

# Perbandingan Algoritma Klasifikasi Sentimen Pada Ulasan Aplikasi *Mobile JKN*

Ramadhoni Gibran Maulana<sup>1✉</sup>), Tegar Hardiansyah Prasetyo<sup>2)</sup>, Muhammad Shidiq Budiman<sup>3)</sup>, Muhammad Fadhlwan Karimuddin<sup>4)</sup>, Rizqy Adriansyah<sup>5)</sup>, Yayan Hendrian<sup>6)</sup>, Shynde Limar Kinanti<sup>7)</sup>

<sup>1-7)</sup> Informatika, Teknik Dan Informatika, Universitas Bina Sarana Informatika, Indonesia

<sup>1)</sup> ramdhon1.ssss@gmail.com ✉

## ABSTRACT

*Sentiment classification plays an important role in evaluating public response to digital services such as BPJS Kesehatan's Mobile JKN application. This study aims to compare the performance of three machine learning algorithms—Support Vector Machine (SVM), Naive Bayes, and K-Nearest Neighbor (KNN) for classifying user sentiment based on reviews in the Google Play Store. A total of 10,000 user reviews were collected using Python and processed on Google Colab. The research process includes text pre-processing, sentiment labeling based on ratings, data splitting, and model training. Evaluation was conducted using accuracy, precision, recall, F1 score, and confusion matrix metrics. The results show that the SVM algorithm provides the best accuracy of 90.9%, followed by Naive Bayes (90.3%) and KNN (86%). These findings prove that SVM is the most effective model for sentiment classification in the context of public services and provide important insights for government policy evaluation and digital service improvement.*

*Keywords:* sentiment analysis, mobile jkn, naive bayes, svm, knn, google play store

## ABSTRAK

Klasifikasi sentimen memiliki peran penting dalam mengevaluasi respons masyarakat terhadap layanan digital seperti aplikasi Mobile JKN milik BPJS Kesehatan. Penelitian ini bertujuan untuk membandingkan kinerja tiga algoritma *machine learning*—*Support Vector Machine* (SVM), *Naive Bayes*, dan *K-Nearest Neighbor* (KNN)—dalam mengklasifikasikan sentimen pengguna berdasarkan ulasan di Google Play Store. Data sebanyak 10.000 ulasan dikumpulkan menggunakan Python dan diproses di Google Colab. Proses penelitian meliputi pra-pemrosesan teks, pelabelan sentimen berdasarkan *rating*, pembagian *dataset*, serta pelatihan model. Evaluasi dilakukan menggunakan metrik akurasi, presisi, *recall*, skor F1, dan *confusion matrix*. Hasil menunjukkan bahwa algoritma SVM memberikan akurasi terbaik sebesar 90,9%, diikuti oleh Naive Bayes (90,3%) dan KNN (86%). Temuan ini membuktikan bahwa SVM merupakan model paling efektif untuk klasifikasi sentimen dalam konteks layanan publik, serta memberikan wawasan penting bagi evaluasi kebijakan pemerintah dan peningkatan layanan digital.

Kata kunci: analisis sentimen, mobile JKN, naive bayes, svm, knn, google play store

## I. PENDAHULUAN

Bidang Teknologi Informasi dan Komunikasi (TIK) terus berkembang pesat dengan cepat, bahkan di sektor jaminan kesehatan yang sekarang sangat bergantung pada penerapan TIK. Kehadiran aplikasi Mobile Program Jaminan Kesehatan Nasional (JKN) menandai transformasi digital BPJS Kesehatan, menggeser model bisnis dari proses administratif konvensional beralih dari layanan fisik di cabang atau fasilitas kesehatan menjadi solusi digital yang dapat diakses peserta secara fleksibel, kapan saja dan di mana saja. Teknologi layanan kesehatan berbasis mobile (*mobile health*) di Indonesia terlihat cukup pesat dengan semakin

banyaknya aplikasi kesehatan yang bermunculan. Namun, adopsi *mobile health* di Tanah Air masih terhambat karena rendahnya jumlah pengguna, termasuk tenaga medis yang memanfaatkan aplikasi tersebut [1].

