JURNAL ILMIAH FIFO
Vol. 18 No. 1 (2026) 41-51
DOI: 10.22441/fifo.2026.v18i1.005

P-ISSN 2085-4315
E-ISSN 2502-8332

Perbandingan K-Nearest Neighbor, Naïve Bayes, dan Support Vector Machine pada Analisis Sentimen Ulasan Aplikasi Photomath

Nurfidah Dwitiyanti[1*], Noni Selvia[2], Nur Alamsyah[3], Sukarno Bahat Nauli[4]

[1,2,3]Universitas Indraprasta PGRI
[4]Universitas Satya Negara Indonesia
Email: [1]nurfidah.pulungan@gmail.com, [2]noni.selvia@gmail.com, [3]alamcbr11@gmail.com, [4]sukarnobahat@usni.ac.id
Penulis Korespondensi[*]

(***received*****:** 06-03-26, ***revised*****:** 16-03-26, ***accepted*****:** 11-04-26)

Abstrak

Penggunaan aplikasi pembelajaran matematika seperti Photomath terus meningkat, namun kajian terkait sentimen pengguna, khususnya dalam bahasa Indonesia, masih terbatas. Penelitian ini bertujuan untuk mengklasifikasikan sentimen ulasan pengguna serta membandingkan kinerja algoritma Naïve Bayes, K-Nearest Neighbor (KNN), dan Support Vector Machine (SVM). Tahapan penelitian diawali dengan prapemrosesan teks, Setelah itu, data diproses melalui ekstraksi fitur menggunakan metode Term Frequency–Inverse Document Frequency (TF-IDF). Dataset yang digunakan terdiri dari 42.672 ulasan pengguna Photomath yang diperoleh melalui teknik *web scraping* dari Google Play Store dan diseleksi menggunakan *purposive sampling*. Data kemudian diberi label sentimen, yaitu positif, netral, dan negatif berdasarkan nilai rating. Berdasarkan penelitian, KNN mencatat akurasi 86,61%. Namun, kinerjanya pada kelas netral dan negatif masih belum optimal karena data yang tidak seimbang. Meskipun SMOTE dapat meningkatkan recall, akurasi justru menurun. Sebaliknya, SVM terbukti sebagai algoritma terbaik dengan akurasi 88,94% dan F1-score makro tertinggi. Temuan ini menegaskan bahwa pemilihan algoritma dan strategi data tidak seimbang sangat berpengaruh terhadap performa klasifikasi sentimen.

Kata Kunci: Analisis Sentimen, K-Nearest Neighbor, Naïve Bayes, Support Vector Machine, Photomath

Abstract

The use of math learning apps like Photomath continues to increase, but studies related to user sentiment, particularly in Indonesian, are still limited. This study aims to classify user review sentiment and compare the performance of the Naïve Bayes, K-Nearest Neighbor (KNN), and Support Vector Machine (SVM) algorithms. The research phase begins with text preprocessing. After that, the data is processed through feature extraction using the Term Frequency–Inverse Document Frequency (TF-IDF) method. The dataset used consists of 42,672 Photomath user reviews obtained through web scraping techniques from the Google Play Store and selected using purposive sampling. The data is then labeled with sentiment ratings: positive, neutral, and negative. Based on the study, KNN recorded an accuracy of 86.61%. However, its performance in the neutral and negative classes is still not optimal due to imbalanced data. Although SMOTE can improve recall, accuracy actually decreases. In contrast, SVM proved to be the best algorithm with 88.94% accuracy and the highest macro F1-score. These findings confirm that algorithm selection and imbalanced data strategy significantly influence sentiment classification performance.

Keywords*: Sentiment Analysis, K-Nearest Neighbor, Naïve Bayes, Support Vector Machine, Photomath*

1. PENDAHULUAN

Kemajuan teknologi digital telah memberikan dampak besar terhadap sektor pendidikan, termasuk dalam proses pembelajaran matematika. Salah satu aplikasi yang banyak dimanfaatkan sebagai media belajar adalah Photomath, yang memungkinkan pengguna memperoleh solusi soal matematika secara cepat melalui pemindaian, lengkap dengan langkah penyelesaiannya. Dengan jumlah unduhan yang telah melampaui 100 juta di Google Play Store [1], aplikasi ini menjadi salah satu alat bantu belajar yang cukup populer di kalangan siswa dan pendidik. Meskipun demikian, tanggapan pengguna terhadap aplikasi ini bervariasi, bergantung pada kebutuhan serta pengalaman masing-masing individu. Ulasan pengguna di platform seperti Google Play Store

P-ISSN 2085-4315
E-ISSN 2502-8332

menunjukkan beragam pandangan, mulai dari apresiasi terhadap kemudahan penggunaan hingga kritik terkait keterbatasan fitur berbayar atau kesulitan dalam memahami penjelasan yang diberikan [2].

Analisis sentimen terhadap ulasan tersebut menjadi penting untuk memperoleh gambaran menyeluruh mengenai persepsi pengguna sekaligus memberikan umpan balik bagi pengembang dalam meningkatkan kualitas aplikasi. Analisis sentimen merupakan salah satu teknik dalam text mining yang digunakan untuk mengelompokkan opini ke dalam kategori positif, negatif, atau netral. Salah satu algoritma yang umum digunakan dalam klasifikasi teks adalah K-Nearest Neighbor (KNN), yang dikenal mampu menangani pola data yang kompleks serta menyesuaikan diri dengan data baru [3]. Metode ini bekerja dengan mengidentifikasi sejumlah tetangga terdekat dan menentukan kelas berdasarkan mayoritas, sehingga sesuai untuk analisis sentimen yang memerlukan pemahaman konteks dari teks ulasan [4].

