembayaran tagihan, investasi, serta manajemen keuangan, *mobile banking* telah menjadi bagian yang tidak dapat dipisahkan dari kehidupan masyarakat modern. Layanan *M-banking* memungkinkan nasabah untuk melakukan berbagai transaksi dan pengecekan keuangan, seperti melihat dari riwayat transaksi, memeriksa saldo terkini, dan melakukan aktivitas lainnya, tanpa harus mengunjungi lokasi bank atau mesin ATM [2].

Di Indonesia, perkembangan *mobile banking* telah menunjukkan dampak yang signifikan dalam meningkatkan efisiensi operasional dan kepuasan pelanggan. Namun, meskipun memberikan banyak manfaat, tidak semua nasabah merasa puas dengan layanan yang tersedia. Beberapa tantangan yang masih sering dihadapi adalah kecepatan dalam melakukan transaksi, kestabilan sistem, serta pengalaman pengguna (*user experience*) yang belum optimal. Faktor-faktor ini dapat berdampak negatif terhadap kepuasan dan loyalitas pelanggan, sehingga bank perlu terus melakukan evaluasi dan penyempurnaan layanan mereka. Bank BNI menjadi salah satu termuka di Indonesia yang berdiri sejak tahun 1946, Bank Negara Indonesia (BNI) terus berupaya meningkatkan inovasi dalam layanan digital nya. Sebelumnya, BNI menghadirkan aplikasi *mobile banking* BNI Mobile sebagai solusi perbankan digital bagi nasabahnya. Dilihat dari opini pada *Play Store* pengguna/nasabah mengalami kesulitan saat melakukan pemindaian wajah (aktivasi akses login), kesulitan dalam proses transaksi, serta menemukan masalah kegagalan transfer meskipun saldo telah berkurang dan tampilan yang kurang menarik pada BNI Mobile. Dalam upaya memberikan pengalaman yang lebih personal, intuitif, dan berbasis teknologi terkini, BNI meluncurkan aplikasi terbaru bernama Wondr. Aplikasi ini dikembangkan untuk memenuhi kebutuhan masyarakat di era digital dengan menghadirkan layanan yang lebih canggih dan *user-friendly*.

Menariknya, kondisi penelitian terbaru menunjukkan bahwa analisis ulasan aplikasi perbankan di toko aplikasi digital menjadi metode yang semakin banyak digunakan untuk mengevaluasi performa dan persepsi layanan keuangan digital. Misalnya, studi-studi mutakhir memanfaatkan teknik *machine learning* untuk mengolah ribuan ulasan pengguna guna mengetahui tren sentiment, efektivitas fitur, serta kelemahan dari suatu layanan. Salah satu metode yang efektif untuk keluhan pengguna dapat dipelajari melalui pendekatan analisis sentiment [3]. Analisis sentiment adalah cara untuk mengenali dan mengelompokkan pendapat dalam sebuah teks, apakah bersifat positif, negatif, atau netral [4][5]. Dengan demikian, analisis sentiment dapat menjadi alat ukur yang relevan dan strategis dalam meningkatkan kepuasan serta loyalitas pelanggan di industri perbankan.

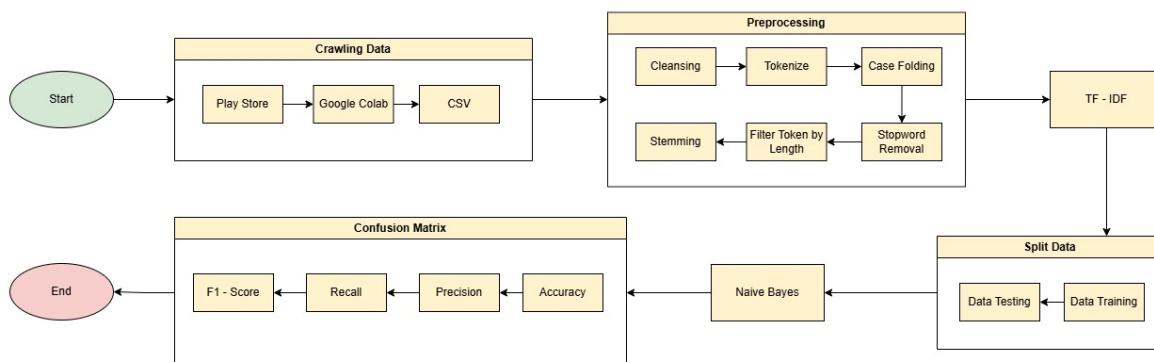
Penelitian yang dilakukan oleh [6] menganalisis sentiment pengguna aplikasi BRIMO pada *Play Store* dengan memanfaatkan metode *Naive Bayes* untuk menganalisis data ulasan positif (1.012 data) dibandingkan ulasan negatif (894 data), meskipun terdapat cukup banyak keluhan. Metode *Naive Bayes* menghasilkan akurasi 84,52%, *recall* 87,62% dan *precision* 82,51%, menunjukkan efektivitas nya dalam klasifikasi sentimen. Selanjutnya penelitian yang dilakukan oleh [7] melakukan sentiment analisis terhadap ulasan pengguna pada aplikasi DANA di *Play Store* dengan algoritma *Naive Bayes* dengan fitur seleksi *Information Gain*. Tujuan dari penelitian ini adalah mengevaluasi sentimen klasifikasi serta mengukur pengaruh penerapan IG terhadap performa model. Data ulasan dikategorikan menjadi positif dan negatif berbasis leksikon, kemudian dilakukan pembobotan, seleksi fitur, dan pembagian data (80% latihan, 20% uji). Dua model dibandingkan NBC tanpa seleksi fitur (1.106 fitur) dan NBC-IG (536 fitur, nilai $IG > 0.01$). Hasil menunjukkan bahwa model NBC-IG lebih unggul dengan akurasi 85,09% (naik 2,18%), presisi 85,79% (naik 1,83%), dan *recall* 92,09% (naik 1,86%) dibandingkan model NBC standar (akurasi 82,91%, presisi 83,96%, *recall* 90,23%). Studi ini membuktikan bahwa seleksi fitur IG tidak hanya mengurangi dimensi fitur hingga 51,5%, tetapi juga meningkatkan kinerja klasifikasi sentiment. Penelitian berikutnya dilakukan [8] menganalisa sentiment pada aplikasi Blu BCA dengan metode *Naive Bayes* dengan 1000 dataset memiliki akurasi sebesar 85.31% studi memberikan NB memberikan kemampuan klasifikasi yang baik. Kemudian penelitian yang dilakukan oleh [9] melakukan analisis sentiment pada “Aplikasi Bank Digital Pada Google Play Store Menggunakan Algoritma *Naive Bayes*” memberikan akurasi sebesar 94% pada aplikasi *seabank* di *Play Store* dengan total 1000 data.