Sudah banyak penelitian tentang analisis sentimen, beberapa diantaranya ialah penelitian mengenai analisis sentimen kinerja dewan perwakilan rakyat (DPR) pada twitter menggunakan metode *Naive Bayes Classifier*. Adapun hasil dari penelitian tersebut menunjukkan diperoleh bahwa sebanyak 95 twitter tergolong sentimen terindikasi positif tingkat kecenderungan 75%, 693 twitter bersifat netral dengan kecenderungan 79%, dan 758 twitter menunjukkan sentimen

terindikasi negatif kecenderungan sebesar 82%. Namun, kelemahannya terletak pada jumlah data yang terbatas, hanya menggunakan satu platform (X), tidak adanya kategori sentimen netral, serta kurangnya perbandingan dengan algoritma lain dan analisis mendalam terhadap isi opini. Secara keseluruhan, penelitian ini memberikan gambaran awal yang baik tentang penerapan analisis sentimen berbasis *machine learning* di bidang politik, meskipun masih perlu pengembangan pada cakupan data dan variasi metode agar hasilnya lebih representatif [2].

Selanjutnya penelitian mengenai analisis sentimen terhadap program kartu prakerja dengan menggunakan metode *Support Vector Machine* (SVM). Dari hasil pengujian, kernel linear menunjukkan performa lebih baik dengan tingkat akurasi mencapai 98,67%. Penyebaran kuesioner yang tidak merata dan jumlah responden yang sedikit membuat hasil penelitian kurang mewakili seluruh pengguna Mobile JKN. Pendekatan yang hanya menggunakan metode kuantitatif tanpa analisis kualitatif menyebabkan alasan kepuasan pengguna belum tergali secara mendalam. Selain itu, variabel yang diteliti masih terbatas sehingga belum mampu menjelaskan semua faktor yang memengaruhi kepuasan dan penggunaan berkelanjutan [3].

Penelitian mengenai analisis sentimen selanjutnya ialah, perkembangan Vtuber dengan metode SVM berbasis SMOTE [4]. Hasil penelitian menunjukkan bahwa perkembangan Vtuber terhadap masyarakat Indonesia bersifat positif, dengan nilai sentimen positif sebesar 89,09% menggunakan algoritma SVM. Penelitian berikutnya tentang analisis sentimen ialah, pengguna Instagram di Indonesia pada ulasan smartphone menggunakan *Naïve Bayes* [5]. Hasilnya menunjukkan akurasi mencapai 73%, sedangkan versi Multinomial *Naïve Bayes* memberikan hasil lebih tinggi dengan akurasi 81%. Penelitian yang dilakukan Larasati [6], yang berjudul analisis sentimen pada aplikasi dana dengan metode *random forest*, dalam parameter terbaik dengan data yang diseimbangkan menjadi 250 data per kelasnya nilai yang didapatkan dari rasio data latih 80% dan data uji 20% yaitu *precision* 84%, *recall* 84%, *F1-Score* 84%, dan *accuracy* 84%. Hal ini menunjukkan, sistem dapat mengidentifikasi data dengan kedalaman *tree* 65 dan jumlah *tree* 400 dengan baik.

Adapun penelitian selanjutnya tentang analisis sentimen masyarakat pada penggunaan *e-commerce* menggunakan algoritma KNN [7]. Model sentimen analysis yang dikembangkan menggunakan algoritma NLP dan KNN menunjukkan kinerja yang memuaskan. Akurasi model mencapai 82%, dan nilai dari *cross-validation score* sebesar 80%. Selanjutnya ada penelitian tentang komparasi algoritma *naive bayes* dan *logistic regression* untuk analisis sentimen [8]. Hasil dari komparasi tersebut mendapatkan nilai algoritma *Naïve Bayes* menghasilkan akurasi 90% dan algoritma *Logistic Regression* menghasilkan akurasi 91%.