Beberapa Penelitian sebelumnya telah membahas ulasan aplikasi dengan menerapkan analisis sentimen melalui berbagai pendekatan. Penelitian yang dilakukan oleh [5] terkait penggunaan aplikasi Checkmath dengan metode KNN menunjukkan tingkat akurasi yang tinggi. Namun penelitian terebut dilakukan dalam lingkup sekolah tertentu dengan jumlah data yang terbatas, sehingga belum menggambarkan opini pengguna dalam skala yang lebih luas seperti data dari Google Play Store. Penelitian lain oleh [6] penggunaan metode KNN pada aplikasi Shopee dengan 2000 data ulasan dan membagi sentimen menjadi positif, negatif, dan netral. Hasilnya menunjukkan akurasi sebesar 70%, namun penelitian tersebut menegaskan bahwa distribusi kelas yang tidak seimbang memengaruhi nilai recall dan F1-score namun dampak ketidakseimbangan data terhadap performa metode KNN belum dianalisis. Sementara itu, Penelitian terbaru yang dilakukan oleh [2] mengoptimalkan dimensi Word2Vec untuk analisis sentimen Photomath dan membandingkan Random Forest serta SVM. Hasilnya menunjukkan SVM lebih unggul, namun penelitian tersebut tidak menggunakan KNN sebagai fokus utama dan hanya membagi sentimen menjadi dua kategori dengan menghapus ulasan netral dari dataset, sehingga belum sepenuhnya mencerminkan kondisi sentimen yang sebenarnya.

Berdasarkan penelitian sebelumnya, masih terdapat celah kajian, yaitu belum adanya evaluasi kinerja KNN pada ulasan Photomath dengan tiga kategori sentimen serta analisis dampak ketidakseimbangan data dalam skala besar. Tantangan utama meliputi distribusi data yang tidak seimbang dan penggunaan bahasa informal yang memengaruhi akurasi, serta keterbatasan KNN dalam menentukan nilai K optimal dan sensitivitas terhadap *noise* [7], [8]. Penelitian ini bertujuan mengatasi permasalahan tersebut dengan menerapkan KNN pada dataset ulasan berbahasa Indonesia berskala besar, disertai prapemrosesan seperti *stemming* dan *stopword removal*. Evaluasi dilakukan menggunakan metrik *precision*, *recall*, dan *F1-score*, serta menganalisis pengaruh ketidakseimbangan data melalui penerapan SMOTE. Selain itu, kinerja KNN dibandingkan dengan Naïve Bayes dan SVM untuk memperoleh evaluasi yang lebih komprehensif.

Hasil penelitian diharapkan memberikan masukan bagi pengembang dalam meningkatkan kualitas aplikasi serta menjadi referensi pemanfaatan media pembelajaran matematika. Selain itu, penelitian ini berkontribusi dalam pengembangan model analisis sentimen yang lebih andal pada kondisi data tidak seimbang dengan tiga kategori sentimen.

2. METODE PENELITIAN

2.1 Metode Penelitian dan Sumber Data

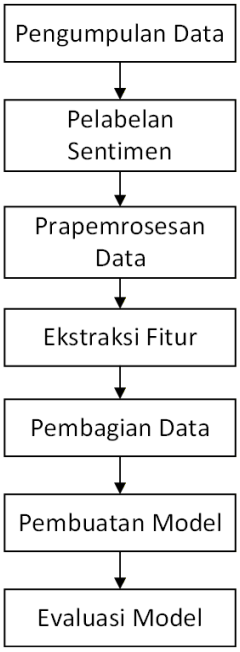
Penelitian ini menggunakan pendekatan kuantitatif berbasis algoritma KNN untuk mengklasifikasikan sentimen ulasan pengguna aplikasi Photomath. Pemilihan KNN didasarkan pada kemampuannya mengelompokkan data teks berdasarkan tingkat kemiripan dengan data latih yang telah terklasifikasi.

Sumber data penelitian berasal dari ulasan pengguna aplikasi Photomath di Google Play Store yang dikumpulkan melalui teknik web scraping. Total data yang terkumpul berjumlah 42.672 ulasan. Kriteria yang digunakan meliputi penggunaan bahasa Indonesia, dan memiliki skor penilaian lengkap. Kemudian dataset ini digunakan sebagai dasar dalam proses klasifikasi sentimen menjadi tiga kategori, yaitu positif, netral dan negatif.

2.2. Tahapan penelitian

Analisis dilakukan secara sistematis melalui beberapa tahapan utama, yaitu pengumpulan data, pelabelan sentimen, prapemrosesan data, ekstraksi fitur, pembagian data, pemodelan, dan evaluasi, serta visualisasi hasil. Untuk memperjelas tahapan penelitian, Gambar 1 menyajikan alur proses analisis sentimen yang dilakukan secara sistematis mulai dari data mentah hingga visualisasi hasil.

JURNAL ILMIAH FIFO
Vol. 18 No. 1 (2026) 41-51
DOI: 10.22441/fifo.2026.v18i1.005
P-ISSN 2085-4315
E-ISSN 2502-8332



Gambar 1. Diagram alur proses analisis sentimen

Setiap tahapan pada Gambar 1 dijelaskan sebagai berikut:

a. Pengumpulan Data

Tahapan pertama pada penelitian ini dimulai dari pengumpulan data ulasan pengguna aplikasi Photomath dengan tujuan untuk memperoleh data ulasan pengguna sebagai bahan analisis sentimen. Data yang dikumpulkan menyajikan opini pengguna untuk memberikan gambaran yang terhadap persepsi pengguna terhadap aplikasi Photomath.

Proses ini dilakukan menggunakan teknik web scraping dengan bantuan pustaka *google-play-scraper* pada bahasa pemrograman Python. Data yang dikumpulkan mencakup teks ulasan dan skor rating yang diberikan oleh pengguna. Selanjutnya data mentah yang diperoleh disimpan dan dipersiapkan untuk diolah pada tahapan beriikutnya.

b. Pelabelan Sentimen

Tahapan kedua yaitu pelabelan sentimen yang bertujuan untuk mengelompokkan data ulasan ke dalam kategori tertentu berdasarkan kecenderungan opini yang terkandung didalamnya. Proses ini dilakukan agar data digunakan dalam model klasifikasi. Kategori sentimen ditentukan berdasarkan skor bintang, yaitu: positif (4–5), netral (3), dan negatif (1–2). Proses ini bertujuan untuk mengubah data mentah menjadi data terstruktur yang siap dianalisis.