Salah satu teknik pembelajaran mesin yang cukup sering digunakan dalam analisis sentiment adalah *Naive Bayes Classifier* (NBC). Metode ini dikenal sederhana dan mudah diterapkan, namun tetap mampu memberikan hasil yang cukup baik di berbagai kasus. Meski begitu, kelemahan utama dari *Naive Bayes* adalah anggapannya bahwa setiap fitur bersifat independen satu sama lain, yang bisa memengaruhi tingkat akurasi pada kondisi tertentu [10]. Penelitian lain [11] juga menunjukkan bahwa metode ini mampu menghasilkan klasifikasi sentiment dengan tingkat *accuracy* mencapai 80%, sehingga membuktikan bahwa *Naive Bayes* cukup efektif untuk menganalisis teks, khususnya pada data dari media sosial.

Berdasarkan latar belakang tersebut, Penelitian ini bertujuan untuk menganalisis sentiment dari ulasan aplikasi *mobile banking* Wondr di *Play Store* dengan mengelompokkan sentimen positif, netral dan negatif menggunakan pendekatan *Naïve Bayes*. Hasil dari penelitian ini diharapkan dapat memberikan gambaran objektif mengenai persepsi pengguna terhadap aplikasi Wondr, serta menjadi dasar bagi pihak bank untuk melakukan evaluasi, perbaikan, dan pengembangan lanjutan terhadap layanan *mobile banking* agar lebih kompetitif dan berorientasi pada kebutuhan nasabah.

2. METODE PENELITIAN

Penelitian ini diawali dengan mengumpulkan ulasan pengguna aplikasi Wondr dari *Play Store*, yang dipilih karena merupakan *platform* populer bagi pengguna *Android* untuk mengunduh aplikasi dan memberikan ulasan. Pengambilan data dilakukan melalui *scrapping* menggunakan *Google Colab*, alat berbasis *cloud* yang efisien dalam menangani data dalam jumlah besar. Untuk mendukung analisis di tahap berikutnya, data hasil pengumpulan disimpan dalam format CSV. Tahap selanjutnya adalah *preprocessing*, yang dirancang untuk membersihkan dan mengatur data sebelum melakukan analisis. Proses ini mencakup beberapa langkah utama, yaitu *cleansing* data (menghapus kata atau simbol yang tidak diperlukan), *tokenization* (membagi teks menjadi kata - kata), *case folding* (konversi menjadi format huruf kecil), *stopword removal* (menghilangkan kata yang umum dan tidak bermakna signifikan), *filtering token by length* (menyaring kata berdasarkan panjangnya), dan *stemming* (konversi kata ke bentuk dasarnya). Setelah *preprocessing* selesai, data dianalisis menggunakan algoritma NB dengan fitur TF-IDF untuk mengukur bobot kata dalam teks. *Wordcloud* digunakan untuk memvisualisasikan kata - kata yang paling sering muncul dalam ulasan pengguna. Evaluasi kinerja model dalam mengklasifikasikan sentimen pengguna dilakukan menggunakan metode *confusion matrix*, dengan metrik seperti akurasi, presisi, *Recall*, dan *F1-Score* guna mengukur kinerja model dalam klasifikasi sentiment pengguna. Pada proses ini memberikan gambaran sistematis dalam pengolahan teks untuk menghasilkan model analisis sentimen yang optimal. Alur diagram penelitian ini Seperti ditunjukkan pada Gambar 1.



Gambar 1. Diagram alur penelitian

2.1. Pengumpulan Data

Pada penelitian ini, data dikumpulkan dari *opini* aplikasi Wondr di *Play Store*, menghasilkan dataset berjumlah 27.124 ulasan. Proses pengambilan data dilakukan pada 05 Maret 2025 menggunakan *Google Colab*, yang memungkinkan akses dan pengolahan data dalam skala besar secara *efisien*. Data yang dikumpulkan selanjutnya disimpan dalam file format CSV sehingga memudahkan analisis tahap berikutnya. Ilustrasi tahapan pengumpulan data ditampilkan pada Gambar 2.



Gambar 2. Tahapan mengumpulkan Data

Pengambilan data dilakukan dengan *Google Colab*, yang menggunakan *library* dari *google_play_scraper* untuk mengakses ulasan aplikasi dari *Play Store*. Dalam proses ini, *App ID* aplikasi Wondr (id.bni.wondr) dimasukkan sebagai parameter utama untuk mengidentifikasi aplikasi yang akan diambil datanya. Setelah terhubung, *Google Colab* dapat mengambil data ulasan dari aplikasi Wondr, termasuk informasi seperti teks ulasan, *rating*, tanggal ulasan, dan identitas pengguna secara anonim.