Beberapa penelitian sebelumnya telah menerapkan algoritma *machine learning* seperti *Support Vector*

*Machine* (SVM), *Naïve Bayes*, dan *K-Nearest Neighbor* (KNN) untuk analisis sentimen pada berbagai konteks aplikasi digital. Namun, penelitian ini membandingkan performa ketiga algoritma tersebut menggunakan dataset ulasan aplikasi Mobile JKN yang masih terbatas. Kondisi ini menunjukkan adanya kesenjangan penelitian dalam memahami sejauh mana efektivitas masing-masing algoritma ketika diterapkan pada konteks layanan publik berbasis digital di bidang kesehatan. Berdasarkan hal tersebut, penelitian ini berupaya menyajikan perbandingan performa ketiga algoritma tersebut dengan menggunakan data ulasan pengguna aplikasi Mobile JKN dari Google Play Store. Sehingga penelitian ini berkontribusi dengan memberikan gambaran perbandingan efektivitas algoritma klasifikasi sentimen pada layanan publik digital di sektor kesehatan serta menjadi referensi bagi BPJS Kesehatan dalam mengevaluasi umpan balik pengguna dan meningkatkan kualitas layanan Mobile JKN. Diharapkan bahwa hasil penelitian ini mampu memberikan hasil performa pada masing-masing algoritma dan dapat memberikan kontribusi dalam bidang pengembangan teknologi pelayanan publik serta menjadi acuan bagi pengambil kebijakan dalam melakukan evaluasi dan pengembangan lebih lanjut terhadap sistem *e-health* di Indonesia.

## II. METODE PENELITIAN

### A. Machine Learning

*Machine Learning* (ML) merupakan satu dari banyak teknologi kecerdasan buatan berorientasi pada kemampuan mekanisme untuk mempelajari pola dari data. Fokus utama ML adalah mengembangkan sistem yang dapat mempelajari informasi terproses otomatis tanpa diberi instruksi diungkapkan secara langsung oleh manusia. Secara proses pembelajarannya, (ML) memerlukan data yang valid dan berkualitas sebagai bahan pelatihan (*training*), yang kemudian digunakan untuk menguji kemampuan model (*testing*) dalam menghasilkan *output* yang akurat dan optimal [9].

Pemilihan algoritma SVM, *Naïve Bayes*, dan KNN didasarkan pada perbedaan karakteristik dan efektivitasnya dalam klasifikasi teks. SVM memiliki kemampuan generalisasi yang baik, yaitu mampu mengklasifikasikan data baru yang tidak termasuk dalam data pelatihan karena menggunakan pendekatan *Structural Risk Minimization* (SRM) yang meminimalkan kesalahan pelatihan sekaligus kompleksitas model [10]. *Naïve Bayes* dipilih karena sederhana dan efisien, sedangkan KNN efektif mengelompokkan data berdasarkan kemiripan. Algoritma lain seperti *Random Forest* atau model *deep learning* tidak digunakan karena penelitian ini difokuskan pada perbandingan algoritma klasik yang umum digunakan dan tidak memerlukan sumber daya komputasi besar.

### B. Support Vector Machine

*Support Vector Machine* (SVM) bekerja dengan memanfaatkan pola linear yang terbentuk oleh fitur-fitur yang luas sebagai dasar pembelajarannya. Metode ini menggabungkan prinsip optimasi matematis

dengan konsep statistik dalam proses pelatihan modelnya [11].

#### C. Naive Bayes

*Naive Bayes* termasuk sekelompok model pengklasifikasi yang bekerja mengacu pada *Teorema Bayes* dengan praduga utama bahwa semua ciri khas atau variabel prediktor saling independen. Namun kenyataannya, asumsi ini seringkali tidak terpenuhi, khususnya ketika ada keterkaitan kompleks antar fitur dan jumlah data yang tersedia terbatas [12].