c. Prapemrosesan Data

Prapemrosesan data dilakukan untuk meningkatkan kualitas teks agar lebih bersih dan siap digunakan dalam proses klasifikasi, mengingat ulasan pengguna umumnya mengandung kata tidak baku dan simbol yang tidak relevan. Tahapan ini meliputi *text cleaning*, *case folding*, tokenisasi, normalisasi kata, *stopword removal*, serta *stemming* menggunakan Sastrawi untuk memperoleh bentuk dasar kata.

d. Ekstraksi Fitur

Tahap ekstraksi fitur dilakukan untuk mengonversi data teks menjadi bentuk numerik agar dapat diproses oleh algoritma klasifikasi, karena data teks tidak dapat diolah secara langsung tanpa transformasi. Pada penelitian ini, pendekatan yang diterapkan adalah *Term Frequency–Inverse Document Frequency* (TF-IDF), yang menetapkan bobot pada setiap kata berdasarkan seberapa sering kata tersebut muncul dalam suatu dokumen dan dalam keseluruhan kumpulan teks (korpus). Implementasi dilakukan menggunakan pustaka *scikit-learn* pada Python dengan pembatasan jumlah fitur hingga 5.000 untuk meningkatkan efisiensi pemrosesan.

e. Pembagian Data

Tahap berikutnya adalah pemisahan dataset menjadi data latih dan data uji guna mengevaluasi kinerja model serta mengukur kemampuan generalisasi terhadap data baru. Proses ini dilakukan menggunakan fungsi *train_test_split* dari pustaka *scikit-learn* dengan proporsi 80% sebagai data latih dan 20% sebagai data uji. Data

P-ISSN 2085-4315
E-ISSN 2502-8332

latih digunakan untuk membangun model, sedangkan data uji dimanfaatkan untuk mengukur performa model yang dihasilkan.

f. Pembuatan Model

Penelitian ini berfokus pada pembangunan model sentimen menggunakan metode KNN, di mana klasifikasi ditentukan berdasarkan mayoritas suara dari *k* tetangga terdekat. Berdasarkan referensi [9], nilai *k* harus diuji secara empiris karena tidak ada formula tetap untuk menjamin akurasi. Hasil pengujian terhadap beberapa nilai ganjil menunjukkan bahwa $k = 9$ memberikan keseimbangan bias-varians yang paling ideal serta akurasi tertinggi. Untuk memperkuat analisis, performa KNN dibandingkan dengan dua algoritma populer lainnya, yaitu Naïve Bayes dan SVM, guna melihat efektivitas masing-masing dalam mengolah teks ulasan.

g. Evaluasi Model

Untuk menguji kinerja model yang telah dibangun, dilakukan tahapan evaluasi yang berfokus pada kemampuan klasifikasi dalam kondisi data tidak seimbang. Pengukuran performa merujuk pada metrik *confusion matrix* yang mencakup akurasi, *precision*, *recall*, dan *F1-score* berdasarkan perbandingan label prediktif dan aktual [10]. Guna mengoptimalkan hasil, teknik SMOTE digunakan untuk menyeimbangkan data, sehingga perubahan performa model dapat dianalisis secara mendalam. Hasil analisis tersebut kemudian dipresentasikan secara visual menggunakan *word cloud* untuk setiap kategori sentimen.

3. HASIL DAN PEMBAHASAN

3.1 Pengumpulan Data

Tahapan pengumpulan data dilakukan dengan menggunakan teknik *web scraping* dengan menghasilkan 42.672 ulasan pengguna Photomath dari Google Play Store yang ditunjukkan pada Gambar 2. Data yang ditampilkan terdiri dari tiga atribut utama, yaitu *content* (isi ulasan), *score* (rating), dan sentimen (label hasil klasifikasi awal).

```
print(ulasan_df[['content','score', 'sentimen']])

                                                content  score sentimen
0      atur foto harus kecil gitu soalnya panjang jdi...      1  negatif
1      kenapa ya waktu mau ngitung tuh ada tulisan "h...      4  positif
2      aplikasi yang sangat membantu dalam belajar ma...      5  positif
3      sekarang jadi berbayar, padahal dulu bisa liha...      2  negatif
4      Sebenarnya bagus, tapi untuk soal "nilai mutla...      5  positif
...                                                  ...    ...      ...
42667                                          👎👎👎      1  negatif
42668  Aplikaai ini???????????¿?????????????????????...      5  positif
42669                  ?????????????????? M . ........      5  positif
42670                                             None      5  positif
42671                                             None      4  positif

[42672 rows x 3 columns]
```

Gambar 2. Hasil web scraping untuk ulasan pengguna Photomath

3.2 Pelabelan Sentimen

Berdasarkan gambar 2, proses pelabelan sentimen dilakukan secara otomatis dengan mengacu pada nilai rating yang diberikan pengguna. Ulasan dengan skor 4 dan 5 dikategorikan sebagai sentimen positif, skor 3 sebagai netral, serta skor 1 dan 2 sebagai negatif. Hasil pelabelan ini terlihat pada kolom sentimen yang telah terisi dengan tiga kategori utama, yaitu positif, netral, dan negatif. Berdasarkan skor penilaian, ulasan dibagi menjadi tiga kategori sentimen, yaitu positif (36696 ulasan, 85,99%), netral (2001 ulasan, 4,69%), dan negatif (3975 ulasan, 9,32%). Distribusi ini menunjukkan adanya ketidakseimbangan kelas (*imbalanced dataset*), di mana ulasan positif mendominasi. Kondisi ini berpotensi memengaruhi kinerja model pada kelas minoritas.

Contoh pada data menunjukkan bahwa ulasan dengan isi yang bernada apresiatif seperti “aplikasi ini cukup bagus” diklasifikasikan sebagai sentimen positif, sedangkan ulasan yang mengandung keluhan seperti “tidak bisa dipakai…” dikategorikan sebagai negatif. Hal ini menunjukkan bahwa pendekatan berbasis rating cukup mampu merepresentasikan kecenderungan opini pengguna secara umum. Namun demikian, dari hasil yang ditampilkan juga terlihat adanya variasi bentuk teks, seperti penggunaan bahasa informal, singkatan, serta karakter berulang (misalnya tanda tanya berlebihan). Kondisi ini menunjukkan bahwa data ulasan masih

P-ISSN 2085-4315
E-ISSN 2502-8332

mengandung noise, sehingga memerlukan tahap prapemrosesan lebih lanjut agar kualitas data meningkat sebelum dilakukan proses klasifikasi.