2.2. Tahapan *Preprocessing*

Preprocessing data merupakan fase krusial dalam klasifikasi, yang berfungsi menghilangkan kata-kata yang tidak diperlukan dari kumpulan data. Langkah ini membantu mengatasi masalah seperti data/kata berulang, data yang mengganggu, atau data yang tidak lengkap. Berikut ini adalah langkah-langkah dalam proses pembersihan teks [12].

1) *Cleansing*

Cleansing merupakan proses penghapusan karakter - karakter yang tidak dalam kategori dalam alfabet dari teks, dengan tujuan untuk mengurangi elemen-elemen yang tidak relevan, tanda baca, link, *hashtag* (#), *mention* (@), emotikon, dan lainnya agar lebih mudah dianalisis [13].

2) *Tokenize*

Tokenisasi berarti memotong teks menjadi bagian - bagian kecil, seperti kata atau kalimat, agar komputer bisa lebih mudah memahaminya dan mengolahnya di tahap selanjutnya [14].

3) *Case Folding*

Case folding merupakan langkah mengubah semua huruf atau kata dalam teks menjadi huruf kecil. Pada proses ini, tanda baca dan angka juga dihapus, sehingga hanya tersisa huruf a hingga z saja [15].

4) *Stopword Removal*

Stopword removal merupakan proses menghapus kata - kata yang tidak terlalu berpengaruh terhadap makna teks. Kata-kata seperti 'dan', 'di', 'tetapi', atau 'atau' biasanya dihilangkan karena tidak menambah arti penting atau emosi dalam ulasan [16].

5) *Filter Length by Token*

Filter Token by Length ini bertujuan menyaring kata berdasarkan jumlah hurufnya. Kata yang terlalu pendek atau terlalu panjang akan dihapus agar data yang tersisa lebih sesuai untuk dianalisis oleh model [17].

6) *Stemming*

Dalam proses stemming, kata-kata diubah ke bentuk paling dasar agar lebih mudah dianalisis. Proses ini membantu menyamakan kata-kata yang berbeda bentuk tapi punya arti yang sama [18].

2.3. *Labeling*

Dataset yang diperoleh melalui proses *scraping* masih bersifat *unsupervised*, artinya belum memiliki *label*. Oleh karena itu, data tersebut akan diberi label sebelum dilakukan pemodelan. Dalam analisis sentimen, biasanya digunakan tiga jenis label, yaitu label positif, label negatif, dan label netral. Agar pelabelan data dilakukan secara tepat, penelitian ini menggunakan script Python untuk memberi label berdasarkan *rating*: nilai 1–2 dianggap positif, nilai 3 sebagai netral, dan nilai 4–5 sebagai negatif.

2.4. *Term Frequency Inverse Document Frequency (TF-IDF)*

TF-IDF merupakan salah satu dari teknik pembobotan yang digunakan dalam pengolahan text untuk menilai sejauh mana suatu kata dianggap penting dalam konteks sebuah dokumen dibandingkan dengan keseluruhan koleksi dokumen yang ada [17]. Dalam metode ini, *Term Frequency* (TF) berfungsi sebagai metrik untuk menilai frekuensi kemunculan suatu kata dalam suatu dokumen, sehingga mencerminkan signifikansinya dalam kaitannya dengan konten dokumen tersebut. Sebaliknya, *IDF (Inverse Document Frequency)* mengevaluasi kelangkaan atau keunikan suatu kata dalam kumpulan dokumen yang lebih luas. Kombinasi kedua nilai ini membantu untuk meminimalkan bobot kata - kata yang sering muncul di berbagai dokumen (seperti kata - kata umum), semakin jarang sebuah kata muncul, tapi tetap penting, maka bobotnya akan semakin besar. Metode TF-IDF terbukti sangat efektif dalam meningkatkan kualitas analisis teks, terutama

dalam aplikasi seperti klasifikasi teks, *clustering*, dan pencarian informasi, di mana pemahaman tentang relevansi kata terhadap dokumen sangat penting.

2.5. Penerapan Algoritma *Naïve Bayes*

Penelitian ini memakai metode klasifikasi *Naïve Bayes* untuk memperkirakan kategori teks dengan menghitung probabilitasnya menurut *teorema Bayes*. Ada tiga jenis utama dari *Naïve Bayes*, yaitu *GaussianNB*, yang digunakan untuk data numerik dengan distribusi normal; *MultinomialNB*, yang cocok untuk data diskrit seperti frekuensi kemunculan kata dalam teks; dan *BernoulliNB*, yang diterapkan pada data *biner* yang menunjukkan apakah suatu kata ada atau tidak ada dalam teks. Model ini dikenal karena kecepatannya, kesederhanaannya, dan efektivitasnya dalam menganalisis serta mengklasifikasikan data teks dalam jumlah besar.

2.6. Penilaian Kinerja Model

Penilaian dilakukan untuk evaluasi seberapa baik kinerja model dengan melihat akurasi metode melalui *confusion matrix* dengan nilai akurasi, dan nilai presisi dari setiap model. Setelah data uji dilakukan banding dengan data latih, model akan memberikan daftar kategori yang disebut prediksi pada kelas. Hasil prediksi ini kemudian dicocokkan dengan kelas asli dari data uji yang sebelumnya tersembunyi. Dari sini, metrik seperti menilai akurasi, presisi, *recall*, dan *F1-score* dapat dihitung dan dianalisis untuk menilai seberapa akurat model dalam melakukan klasifikasi [4].