#### D. K-Nearest Neighbors

Metode klasifikasi *K-Nearest Neighbor* (K-NN) bekerja dengan mengidentifikasi sejumlah tetangga terdekat dari data yang tidak memiliki label, kemudian menetapkan label berdasarkan mayoritas kelas dari tetangga tersebut dalam data pelatihan. Teknik ini termasuk salah satunya pendekatan identifikasi pola cukup populer juga banyak dipakai dalam mekanisme kategorisasi berbagai jenis informasi. Semakin tinggi tingkat kemiripan antara suatu dokumen dengan dokumen lain, maka semakin besar kemungkinan keduanya tergolong dalam kategori yang sama. Sebaliknya, perbedaan yang besar menandakan kemungkinan klasifikasi yang berbeda. Untuk mengukur kesamaan tersebut, digunakan sejumlah metode seperti tingkat kemiripan, kovariansi, dan keeratan hubungan. Khususnya, tingkat kemiripan dipakai menilai kedekatan representasi vektor dari dokumen dan *query*—kian tinggi kemiripan vektor, semakin besar relevansinya [13].

Tahapan umum dalam proses klasifikasi menggunakan metode K-NN adalah sebagai berikut:

- 1) Memilih nilai K yang tepat, yaitu jumlah neighbor terdekat yang diperhitungkan;
- 2) Mengukur jarak antar-data yang ingin diklasifikasikan dengan seluruh bahan pelatihan;
- 3) Mencari hasil perhitungan selisih tersebut dari paling dekat.
- 4) Memilih k data dengan jarak terdekat;
- 5) Mengidentifikasi label dari data tetangga tersebut;
- 6) Menentukan label untuk data baru berdasarkan jumlah terbanyak dari kelas tetangga terdekat.

Secara umum, penghitungan kedekatan antar data dapat dilakukan dengan menggunakan metode jarak geometri, salah satunya melalui rumus tertentu untuk mengukur seberapa jauh atau dekat posisi masing-masing titik data.

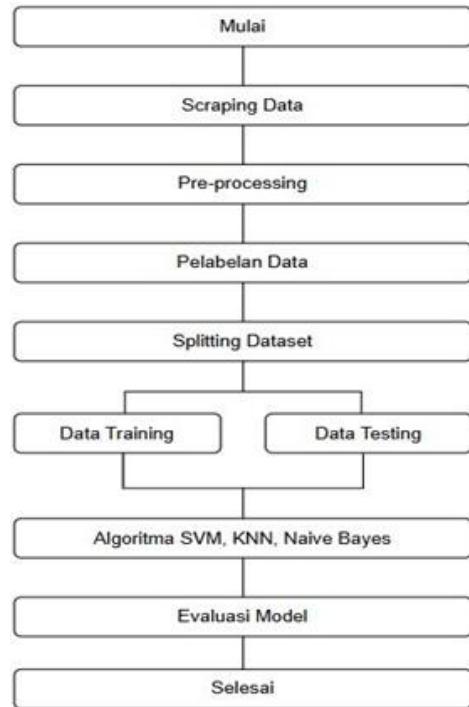
$$d = \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2 + \dots + (a_n - b_n)^2} \quad (1)$$

$$d = \sqrt{\sum_{i=1}^n (a_i - b_i)^2}$$

#### E. Kerangka Penelitian

Kerangka penelitian berfungsi untuk memberikan gambaran sistematis mengenai proses penelitian yang akan dilakukan. Dengan adanya kerangka penelitian

(Gambar 1), peneliti dapat memastikan bahwa setiap langkah penelitian memiliki keterkaitan yang logis.



Gambar 1. Kerangka Penelitian

### III. HASIL DAN PEMBAHASAN

#### A. Metode penelitian

Proses dimulai dengan langkah awal dalam merancang keseluruhan kegiatan. Setelah itu, dilakukan pengambilan data tersebut menggunakan teknik panen data dari Google Play Store guna memperoleh bahan mentah yang akan diolah. Data yang telah diperoleh kemudian melalui tahap *preprocessing*, seperti pembersihan data dari simbol, emoji dan angka, normalisasi kata yang tidak baku, dan penyesuaian format agar siap digunakan pada proses selanjutnya.