3.3 Hasil Prapemrosesan

Tahapan Prapemrosesan berhasil menghilangkan elemen non-informatif seperti URL, angka, tanda baca, dan emotikon, serta mengubah kata ke bentuk dasar. Semua tahapan ini bertujuan untuk meningkatkan kualitas fitur teks dan stabilitas performa model klasifikasi[11], [12]. Hasilnya, teks ulasan menjadi lebih seragam dan siap dikonversi ke representasi numerik.

a. Pembersihan teks

Tahapan pembersihan teks merupakan proses yang harus dilakukan dalam prapemrosesan data dengan menghilangkan tanda baca, angka, simbol, dan spasi ganda yang tidak relevan dengan penelitian [2]. Contoh proses pembersihan teks pada Tabel 1 menunjukkan kalimat panjang yang mengandung emotikon (👍❤️) menjadi versi yang bersih dan hanya berisi kata-kata relevan.

Tabel 1. Proses Pembersihan Teks

Sebelum pembersihan kata	Setelah pembersihan kata
aplikasi ini cukup bagus, untungya aku nemuin nih aplikasi tepat waktu, kalo enggak, aku gak bisa nyelesain PR tepat waktu 👍❤️ Tapi inget! Kalo make aplikasi ini ngambil fotonya harus bener! Aku aja sempet salah ambil foto beberapa kali, tapi untungnya berhasil ke foto soal yang aku maksud, tetap semangat klean semua yang mengerjakan matematika ✊❤️	aplikasi ini cukup bagus untungya aku nemuin nih aplikasi tepat waktu kalo enggak aku gak bisa nyelesain PR tepat waktu Tapi inget Kalo make aplikasi ini ngambil fotonya harus bener Aku aja sempet salah ambil foto beberapa kali tapi untungnya berhasil ke foto soal yang aku maksud tetap semangat klean semua yang mengerjakan matematika

b. Case folding

Setelah pembersihan teks, seluruh teks diubah menjadi huruf kecil. Proses *case folding* mencegah duplikasi fitur (mis. “Aplikasi” vs “aplikasi”) ketika representasi vektor dibangun, dan merupakan langkah dasar yang selalu direkomendasikan dalam pipeline NLP [13]. Tabel 2 menunjukkan hasil dari proses case folding pada salah satu ulasan pengguna Photomath.

Tabel 2. Proses Case Folding

Sebelum case folding	Setelah case folding
Aplikasi yang sangat bagus untuk membantu dalam pembelajaran matematika Tapi tolong tambahkan semua simbol matematika yang lengkap serta operasinya	aplikasi yang sangat bagus untuk membantu dalam pembelajaran matematika tapi tolong tambahkan semua simbol matematika yang lengkap serta operasinya

c. Tokenisasi

Pada tabel 3 menampilkan teks hasil case folding yang kemudian dipecah menjadi token (kata). Tokenisasi membuat teks yang tadinya berupa kalimat menjadi daftar kata yang mudah dihitung frekuensi dan bobotnya (TF-IDF). Tokenisasi untuk Bahasa Indonesia harus menangani tanda hubung, singkatan, dan angka yang sudah dibersihkan sebelumnya agar unit token representatif terhadap makna bahasa [14].

Tabel 3. Proses Tokenisasi

Sebelum tokenisasi	Setelah tokenisasi
aplikasi yang sangat bagus untuk membantu dalam pembelajaran matematika tapi tolong tambahkan semua simbol matematika yang lengkap serta operasinya	[aplikasi, yang, sangat, bagus, untuk, membantu, dalam, pembelajaran, matematika, tapi, tolong, tambahkan, semua, simbol, matematika, yang, lengkap, serta, operasinya]

d. Normalisasi kata

P-ISSN 2085-4315
E-ISSN 2502-8332

Tahapan normalisasi kata adalah proses teks yang dilakukan untuk memperbaiki kata-kata yang salah eja atau kata yang disingkat [15], serta mengganti variasi kata tidak baku menjadi bentuk baku. Misalkan "tapi" → "tetapi" pada contoh yang ditampilkan pada tabel 4.

Tabel 4. Proses Normalisasi Kata

Sebelum normalisasi kata	Setelah normalisasi kata
[aplikasi, yang, sangat, bagus, untuk, membantu, dalam, pembelajaran, matematika, tapi, tolong, tambahkan, semua, simbol, matematika, yang, lengkap, serta, operasinya]	[aplikasi, yang, sangat, bagus, untuk, membantu, dalam, pembelajaran, matematika, tetapi, tolong, tambahkan, semua, simbol, matematika, yang, lengkap, serta, operasinya]

e. Stopword removal

Kata-kata fungsi (*stopwords*) yang tidak menyumbang banyak informasi semantik dihapus (mis. "yang", "untuk", "dalam"). Penggunaan daftar stopword berbahasa Indonesia—seperti yang tersedia di pustaka Sastrawi atau daftar stopword yang disesuaikan—membantu memperkecil dimensi fitur dan konsentrasi pada kata bermuatan informasi untuk sentimen[16]. Hasil dari proses stopword dapat dilihat pada tabel 5.

Tabel 5. Proses Stopword Removal

Sebelum stopword removal	Setelah stopword removal
[aplikasi, yang, sangat, bagus, untuk, membantu, dalam, pembelajaran, matematika, tetapi, tolong, tambahkan, semua, simbol, matematika, yang, lengkap, serta, operasinya]	[aplikasi, sangat, bagus, membantu, pembelajaran, matematika, tambahkan, semua, simbol, matematika, lengkap, operasinya]

f. Stemming

Terakhir, kata-kata diubah ke bentuk dasar menggunakan Sastrawi Stemmer (khusus Bahasa Indonesia), sehingga variasi bentukan kata (mis. "membantu", "bantuan") dijadikan satu akar ("bantu"). Hasil dari proses stemming disajikan pada tabel 6. Stemming mengurangi sparsity dan meningkatkan kemampuan model untuk menangkap pola akar kata dalam TF-IDF atau representasi lainnya[17].