Berikut ini rumus *confusion matrix* untuk mengukur kinerja model:

$$\text{Rumus Akurasi} = \frac{(TP+TN)}{(TP+TN+FP+FN)} \quad (1)$$

$$\text{Rumus Presisi} = \frac{TP}{(TP+FP)} \quad (2)$$

$$\text{Rumus Recall} = \frac{TP}{(TP+FN)} \quad (3)$$

$$\text{Rumus F1-Score} = 2 \times \frac{(\text{Precision} \times \text{Recall})}{(\text{Precision} + \text{Recall})} \quad (4)$$

True Positive (TP) menunjukkan dari jumlah data yang berhasil diprediksi sebagai nilai positif dan memang benar-benar termasuk dalam kategori positif. *True Negative* (TN) menunjukkan prediksi yang tepat terhadap data negatif. Sementara itu, *False Positive* (FP) adalah ketika model salah memprediksi data negatif sebagai positif. Sebaliknya, *False Negative* (FN) terjadi saat data positif diklasifikasikan secara keliru sebagai *negative*.

3. HASIL DAN PEMBAHASAN

3.1 Pengumpulan Data

Pengumpulan data dilakukan menggunakan *platform Google Colab* dan *Python* sebagai bahasa pemrogramannya. Dalam tahapan proses ini, library *google_play_scraper* dimanfaatkan untuk mengambil data pada tanggal 5 Maret 2025. Aplikasi yang diambil ulasannya memiliki App ID *id.bni.wondr*. Setelah berhasil dikumpulkan, data disimpan dalam file bernama *dataset_wondr* dengan format CSV. Secara keseluruhan, data didapatkan sebanyak 27.124. Proses mengumpulkan data ditampilkan pada Gambar 3.

```
# Import Data
df = pd.read_csv('/content/drive/MyDrive/sentimen_analysis/dataset_wondr.csv', sep=",", encoding='utf-8')
df
```

	content	score	at
0	Lama banget fitur setor tunainya, kalo lama mu...	1	2024-10-02 00:48:02
1	Saya mau masuk aplikasi aja susah banget kaya ...	1	2024-10-02 00:06:16
2	Sedikit masukan saja. Pada verifikasi wajah to...	4	2024-10-01 23:53:24
3	Halo min kenapa si eror system terus saat veri...	4	2024-10-01 23:32:29
4	Bintang 3 dulu, soalnya keterangan transaksi b...	3	2024-10-01 23:02:49

Gambar 3. Proses pembacaan data

Berikut ini adalah hasil *scrapping* pada ulasan Wondr di *Play Store* yang berhasil dilakukan dalam bentuk CSV terdapat tiga kolom yaitu *content*, *score* dan *at* yang digunakan pada penelitian ini. Hasil pengumpulan data dengan cara *scrapping* seperti terlihat pada Tabel 1.

Tabel 1. Hasil *scrapping* data

content	score	at
lebih oke lagi ya bank bni good	5	2025-03-05 08:06:33
proses transaksi praktis dan pembuatan rekening sangat mudah	4	2025-03-05 08:06:14
oke sangat membantu	5	2025-03-05 08:00:30
keren cepet skr mah	5	2025-03-05 08:00:09
kualitas masih dibawah bni mobile banking misalnya untuk topup ewallet dana nominalnya hanya maksimal jthari	3	2025-03-05 04:34:57

3.2 Pre-processing

Setelah data diperoleh, tahap berikutnya adalah membersihkan dan memproses data tersebut menggunakan *Python*. Tahapan *preprocessing* ini merupakan tahapan yang penting terutama pada bagian *cleansing* terhadap data yang dikumpulkan untuk mengeliminasi kesalahan yang terdapat pada data. Pada penelitian ini proses *pre-processing* dilakukan proses *replace*, *trim* dan *remove*.

1) *Cleansing Data*

Karakter non-alfabet seperti (?), (!), dan (,) dihilangkan pada tahap ini. Hasil data yang telah dibersihkan disajikan pada Tabel 2.

Tabel 2. Proses hasil *cleansing*

Pra <i>Cleansing</i>	Pasca <i>Cleansing</i>
Hallo, izin bertanya. Untuk sisa uang di rekening apakah dikenakan biaya admin (potongan saldo)??	hallo izin bertanya untuk sisa uang di rekening apakah dikenakan biaya admin potongan saldo

2) *Tokenize*

Pada tahap ini, kalimat diuraikan menjadi kata-kata individual. Contoh hasilnya disajikan di Tabel 3.

Tabel 3. Proses hasil *Tokenize*

Pra <i>Tokenize</i>	Pasca <i>Tokenize</i>
Halo min kenapa si eror system terus saat verifikasi wajah gak bisa masuk efeknya jadi telah mencoba berkali dianggap kebanyakan dan coba lagi besok tolong perbaiki	['halo', 'min', 'kenapa', 'si', 'eror', 'system', 'terus', 'saat', 'verifikasi', 'wajah', 'gak', 'bisa2', 'masuk', 'efeknya', 'jadi', 'telah', 'mencoba', 'berkali2', 'dianggap', 'kebanyakan', 'dan', 'coba', 'lagi', 'besok', 'tolong', 'perbaiki']

3) *Case Folding*

Pada tahap *case folding*, semua teks diubah menjadi huruf kecil untuk memastikan format yang konsisten. Proses ini membantu pemrosesan data berikutnya. Hasil *case folding* disajikan pada Tabel 4.