Tahap berikutnya adalah pelabelan data, yang dilakukan untuk memberikan informasi kategori pada masing-masing data sesuai kebutuhan, sehingga model dapat mempelajari pola dari data tersebut secara efektif. Setelah data siap, proses pelatihan model dilaksanakan memakai logika *Support Vector Machine*, *K-Nearest Neighbor*, dan *Naive Bayes*. Informasi ini dibagi menjadi dua yaitu, data latih dan data uji coba. Kemudian, performanya seperti akurasi dan presisi dinilai menggunakan *Confusion Matrix*. Hasil evaluasi ini dijadikan dasar untuk melakukan perbandingan terhadap beberapa model atau pendekatan guna menentukan metode terbaik. Setiap tahapan proses ini diakhiri dengan fase penyusunan hasil akhir, yang mencakup analisis menyeluruh dan kesimpulan dari penelitian yang telah dilakukan.

#### B. Sumber Data

Informasi yang dianalisis dalam penelitian ini didapat melalui proses *crawling* melalui platform

Google Play Store Android dengan bantuan skrip Python yang dijalankan di lingkungan Google Colab. Data ini diambil dengan mengakses dan mengekstraksi informasi dari ulasan pengguna aplikasi JKN Mobile. Proses *crawling* dilakukan secara otomatis untuk mengumpulkan data secara masif dan efisien, sehingga mampu menghimpun sebanyak 10.000 entri data ulasan pengguna. Hasil dari proses *crawling* ini disimpan dan dibagi ke dalam lima tabel, yang masing-masing merepresentasikan aspek-aspek berbeda yaitu data ulasan seperti review id, username, rating, review text, dan tanggal.

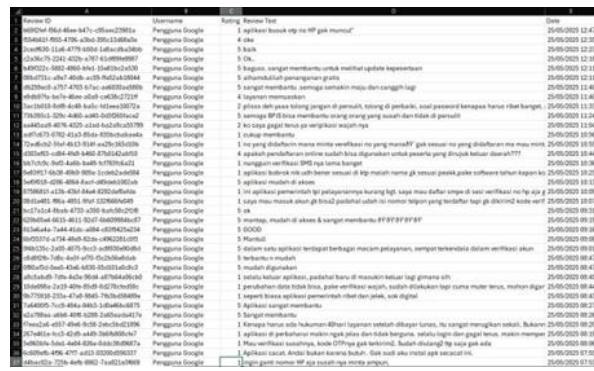
### C. Teknik Pengambilan Data

Metode *web crawling* dimanfaatkan untuk mendapatkan data studi, Sebagaimana dikatakan program atau kode otomatis yang disebut *Web Crawler* memiliki kemampuan untuk mengunjungi *website* sekaligus menggali data relevan ide pokok tertentu [14]. Hal ini, proses *crawling* dilakukan terhadap ulasan pengguna aplikasi di JKN Mobile, dengan memanfaatkan bahasa pemrograman Python serta pustaka pendukung library seperti google-play-scrapers, pandas, dan numpy, scikit-learn, nltk, matplotlib, dan seaborn. Seluruh proses dijalankan pada platform Google Colab untuk mempermudah eksekusi dan pengolahan data dalam skala besar.

#### D. Mengumpulkan Data

Platform Google Colaboratory, atau Google Colab, adalah lingkungan kerja berbasis *cloud* yang mendukung bahasa pemrograman Python untuk pengumpulan data dalam penelitian ini. Dengan memanfaatkan library python google-play-scraper yang memungkinkan mengambil data *review* sebuah aplikasi dari Google Play Store secara otomatis.