Tabel 6. Proses Stemming

Sebelum stemming	Setelah stemming
[aplikasi, sangat, bagus, membantu, pembelajaran, matematika, tambahkan, semua, simbol, matematika, lengkap, operasinya]	[aplikasi, sangat, bagus, bantu, ajar, matematika, tambah, semua, simbol, matematika, lengkap, operasi]

3.4 Ekstraksi Fitur

Tahapan ekstraksi fitur dilakukan setelah prapemrosesan dengan menerapkan TF-IDF untuk menghitung bobot kata berdasarkan frekuensi kemunculannya pada seluruh dokumen. Hasil dari ekstraksi ini berupa matriks fitur yang akan digunakan dalam tahap pelatihan model.

3.5 Model KNN

Tahapan klasifikasi sentimen diawali dengan mendistribusikan data ke dalam sekumpulan data latih dan data uji. Implementasi klasifikasi ini menggunakan pendekatan *K-Nearest Neighbor* (KNN) dengan perbandingan data uji sebesar 20% dan data latih sebesar 80% [19]. Pengujian dengan variasi nilai K dilakukan untuk mengevaluasi stabilitas performa model. Seluruh data hasil pengujian tersebut telah dirangkum pada Tabel 7.

Tabel 7. Hasil KNN dengan beberapa nilai K

K	Nilai Akurasi
3	0.8527
5	0.8605
7	0.8654
9	0.8661

JURNAL ILMIAH FIFO
Vol. 18 No. 1 (2026) 41-51
DOI: 10.22441/fifo.2026.v18i1.005
P-ISSN 2085-4315
E-ISSN 2502-8332

Dari tabel 7 diperoleh hasil pengujian dengan variasi nilai K (K=3, 5, 7, dan 9) menunjukkan bahwa kinerja model KNN relatif stabil dengan rentang akurasi antara 0.8527 hingga 0.8661. Model KNN dengan k=9 memberikan hasil akurasi terbaik sebesar 0.8661, diikuti oleh K = 7 dan K = 5 dengan nilai akurasi masing-masing yaitu 0,8654 dan 0,8605. Sedangkan model KNN dengan nilai terendah pada k = 3 memperoleh 0.8527. Perbedaan akurasi antar nilai K tergolong sangat kecil (kurang dari 0,2%). Hal ini menunjukkan bahwa model KNN pada penelitian ini dilakukan pada nilai K yang relatif besar (K = 5, 7, dan 9), di mana nilai K yang terlalu kecil berpotensi membuat model lebih sensitif terhadap noise. Sementara, penentuan nilai K yang terlalu besar dapat menyebabkan model terlalu menggeneralisasi data, karena keputusan klasifikasi dipengaruhi oleh terlalu banyak tetangga yang mungkin tidak relevan dengan kelas sebenarnya.[20]

3.6 Evaluasi Model KNN

Model KNN dengan parameter k = 9 menghasilkan akurasi keseluruhan sebesar 86,61%. Nilai evaluasi model per kategori sentimen ditunjukkan pada Tabel 8.

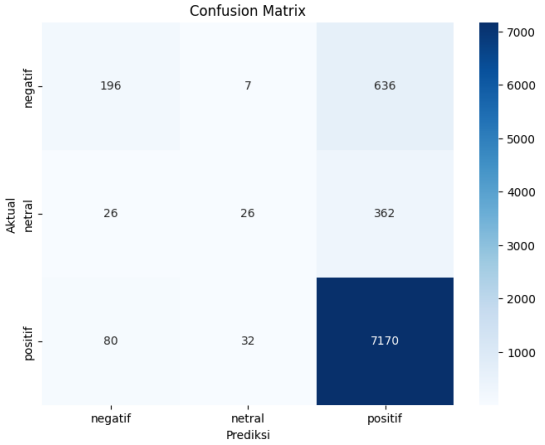
.Tabel 8. Nilai Precision, Recall, dan F1-Score Model KNN

Sentimen	*Precision*	*Recall*	*F1-Score*	*Support*
Negatif	0.65	0.23	0.34	839
Netral	0.40	0.06	0.11	414
Positif	0.88	0.98	0.93	7282

Tabel 8 menyajikan hasil evaluasi model KNN menggunakan metrik *precision*, *recall*, dan *F1-score* pada kasus klasifikasi multikelas dengan distribusi data yang tidak seimbang, karena akurasi saja tidak cukup merepresentasikan kinerja model secara menyeluruh. Hasil evaluasi juga diperkuat melalui visualisasi *confusion matrix*. Model menunjukkan performa sangat baik pada kelas positif, dengan nilai *recall* sebesar 0,98 yang menandakan hampir seluruh data positif berhasil diklasifikasikan dengan benar. Hal ini terlihat dari *confusion matrix*, di mana sebanyak 7170 data positif diprediksi secara tepat.

Sebaliknya, kinerja model pada kelas negatif masih rendah, ditunjukkan oleh nilai *recall* sebesar 0,23 yang mengindikasikan hanya sebagian kecil data negatif yang berhasil dikenali. *Confusion matrix* menunjukkan bahwa 636 data negatif justru diprediksi sebagai positif. Kondisi ini mengindikasikan adanya kemiripan pola kata antara kelas negatif dan positif. Temuan ini sejalan dengan penelitian sebelumnya yang menyatakan bahwa KNN cenderung bias terhadap kelas mayoritas pada dataset tidak seimbang [21].

Kelas netral memiliki performa terendah. Dari 414 data netral, hanya 26 yang berhasil diprediksi dengan benar. Sebagian besar 362 justru diklasifikasikan sebagai positif. Hal ini menunjukkan bahwa karakteristik teks netral seringkali memiliki kosakata yang mirip dengan ulasan positif, namun dengan intensitas ekspresi yang lebih lemah. Dalam representasi TF-IDF, hanya berbasis frekuensi kata sehingga kurang mampu menangkap hubungan semantik antar kata, yang dapat menyebabkan kesulitan membedakan konteks sentimen[22]. Untuk memahami pola kesalahan klasifikasi yang terjadi pada model, hasil prediksi divisualisasikan menggunakan confusion matrix seperti yang ditunjukkan pada Gambar 3.