Tabel 4. Data hasil *Case Folding*

Pra Case Folding	Pasca Case Folding
Kenapa kode Gmail yang di kirim selalu tidak bisa di loginkan sampai sekarang tidak bisa di pakai apk nya	kenapa kode gmail yang di kirim selalu tidak bisa di loginkan sampai sekarang tidak bisa di pakai apk nya

4) *Stopword Removal*

Proses *stopword removal* diantaranya menghilangkan kata-kata berhenti melibatkan penghilangan kata-kata yang tidak memiliki makna substansial untuk analisis, termasuk konjungsi dan preposisi. Contoh hasil dari *stopword removal* disajikan pada Tabel 5.

Tabel 5. Proses pada *Stopword Removal*

Pra Stopword Removal	Pasca Stopword Removal
Tidak ada fitur tarik dan setor tunai di mesin ATM / cardless transaction. Belum bisa menggantikan BNI Mobile	['fitur', 'tarik', 'setor', 'tunai', 'mesin', 'ATM', 'cardless', 'transaction', 'bisa', 'menggantikan', 'BNI', 'Mobile']

5) Tahap *Filter Token* berdasarkan *Length*

Pada tahapan *filter token by length*, kata yang memiliki panjang huruf yang dianggap tidak penting akan dihilangkan. Hasilnya disajikan pada Tabel 6.

Tabel 6. Data hasil dari *Filter Token*

Pra Filter Length by Token	Pasca Filter Length by Token
Knpa yg ttp nggk bisa di verifikasi kode otpnya pdahal udh benar tpi ttp.aja salah	['Knpa', 'nggk', 'bisa', 'verifikasi', 'kode', 'otpnya', 'pdahal', 'benar', 'salah']

6) *Stemming*

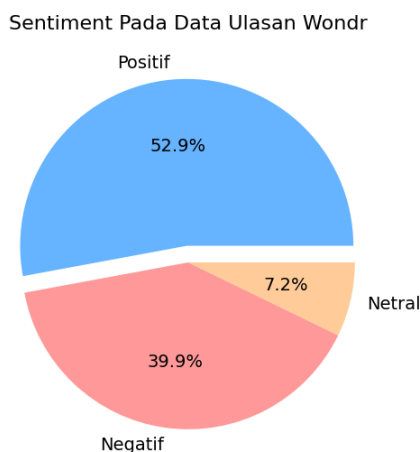
Proses *Stemming* adalah proses konversi kata ke dalam bentuk dasarnya atau akar kata. Contoh hasil dari proses *stemming* ini disajikan pada Tabel 7.

Tabel 7. Proses hasil dari *Stemming*

Pra Stemming	Pasca Stemming
aplikasi membingungkan soal verifikasi wajah selalu tidak bisa sudah di coba berkali-kali di tempat terang tetap tidak bisa	['aplikasi', 'bingung', 'soal', 'verifikasi', 'wajah', 'selalu', 'tidak', 'bisa', 'sudah', 'coba', 'kali', 'tempat', 'terang', 'tetap', 'tidak', 'bisa']

3.3 *Labelling*

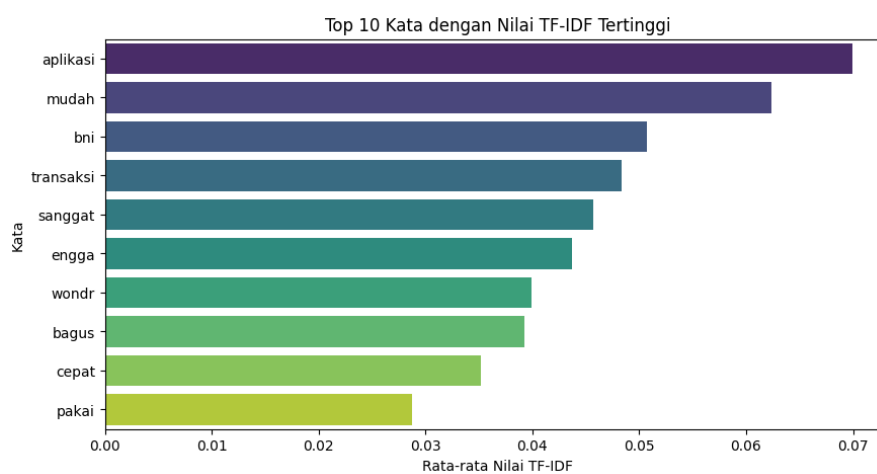
Data yang telah selesai pada tahapan *pre-processing* akan diberi *label* untuk sentimen negatif, netral, dan positif. Tujuan pemberian label ini adalah untuk mengklasifikasikan data berdasarkan emosi atau *opini* yang terkandung di dalamnya, sehingga memudahkan dalam analisis lebih lanjut. Setelah melakukan *labelling* pada data, selanjutnya kita bisa melihat visualisasi dari data berdasarkan sentimen yang ada dengan tujuan untuk memudahkan pemahaman pola sentimen yang terkandung dalam data, serta memberikan wawasan yang lebih jelas mengenai distribusi sentimen dengan label positif, netral dan negatif dalam konteks tertentu. Visualisasi ini juga membantu dalam analisis tren dan pengambilan keputusan berbasis data, seperti pada Gambar 4.



Gambar 4. Visualisasi Sentiment

3.4 Pembobotan TF-IDF

TF-IDF merupakan teknik untuk menilai pentingnya kata pada sebuah dokumen dibanding dokumen lain. Proses ini mengubah kata menjadi angka agar bisa diproses oleh mesin. Dalam penelitian ini, digunakan TF-IDF *Vectorizer* untuk konversi teks menjadi bentuk nilai numerik. Proses TF-IDF dilakukan dalam dua tahap. Pertama, menghitung jumlah frekuensi kemunculan sebuah kata muncul pada dokumen. Kedua, menilai seberapa penting kata tersebut dengan melihat sebarannya di seluruh dokumen. Perhitungan ini membantu menentukan kata-kata yang paling berpengaruh dalam sebuah dokumen. Sepuluh kata teratas dengan frekuensi kemunculan tertinggi disajikan pada Gambar 5.