Informasi diperoleh cara menginisialisasi *script crawling* yang ditulis dalam Google Colab. Script tersebut mengakses halaman aplikasi JKN Mobile pada Google Play Store dan secara otomatis mengambil informasi ‘userId, userName, userImage, content, score, thumbsUpCount, reviewCreatedVerison, date, replyContent, repliedAt, appVersion’ dengan 10.000 informasi data pengguna. Hasil *scraping* data ditunjukkan pada Gambar 2.



Gambar 2. Hasil Scraping Data

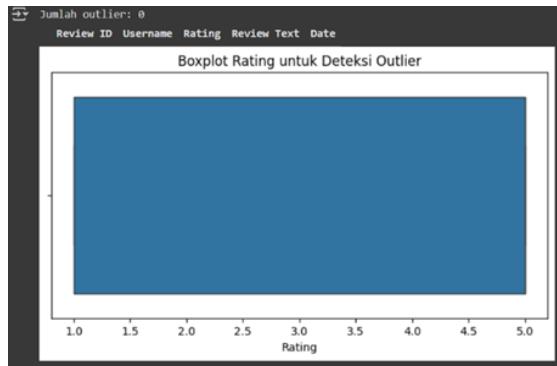
Pengambilan data ulasan pengguna terhadap aplikasi JKN Mobile dilakukan dengan menggunakan

library `google-play-scraper` memungkinkan untuk melakukan ekstraksi data dari Google Play Store secara otomatis dan terstruktur.

### E. *Outliers*

*Outliers* adalah nilai-nilai data yang jauh berbeda dari sebagian besar data dalam kumpulan. Teknik untuk mendeteksi *outliers* antara lain adalah visualisasi data, seperti *box plots*, atau menggunakan statistik, seperti skor Z [15].

Metode yang digunakan *Interquartile Range* (IQR), yaitu pendekatan indikator statistik untuk mengetahui penyebaran bahan Q1 dan Q3. IQR dihitung dengan mengurangkan Q1 dari Q3, dan nilai outlier didefinisikan sebagai data yang berada di luar batas bawah ( $Q1 - 1.5 * IQR$ ) dan batas atas ( $Q3 + 1.5 * IQR$ ).



Gambar 3. Hasil *Outlier*

Dalam pengecekan outlier perhitungan IQR dari kolom *Rating*, nilai batas bawah dan atas diidentifikasi, dan kemudian data yang berada di luar rentang tersebut dipilih. Hasilnya menunjukkan bahwa tidak ada *outlier* pada kolom *Rating*. Hal ini menunjukkan bahwa seluruh data rating yang diperoleh dari ulasan pengguna aplikasi JKN *Mobile* berada dalam rentang yang wajar dan valid, yaitu antara nilai 1 hingga 5, dan ditampilkan dengan visualisasi *boxplot*.

#### F. Missing Value

```
Jumlah missing value per kolom:  
Review ID      0  
Username       0  
Rating         0  
Review Text    0  
Date           0  
dtype: int64
```

Gambar 4. Hasil *Missing Value*

Untuk memastikan kualitas data sebelum dilakukan proses *pre-processing* dan analisis, langkah awal yang dilakukan adalah pemeriksaan nilai yang kosong atau *missing value* pada dataset dari hasil *scraping* ulasan aplikasi JKN Mobile. Dataset dibaca menggunakan library pandas dari file *hasil\_scraping\_ulasan\_app\_mobile\_jkn.csv*, kemudian dilakukan pengecekan jumlah nilai kosong di setiap kolom dengan fungsi *isnull().sum()*. Hasil dari

pemeriksaan ini menunjukkan berapa banyak entri data yang tidak terisi pada setiap kolom dalam *dataset*.

### G. Pre-Processing Data

#### 1) Convert File Ke CSV

Data yang sudah melalui proses *crawling* akan di convert ke format Comma-Separated Values (CSV) dengan *library* csv dan data akan melalui proses *sorting* untuk mengambil beberapa *field* yaitu ‘reviewId, userName, score, content, dan tanggal’.