Gambar 3. Confusion Matrix Pada Rasio 80:20 dengan Nilai K=9

P-ISSN 2085-4315
E-ISSN 2502-8332

Berdasarkan hasil analisis, model terbukti unggul dalam mendeteksi sentimen positif, namun kurang optimal dalam mengklasifikasikan kategori netral dan negatif. Kendala utama muncul dari ketidakseimbangan distribusi data serta tumpang tindih kosakata antarlabel. Meskipun KNN efektif dalam mengenali pola kata positif, rendahnya F1-score pada kategori lainnya menunjukkan kerentanan model terhadap data yang tidak seimbang, sebagaimana dikuatkan oleh penelitian [8].

Sebagai solusi, penelitian ini menerapkan SMOTE untuk menyeimbangkan kelas minoritas dalam dataset melalui pembuatan sampel sintetis [23]. Langkah ini bertujuan untuk meminimalisasi kesenjangan frekuensi antar kelas sebelum dilakukan pengujian ulang. Selanjutnya, perbandingan akurasi antara model sebelum dan sesudah penyeimbangan disajikan pada Tabel 9.

Tabel 9. Perbandingan Kinerja Model KNN tanpa SMOTE dan dengan SMOTE

Metode	Akurasi	Precision Macro	Recall Macro	F1-Score Macro
KNN (Tanpa SMOTE)	0.866081	0.642274	0.427011	0.460091
KNN (Dengan SMOTE)	0.570006	0.434224	0.540080	0.407530

Berdasarkan Tabel 9, model KNN tanpa SMOTE menghasilkan akurasi lebih tinggi yaitu 0,866081, dibandingkan dengan KNN dengan SMOTE yang hanya sebesar 0,570006. Hal ini menunjukkan bahwa tanpa penyeimbang data, model cenderung akurat secara keseluruhan karena lebih banyak memprediksi kelas mayoritas, yaitu kelas positif. Namun, jika dilihat dari nilai recall macro, KNN dengan SMOTE menunjukkan nilai yang lebih tinggi yaitu 0.540080, dibandingkan dengan tanpa SMOTE sebesar 0,427011. Peningkatan ini menunjukkan bahwa setelah dilakukan penyeimbangan data, model menjadi lebih mampu mengenali data pada kelas minoritas, yaitu sentimen netral dan negatif. Temuan ini sejalan dengan penelitian terbaru yang menyatakan bahwa teknik oversampling seperti SMOTE dapat meningkatkan kemampuan model dalam mendeteksi kelas minoritas pada dataset yang tidak seimbang, meskipun terkadang diikuti penurunan akurasi keseluruhan.[24], [25]. Di sisi lain, nilai precision macro dan F1-score macro pada KNN dengan SMOTE justru mengalami penurunan dibandingkan tanpa SMOTE. Precision macro turun dari 0,642274 menjadi 0,434224, dan F1-score macro turun dari 0,460091 menjadi 0,407530. Hal ini menunjukkan bahwa meskipun model lebih mampu menemukan data minoritas, tingkat ketepatan prediksi menjadi lebih rendah, sehingga terjadi peningkatan kesalahan klasifikasi. Hal ini menunjukkan adanya trade-off antara akurasi dan keseimbangan performa model pada dataset tidak seimbang.

Untuk memberikan gambaran secara keseluruhan terhadap kinerja model, dilakukan perbandingan dengan metode klasifikasi lain yang umum digunakan dalam analisis sentimen. Pemilihan kedua metode tersebut didasarkan pada karakteristiknya yang berbeda dalam menangani data teks, dimana Naive Bayes dikenal efisien dan sederhana, sedangkan SVM memiliki kemampuan yang baik dalam menangani dimensi fitur yang tinggi dari representasi teks [26]. Evaluasi perbandingan kinerja model dilakukan menggunakan metrik akurasi, *precision*, *recall*, dan *F1-score*, dengan fokus pada nilai *macro* untuk menilai performa model secara keseluruhan pada setiap kelas. Hasil evaluasi tersebut disajikan pada Tabel 10.

Tabel 10. Perbandingan Kinerja Model KNN, Naive Bayes dan SVM

Metode	Akurasi	Precision Macro	Recall Macro	F1-Score Macro
KNN	0.866081	0.642274	0.427011	0.460091
Naïve Bayes	0.876743	0.681938	0.434043	0.459769
SVM	0.889397	0.800543	0.487371	0.513369

Berdasarkan Tabel 10, SVM menunjukkan kinerja terbaik dengan akurasi tertinggi sebesar 0,8894 serta nilai precision macro 0,8005, recall macro 0,4874, dan F1-score macro 0,5134. Hal ini menunjukkan bahwa SVM lebih efektif dalam mengklasifikasikan data secara seimbang pada setiap kelas. Naïve Bayes berada pada posisi kedua dengan akurasi 0,8767, namun nilai recall macro (0,4340) dan F1-score macro (0,4598) masih rendah, sehingga kemampuan dalam mengenali seluruh data, terutama kelas minoritas, belum optimal. Sementara itu, KNN memperoleh akurasi 0,8661 dengan nilai recall macro 0,4270 dan F1-score macro 0,4601 yang relatif mendekati Naïve Bayes, tetapi memiliki precision macro yang lebih rendah (0,6423). Secara keseluruhan, hasil ini menunjukkan bahwa SVM lebih unggul dalam menangani data teks berbasis TF-IDF,

P-ISSN 2085-4315
E-ISSN 2502-8332

khususnya pada kondisi data tidak seimbang, sejalan dengan penelitian sebelumnya [27]. Di sisi lain, KNN dan Naïve Bayes masih memiliki keterbatasan dalam mendeteksi kelas minoritas.