Gambar 5. Kata dengan nilai TF-IDF tertinggi

Vektorisasi teks di lakukan dengan memanfaatkan library *TfidfVectorizer* yang miliki pustaka *scikit learn*, yang mengubah teks pada kolom “*NORMALIZED*” menjadi bentuk numerik. Proses perhitungan representasi ini didasarkan pada nilai bobot kata menggunakan rumus TF-IDF. Nilai bobot yang diperoleh akan menunjukkan tingkat kepentingan suatu kata pada dokumen tertentu, jika dibandingkan dengan keseluruhan koleksi dokumen. Berikut ini adalah hasil dari proses pembobotan TF-IDF dalam bentuk *array* yang telah berhasil dilakukan, disajikan pada Gambar 6 dibawah.

```
[ ] Train_X_Tfidf.toarray()
array([[0.      , 0.      , 0.      , ..., 0.83527056, 0.      ,
        0.      ],
       [0.      , 0.      , 0.      , ..., 0.      , 0.      ,
        0.      ],
       [0.      , 0.      , 0.      , ..., 0.      , 0.      ,
        0.      ],
       ...,
       [0.      , 0.      , 0.      , ..., 0.      , 0.      ,
        0.      ],
       [0.      , 0.      , 0.      , ..., 0.      , 0.      ,
        0.      ],
       [0.      , 0.      , 0.      , ..., 0.      , 0.      ,
        0.      ]])
```

Gambar 6. Pembobotan hasil metrik TF-IDF

Untuk memudahkan melihat kata-kata dengan frekuensi sering muncul dalam penelitian ini, digunakan visualisasi dalam bentuk *word cloud*. Dalam tampilan ini, ukuran setiap kata disesuaikan dengan seberapa sering *frequency* kata tersebut muncul semakin sering muncul, semakin besar tampilannya. Dengan cara ini, kita bisa dengan cepat mengenali kata-kata yang paling menonjol dalam data, seperti yang di sajikan pada Gambar 7.



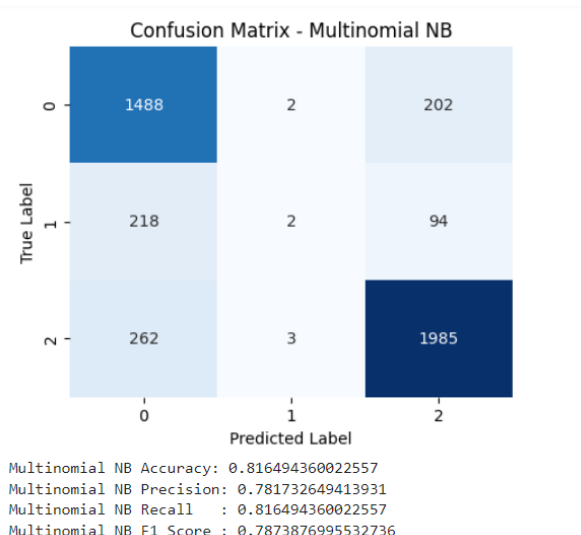
Gambar 7. Worcloud Sentimen

3.5 Model Naïve Bayes

Pada tahap ini, peneliti menggunakan algoritma NB untuk menganalisis ulasan aplikasi Wondr di Play Store. Tujuannya adalah untuk melihat seberapa akurat, tepat (*precision*), dan sensitif (*recall*) hasil analisis sentimen yang diperoleh. Proses perhitungan pada model *Naïve Bayes* membantu menilai bobot setiap kata dalam menentukan apakah ulasan termasuk sentimen negatif, positif, atau netral. Hasil perhitungan ini sangat mempengaruhi tingkat akurasi dari model yang digunakan.

Naive Bayes dipilih sebagai algoritma untuk klasifikasi sentimen pengguna dari ulasan aplikasi Wondr pada *Play Store* di klasifikasi ke dalam tiga label sentimen: yaitu label netral, positif, dan juga negatif. Algoritma *Naive Bayes* digunakan karena cara kerjanya yang menghitung kemungkinan (*probabilitas*) setiap kata berdasarkan data yang ada. Algoritma ini digunakan karena cukup efektif dalam mengolah data teks yang besar dan rumit, dan keunggulannya telah dibuktikan oleh banyak penelitian sebelumnya.

Hasil penerapan model *Naive Bayes* menunjukkan akurasi keseluruhan sebesar 82%, meskipun kinerja pada kategori kelas minoritas (netral) masih perlu perbaikan. Hal ini sesuai dengan temuan yang menyatakan bahwa meskipun *Naive Bayes* efektif dalam menangani data berukuran besar, performanya bisa dipengaruhi oleh distribusi data antar kelas seperti pada Gambar 8.



Gambar 8. Implementasi Model NB

3.6 Evaluasi (*Confusion Matrix*)

Model yang dibangun di evaluasi dengan metrik seperti mendapatkan nilai akurasi, presisi, recall serta F1-score agar dapat diketahui seberapa tepat hasil klasifikasinya. Tabel 8 menyajikan hasil evaluasi dari model *Naïve Bayes* yang digunakan.