#### 2) Cleaning Data

Pada proses *cleaning* data, *dataset* akan melalui beberapa proses seperti menghapus data duplikat, menghapus emoji, menghapus simbol, menghapus tanda baca, dan menghapus angka.

#### 3) Case Folding

Dengan menggunakan perintah *text.lower()* pada python, akan merubah semua *text* pada *dataset* menjadi huruf kecil agar seragam dan konsisten dalam analisis.

#### 4) Normalisasi Kata

Mesin dapat secara konsisten memproses kata-kata yang bermakna sama tetapi ditulis dengan cara yang berbeda dengan menggunakan *library* pandas merubah kata-kata tidak formal dan singkatan bentuk standar sesuai.

#### 5) Tokenize

Disini menggunakan perintah *python text.split()* dapat memecah sebuah *text* panjang menjadi bagian-bagian per kata atau frasa.

#### 6) Stopword Removal

Dengan mengunduh daftar *stopword* bahasa indonesia pada server *Natural Language Toolkit* (NLTK) yang merupakan *library* dari *Natural Language Processing* (NLP), *stopword* kemudian digunakan untuk menyaring kata-kata umum yang tidak mengandung informasi yang signifikan dalam teks, guna meningkatkan kualitas analisis data teks.

#### 7) Stemming Data

Dengan menggunakan *library* Sastrawi yang dapat mengubah kata-kata yang mengalami imbuhan dalam bahasa indonesia. Selain itu, *library* PorterStemmer dan SnowballStemmer juga untuk yang berbahasa inggris.

### H. Pelabelan Data

Pada proses ini akan menambahkan tabel sentimen dari *dataset* ‘*hasil\_preprocessing\_data.csv*’ dengan menggunakan perintah ‘*data['Sentimen'] = data['Peringkat'].apply(lambda x: 'Negatif' if x <= 3 else 'Positif')*’ *rating* dengan hasil 5, dan 4 akan di klasifikasikan sentimen positif, jika *rating* kurang dari 4 maka akan di klasifikasikan negatif.

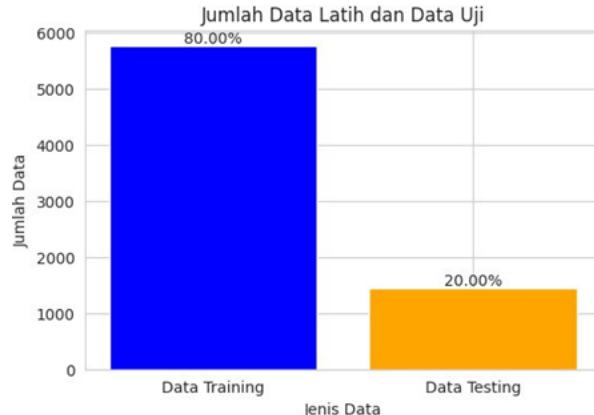
A	B	C
	Rating	Sentiment
1 stemming_data	1 Negatif	
2 aplikasi buku otp no hp muncul	4 Positif	
3 oke	5 Positif	
4 oh	5 Positif	
5 bagus banget update serta	5 Positif	
6 alhamdulillah tangan gratis	5 Positif	
7 bantu moja maju canggih	4 Positif	
8 layan muas	2 Negatif	
9 ptissa otaku tolong susit tolong baik password ribet banget bikin password pakai sp	5 Positif	
10 mohon bantuan orang orang susah susit	2 Negatif	
11 gagal ya verifikasi wajah ya	1 Negatif	
12 bantu	4 Positif	
13 no daftarin verifikasi no managak sesuai no daftar verifikasi	1 Negatif	
14 cari online serta rujuk daerah	1 Negatif	
15 tunggu verifikasi sms ya banget	1 Negatif	
16 aplikasi bobrok nik sesuai ktp nama sesuai peakppake software bobrok kwkwkwk	5 Positif	
17 aplikasi mudah akses	1 Negatif	
18 aplikasi perintah tayang banget daftar sesi verifikasi no hp pesan otp masuk tolong	1 Negatif	
19 matuk akun isi nomor telpon daftar kirim kode verifikasi ya chat wa balas pakai rc	1 Negatif	
20 oh	5 Positif	
21 mantap mudah akses bantu	5 Positif	
22 ganteng	5 Positif	
23 mantul	5 Positif	
24 aplikasi layan kendala verifikasi akun	5 Positif	
25 bantu n mudah	5 Positif	
26 mudah	5 Positif	
27 aplikasi pascek sifit	1 Negatif	
28 salah data pakai verifikasi wajah muter mohon ganti tim ya	1 Negatif	
29 aplikasi perintah ribet jelak sok digital	3 Negatif	
30 aplikasi bantu	5 Positif	
31 bantu	5 Positif	
32 layan bayar lumas rugi ringan bahan rakyat cekik rakyat minggu tanggal me	1 Negatif	
33 aplikasi baharu guna login gagal susit guna	1 Negatif	
34 verifikasi susah kode otpnya kirim ulang	1 Negatif	
35 aplikasi cacat andai butuh suudi instal apk cacat	3 Negatif	
36 ganti nomor hp susah ya ampuan	3 Negatif	
37 ya daftar stuck verifikasi kodepadahadah ganti nomer pesan notif	1 Negatif	