3.7 Visualisasi Hasil Sentimen

Visualisasi *word cloud* digunakan untuk menggambarkan distribusi kata dominan pada setiap kategori sentimen. Gambar 4 menyajikan kata-kata yang dominan pada ulasan dengan sentimen positif. Kata-kata seperti "bagus", "bantu", "mantap", dan "sangat" tampak dominan dalam word cloud. Hal ini menunjukkan bahwa sebagian besar pengguna merasa aplikasi Photomath membantu dalam memahami penyelesaian soal matematika. Selain itu, munculnya kata "terima kasih" dan "banget" mengindikasikan adanya apresiasi pengguna terhadap fungsi aplikasi yang dianggap mempermudah proses belajar matematika. Kata-kata seperti "matematika", "soal", dan "ajar" menegaskan bahwa aplikasi ini memang digunakan sebagai media pembelajaran.

Gambar 4. *Word Cloud* Sentimen Positif

Pada Gambar 5 terlihat kata-kata seperti "lumayan", "coba", dan "kadang" yang muncul cukup sering dalam ulasan dengan sentimen netral. Kata-kata tersebut menunjukkan bahwa sebagian pengguna memberikan tanggapan yang bersifat deskriptif atau evaluatif tanpa menunjukkan penilaian yang terlalu kuat. Ulasan pada kategori ini umumnya berisi pengalaman penggunaan aplikasi atau saran perbaikan, sehingga tidak secara jelas menunjukkan kecenderungan sentimen positif maupun negatif.

Gambar 5. *Word Cloud* Sentimen Netral

Gambar 6 menampilkan kata-kata dominan pada ulasan dengan sentimen negatif. Beberapa kata yang terlihat menonjol antara lain "tidak bisa", "jelek", dan "susah". Kemunculan kata-kata tersebut mencerminkan

P-ISSN 2085-4315
E-ISSN 2502-8332

adanya keluhan pengguna terkait performa aplikasi, seperti kesalahan dalam membaca soal atau keterbatasan fitur tertentu. Visualisasi ini memberikan gambaran mengenai aspek yang masih menjadi perhatian pengguna dan dapat menjadi masukan bagi pengembang untuk meningkatkan kualitas aplikasi.

Gambar 6. *Word Cloud* Negatif

4. KESIMPULAN

Penelitian ini berhasil menerapkan metode K-Nearest Neighbor (KNN) untuk mengklasifikasikan sentimen 42.672 ulasan pengguna Photomath. Model dengan parameter k = 9 menghasilkan akurasi sebesar 86,61%, dengan performa sangat baik pada sentimen positif (F1-score 0,93), namun masih rendah pada sentimen netral dan negatif akibat ketidakseimbangan data. Penerapan SMOTE mampu meningkatkan kemampuan model dalam mengenali kelas minoritas, yang ditunjukkan oleh peningkatan recall, meskipun akurasi keseluruhan menurun. Hal ini menunjukkan adanya trade-off antara akurasi dan keseimbangan performa model. Hasil perbandingan menunjukkan bahwa metode SVM memberikan performa terbaik dibandingkan KNN dan Naïve Bayes, baik dari sisi akurasi maupun F1-score macro. Dengan demikian, pemilihan metode yang tepat serta penanganan data tidak seimbang menjadi faktor penting dalam meningkatkan kinerja klasifikasi sentimen.

DAFTAR PUSTAKA

[1] G. P. Store, "Photomath – Aplikasi Pembelajaran Matematika." Accessed: Mar. 30, 2026. [Online]. Available: https://play.google.com/

[2] D. A. Varissa Azis and Y. Sibaroni, "Optimizing Word2Vec Dimensions for Sentiment Analysis of Photomath Reviews using Random Forest and SVM," *Build. Informatics, Technol. Sci.*, vol. 6, no. 4, pp. 2248–2260, 2025, doi: 10.47065/bits.v6i4.6616.

[3] A. Adiansyah and Wahyudin, "Analisis Sentimen Pada Ulasan Aplikasi Home Credit dengan Metode SVM dan KNN," *J. Komput. Antart.*, vol. 1, no. 4, pp. 174–181, 2023, doi: 10.70052/jka.v1i4.50.

[4] M. N. Muttaqin and I. Kharisudin, "Analisis Sentimen Aplikasi Gojek Menggunakan Support Vector Machine dan K Nearest Neighbor," *UNNES J. Math.*, vol. 10, no. 2, pp. 22–27, 2021, doi: 10.15294/ujm.v10i2.48474.

[5] W. S. J. Rahanra and Kusrini, "Analisa Penggunaan Aplikasi Checkmath Menggunakan Algoritma K-Nearest Neighbor," *Djtechno J. Teknol. Inf.*, vol. 6, no. 1, pp. 114–124, 2025, doi: 10.46576/djtechno.

[6] M. Saifurridho, Martanto, and U. Hayati, "Analisis Algoritma K-Nearest Neighbor Terhadap Sentimen Pengguna Aplikasi Shopee," *J. Inform. Terpadu*, vol. 10, no. 1, pp. 21–26, 2024, doi: 10.54914/jit.v10i1.1054.

[7] I. W. D. P. Dewi and I. G. A. Handayani, "Peranan Aplikasi Photomath dalam Pembelajaran Matematika di Era Literasi Digital (Kajian Pustaka)," *Suluh Pendidik.*, vol. 20, no. 1, pp. 94–101, 2022, doi: 10.46444/suluh-pendidikan.v20i1.411.

[8] A. Saepudin, R. Aryanti, E. Fitriani, and Dahlia, "Optimasi Algoritma SVM Dan k-NN Berbasis Particle Swarm Optimization Pada Analisis Sentimen Fenomena Tagar #2019GantiPresiden," *J. Tek. Komput.*

P-ISSN 2085-4315
E-ISSN 2502-8332

AMIK BSI, vol. VI, no. 1, pp. 95–102, 2020, doi: 10.31294/jtk.v4i2.