Tabel 8. *Confusion Matrix* hasil model

Kelas actual	Positif (0)	Netral (1)	Negatif (2)	Total
Positif	1488	2	202	1692
Netral	218	2	94	314
Negatif	262	3	1985	2250
Total	1968	7	2281	4256

Dari hasil prediksi model, terlihat bahwa jumlah data yang di klasifikasi ke dalam kelas positif dan negatif jauh lebih banyak dibandingkan kelas netral. Ketidakseimbangan ini membuat akurasi model terlihat tinggi, tetapi nilai presisi, *recall*, dan F1-score tidak merata untuk semua kelas. Artinya, meskipun model cukup baik dalam mengenali ulasan positif, model masih kesulitan untuk mengenali ulasan netral dan negatif dengan sama baiknya. Detail hasil evaluasi model dan distribusi data disajikan pada Tabel 9 berikut.

Tabel 9. Evaluasi hasil dari model *Naïve Bayes*

Kelas actual	Nilai
Accuracy	0.82
Precision	(Positif) 0.76, (Netral) 0.29, (Negatif) 0.87
Recall	(Positif) 0.88, (Netral) 0.01, (Negatif) 0.88
F1 - Score	(Positif) 0.81, (Netral) 0.01, (Negatif) 0.88

Hal ini menunjukkan model yang dihasilkan memiliki performa terbaik klasifikasi kelas positif dan negatif dengan nilai akurasi mencapai 82%. Namun, pada kelas netral, perform model cenderung rendah, terutama pada nilai *recall* yang cukup rendah pada kelas tersebut. Ini mengindikasikan bahwa model menunjukkan tingkat akurasi yang tinggi pada label positif dan label negatif, kemampuannya dalam identifikasi kelas yang kurang dominan seperti label netral, masih memerlukan perbaikan agar hasil klasifikasi nya lebih seimbang dan akurat. Model *Naïve Bayes* kurang akurat jika data yang digunakan tidak seimbang, misalnya terlalu banyak ulasan pada salah satu label seperti kelas netral pada penelitian ini, sehingga model cenderung memihak kelas yang jumlahnya lebih banyak. Model ini juga tidak mampu memahami konteks atau arti kata dalam

kalimat secara menyeluruh, hanya mengandalkan frekuensi kemunculan kata. Hal ini perlu ditingkatkan untuk menggunakan metode yang dapat menyeimbangkan data pada label yang minor.

4. KESIMPULAN

Merujuk pada temuan dalam penelitian ini, analisis sentimen terhadap 27.124 opini pengguna aplikasi Wondr di *Play Store* menunjukkan bahwa mayoritas pengguna memberikan tanggapan positif. Sebanyak 52,9% ulasan termasuk dalam kategori positif, 39,9% negatif, dan 7,2% netral. Hal ini mencerminkan penerimaan yang cukup baik dari masyarakat terhadap aplikasi Wondr sebagai layanan *mobile banking* terbaru dari BNI. Penggunaan metode NB terbukti lebih efektif dalam melakukan klasifikasi sentimen, dengan tingkat akurasi mencapai 82%. Performa model juga menunjukkan hasil yang kuat pada sentimen dengan label positif dan negatif, dengan nilai presisi, *recall*, dan *F1-Score* yang cukup tinggi. Pengujian menggunakan *confusion matrix* menunjukkan hasil yang beragam untuk kelas positif, presisi sebesar 76%, *recall* perolehan nilai sebesar 88% dan *F1-Score* sebesar 81%. Temuan ini dapat dimanfaatkan oleh pihak BNI sebagai dasar pertimbangan dalam evaluasi dan pengembangan lanjutan aplikasi Wondr agar semakin sesuai dengan kebutuhan pengguna.

Namun demikian, penelitian ini juga menemukan beberapa kekurangan, khususnya dalam mendeteksi sentimen netral. Hasil klasifikasi menunjukkan bahwa model memiliki performa yang cukup rendah pada kelas netral, dengan *precision* dan *F1-Score* hanya sebesar 1%, serta *recall* yang juga sangat minim. Ini mengindikasikan bahwa model *Naive Bayes* yang digunakan belum optimal dalam menangani data dengan distribusi kelas yang tidak seimbang. Oleh karena itu, perlu dilakukan peningkatan performa model melalui pendekatan *tuning parameter*, serta mempertimbangkan penerapan teknik penyeimbangan data seperti *SMOTE* untuk mengatasi ketimpangan jumlah ulasan antar kelas.

Untuk penelitian yang akan datang, disarankan untuk mencoba algoritma klasifikasi lain yang memiliki performa lebih baik dalam menangani data berupa teks dan sentimen yang kompleks, seperti *Random Forest*, *SVM*, dan metodologi *Deep Learning* seperti *LSTM* atau *BERT*. Selain itu, teknik *resampling* seperti *SMOTE* dapat diterapkan untuk meningkatkan performa model pada kelas yang tidak seimbang, seperti kelas netral yang ditemui dalam penelitian ini.