[9] F. Sodik, B. Dwi, and I. Kharisudin, "Perbandingan Metode Klasifikasi Supervised Learning pada Data Bank Customers Menggunakan Python," in *PRISMA, Prosiding Seminar Nasional Matematika*, 2020, pp. 689–694. [Online]. Available: https://journal.unnes.ac.id/sju/index.php/prisma/ ISSN 2613-9189%0A

[10] D. Musfiroh, U. Khaira, P. Eko, P. Utomo, and T. Suratno, "Sentiment Analysis of Online Lectures in Indonesia from Twitter Dataset Using InSet Lexicon," *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 1, no. April, pp. 24–33, 2021, doi: 10.57152/malcom.v1i1.20.

[11] A. F. Aufar, M. A. Rosid, A. Eviyanti, and I. R. I. Astutik, "Optimizing Text Preprocessing fors Accurate Sentiment Analysis on E-Wallet Reviews," *J. Inf. Comput. Technol. Educ.*, vol. 7, no. 2, pp. 42–50, 2023, doi: 10.21070/jicte.v7i2.1650.

[12] N. A. C.A, R. Novita, Mustakima, and N. E. Rozanda, "The Implementation of TF-IDF and Word2Vec on Booster Vaccine Sentiment Analysis Using Support Vector Machine Algorithm," *Procedia Comput. Sci.*, vol. 234, pp. 156–163, 2024, doi: 10.1016/j.procs.2024.02.162.

[13] I. T. Julianto, D. Kurniadi, and B. B. B. Jr, "Enhancing Sentiment Analysis With Chatbots: A Comparative Study of Text Pre-Processing," *J. Tek. Inform.*, vol. 4, no. 6, pp. 1419–1430, 2023, doi: 10.52436/1.jutif.2023.4.6.1448.

[14] V. Rahmaliyadi and M. M. Maridjan, "Sentiment Analysis of Indonesian-Language Plantix Application Reviews for Plant Disease Diagnosis Using Naive Bayes Methods," *J. Intell. Syst. Technol. Informatics*, vol. 1, no. 2, pp. 62–66, 2025, doi: 10.64878/jistics.v1i2.12.

[15] J. J. A. Limbong, I. Sembiring, and K. D. Hartomo, "Analisis Klasifikasi Sentimen Ulasan Pada E-Commerce Shopee Berbasis Word Cloud Dengan Metode Naive Bayes dan K-Nearest," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 9, no. 2, pp. 347–356, 2022, doi: 10.25126/jtiik.202294960.

[16] A. A. Solihin *et al.*, "Evaluasi Pengaruh Varian Daftar Stopword terhadap Kinerja Klasifikasi Teks Al-Qur ' an dengan Support Vector Machine dan Backpropagation Neural Network," *J. Pendidik. dan Teknol. Indones.*, vol. 5, no. 7, pp. 1867–1880, 2025, doi: 10.52436/1.jpti.875.

[17] Y. Karuniawati, E. Utami, and A. Yaqin, "A Systematic Literature Review of Stemming in Non-Formal Indonesian Language," *Int. J. Innov. Sci. Res. Technol.*, vol. 8, no. 1, pp. 62–69, 2023, doi: 10.5281/zenodo.7547482.

[18] K. H. Manguri and R. N. Ramadhan, "Twitter Sentiment Analysis on Worldwide COVID-19 Outbreaks," *Kurdistan J. Appl. Res.*, vol. 5, no. 3, pp. 54–63, 2020, doi: 10.24017/covid.8.

[19] F. Amandasari and Damayanti, "Perbandingan Kinerja Support Vector Machine dan Naive Bayes dalam Klasifikasi Sentimen Twitter Terhadap Pelayanan BPJS," *J. Pendidik. dan Teknol. Indones.*, vol. 5, no. 3, pp. 645–653, 2025, doi: 10.52436/1.jpti.680.

[20] M. Rizki, A. Hermawan, and D. Avianto, "Optimization of Hyperparameter K in K-Nearest Neighbor Using Particle Swarm Optimization," *JUITA J. Inform.*, vol. 12, no. 1, pp. 71–79, 2024, doi: 10.30595/juita.v12i1.20688.

[21] A. A. Amer, S. D. Ravana, R. Ahamed, and A. Habeeb, "Effective k‐nearest neighbor models for data classification enhancement," *J. Big Data*, 2025, doi: 10.1186/s40537-025-01137-2.

[22] T. F. Abdillah, Hasmawati, and Bunyamin, "Comparison of TF-IDF and GloVe Word Embedding for Sentiment Analysis of 2024 Presidential Candidates," *Build. Informatics, Technol. Sci.*, vol. 6, no. 2, pp. 961–969, 2024, doi: 10.47065/bits.v6i2.5668.

[23] T. Abdillah, U. Khaira, and B. F. Hutabarat, "Komparasi Metode Naive Bayes dan K-Nearest Neighbors Terhadap Analisis Sentimen Pengguna Aplikasi Zenius," *Process. J. Ilm. Sist. Informasi, Teknol. Inf. dan Sist. Komput.*, vol. 19, no. 1, pp. 32–44, 2024, doi: 10.33998/processor.2024.19.1.1596.

[24] J. Pardede and D. P. Pamungkas, "The Impact of Balanced Data Techniques on Classification Model Performance," *Sci. J. Informatics*, vol. 11, no. 2, pp. 401–412, 2024, doi: 10.15294/sji.v11i2.3649.

[25] A. Fajar, A. Fauzi, and A. Faqih, "Optimization of the K-Nearest Neighbors (KNN) Algorithm in Imbalanced Dataset Classification Using the SMOTE Technique," *J. Artif. Intell. Eng. Appl.*, vol. 4, no. 2, 2025, doi: 10.59934/jaiea.v4i2.756.

[26] K. Novianto, H. Herlawati, and A. Hidayat, "Klasifikasi Sentimen iPhone Bekas di Tokopedia menggunakan Naïve Bayes dan Support Vector Machine," *J. Ilm. FIFO*, vol. 17, no. 2, pp. 212–224, 2025, doi: 10.22441/fifo.2025.v17i2.010.

[27] Z. Yang, "Recent Deep Learning Techniques on Short Text Classification Recent Deep Learning Techniques on Short Text Classification," in *EITCE '22: Proceedings of the 2022 6th International Conference on Electronic Information Technology and Computer Engineering*, 2026. doi: 10.1145/3573428.3573468.