REFERENSI

- [1] B. Widyaningsih dan T. Ibnu Afan, “Peran Manajemen Resiko Dalam Meningkatkan Ketahanan Bank Syariah Di Era Digital,” *J. Ekon. Perbank. Syariah*, vol. 9, no. 3, hal. 1459–1470, 2024, doi: 10.30651/jms.v9i3.22933.
- [2] M. Rizky Pratama, Y. R. Ramadhan, dan M. A. Komara, “Analisis Sentimen BRImo dan BCA Mobile Menggunakan Support Vector Machine dan Lexicon Based,” *J. Ilm. Tek. Inform. dan Sist. Informas*, vol. 12, no. 3, hal. 1439–1450, 2023.
- [3] D. Munandar, M. Afdal, Z. Zarnelly, dan R. Novita, “Analisis Sentimen Ulasan Pengguna Aplikasi Mobile Banking Menggunakan Algoritma K-Nearest Neighbor,” *J. Teknol. Sist. Inf. dan Apl.*, vol. 7, no. 3, hal. 1309–1318, Jul 2024, doi: 10.32493/jtsi.v7i3.41409.
- [4] I. S. Arfan, S. Fauziah, dan I. Nawangsih, “Analisis Sentimen Terhadap Cyber Bullying di X Menggunakan Algoritma Naïve Bayes,” *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 4, no. 4, hal. 1411–1419, Agu 2024, doi: 10.57152/malcom.v4i4.1550.
- [5] J. Mantik dkk., “Application Of N-Gram On K-Nearest Neighbor Algorithm To Sentiment Analysis Of TikTok Shop Shopping Features,” *J. Mantik*, vol. 6, no. 3, hal. 2685–4236, 2022.
- [6] M. Khoirul, U. Hayati, dan O. Nurdiawan, “Analisis Sentimen Aplikasi Brimo Pada Ulasan Pengguna Di Google Play Menggunakan Algoritma Naive Bayes,” *J. Mhs. Tek. Inform.*, vol. 7, no. 1, 2023.
- [7] Cindy Caterine Yolanda, Syafriandi Syafriandi, Yenni Kurniawati, dan Dina Fitria, “Sentiment Analysis of DANA Application Reviews on Google Play Store Using Naïve Bayes Classifier Algorithm Based on Information Gain,” *UNP J. Stat. Data Sci.*, vol. 2, no. 1, hal. 48–55, Feb 2024, doi: 10.24036/ujsds/vol2-iss1/147.
- [8] W. Wahyudi, R. Kurniawan, dan Y. Arie Wijaya, “Analisis Sentimen Pengguna Terhadap Aplikasi Blu

- Bca Di Playstore Menggunakan Algoritma Naïve Bayes,” *JATI (Jurnal Mhs. Tek. Inform.,* vol. 8, no. 3, hal. 2511–2517, 2024, doi: 10.36040/jati.v8i3.9216.
- [9] A. H. Hakim, A. L. Hananto, F. Nurapriani, dan B. Huda, “Analisis Sentimen Aplikasi Bank Digital Pada Google Play Store Menggunakan Algoritma Naive Bayes,” *J. Informatics Commun. Technol.,* vol. 7, no. 1, hal. 6–12, 2025.
- [10] S. A. Mahmudah dan A. Yudhistira, “Analisis Sentimen Terhadap Cyberbullying pada Platform Media Sosial X Menggunakan Algoritma Naive Bayes,” *J. Pendidik. dan Teknol. Indones.,* vol. 5, no. 1, hal. 189–200, Jan 2025, doi: 10.52436/1.jpti.628.
- [11] N. C. Siregar, R. R. A. Siregar, dan M. Y. D. Sudirman, “Implementasi Metode Naive Bayes Classifier (NBC) pada Komentar Warga Sekolah Mengenai Pelaksanaan Pembelajaran Jarak Jauh (PJJ),” *J. Teknol.,* vol. 3, no. 1, 2020.
- [12] J. Homepage, N. C. Agustina, D. Herlina Citra, W. Purnama, C. Nisa, dan A. Rozi Kurnia, “MALCOM: Indonesian Journal of Machine Learning and Computer Science The Implementation of Naïve Bayes Algorithm for Sentiment Analysis of Shopee Reviews on Google Play Store Implementasi Algoritma Naive Bayes untuk Analisis Sentimen Ulasan Shopee pada Google Play Store,” vol. 2, hal. 47–54, 2022.
- [13] S. G. Kaparang, D. R. Kaparang, dan V. P. Rantung, “Analisis Sentimen New Normal Pada Masa Covid-19 Menggunakan Algoritma Naïve Bayes,” *JOINTER-JOURNAL INFORMATICS Eng.,* vol. 02, no. 01, 2021, [Daring]. Tersedia pada: <https://t.co/CxptQpYGU9>
- [14] V. Zuliana, I. Maulana, U. Singaperbangsa Karawang, J. H. Ronggo Waluyo, K. Telukjambe Timur, dan K. Karawang, “Analisis Sentimen Program Migrasi Tv Digital Menggunakan Algoritma Naive Bayes Dengan Chi Square,” *J. Inf. dan Komput.,* vol. 10, no. 2, 2022.
- [15] T. Ridwansyah, “KLIK: Kajian Ilmiah Informatika dan Komputer Implementasi Text Mining Terhadap Analisis Sentimen Masyarakat Dunia Di Twitter Terhadap Kota Medan Menggunakan K-Fold Cross Validation Dan Naïve Bayes Classifier,” *Media Online,* vol. 2, no. 5, hal. 178–185, 2022, [Daring]. Tersedia pada: <https://djournal.com/klik>
- [16] A. Halim Anshor dan A. Safuwani, “Analisis Sentimen Opini Warganet Twitter Terhadap Tes Screening Genose Pendeteksi Virus Covid-19 Menggunakan Metode Naïve Bayes Berbasis Particle Swarm Optimization,” *JINTEKS,* vol. 5, no. 1, hal. 170–178, 2023.
- [17] A. Erfina, M. Fani Al-shufi, C. Kaler, K. Cisaat, dan K. Sukabumi, “Analisis Sentimen Aplikasi Jasa Kurir Di Play Store Menggunakan Algoritma Naive Bayes,” *J. Sist. Inf. dan Inform. P-ISSN,* vol. 5, hal. 2622–6901, 2022.
- [18] D. Gibson Nababan *dkk.*, “Merancang Analisis Sentimen Berdasarkan Pendapat Pengguna Aplikasi Grab Dengan Menggunakan Bahasa Pemrograman Python,” *SAINTEK J. Sains, Teknol. Komput.,* vol. 1, no. 1, hal. 21–27, 2024.