Gambar 5. Hasil Pelabelan Data

Setelah seluruh tahapan *pre-processing* teks selesai dilakukan, tahap selanjutnya adalah pemberian label sentimen berdasarkan nilai *rating* yang diberikan oleh pengguna terhadap aplikasi JKN Mobile. *Dataset* hasil *preprocessing* dimuat kembali menggunakan *library* pandas dari file bernama *Hasil\_Preprocessing\_Data.csv*. Dari *dataset* tersebut, diambil dua kolom utama yang akan digunakan, yaitu *stemming\_data* sebagai representasi dari teks ulasan yang telah dibersihkan, dan *Rating* sebagai nilai penilaian numerik dari pengguna.

#### I. Splitting Dataset

Data akan dibagi menjadi 2 untuk keperluan *testing* dan *training*, dengan *ratio* proporsi 80% data latih dan 20% data uji coba memakai *library* ‘*train\_test\_split*’. Pemilihan rasio ini adalah untuk menyeimbangkan antara kebutuhan pelatihan model dan evaluasi performa. Dengan 80% data digunakan untuk pelatihan, model memiliki cukup banyak data untuk mempelajari pola dan karakteristik teks secara representatif. Sementara 20% sisanya digunakan untuk pengujian agar dapat menilai kemampuan model dalam mengklasifikasikan data baru yang belum pernah dilihat sebelumnya. Rasio ini juga membantu menghindari *overfitting*, yaitu kondisi ketika model terlalu menyesuaikan diri dengan data latih sehingga performanya menurun pada data uji.



Gambar 6. Visualisasi *Splitting Dataset*

### J. Hasil Analisis Data

#### 1) Evaluasi Model

Classification Report for SVM:				
	precision	recall	f1-score	support
Negatif	0.922	0.904	0.913	763.000
Positif	0.894	0.914	0.904	675.000
accuracy	0.909	0.909	0.909	0.909
macro avg	0.908	0.909	0.909	1438.000
weighted avg	0.909	0.909	0.909	1438.000

Classification Report for KNN:				
	precision	recall	f1-score	support
Negatif	0.930	0.796	0.857	763.000
Positif	0.801	0.932	0.862	675.000
accuracy	0.860	0.860	0.860	0.860
macro avg	0.865	0.864	0.859	1438.000
weighted avg	0.869	0.860	0.859	1438.000

Classification Report for Naive Bayes:				
	precision	recall	f1-score	support
Negatif	0.878	0.949	0.912	763.000
Positif	0.936	0.850	0.891	675.000
accuracy	0.903	0.903	0.903	0.903
macro avg	0.907	0.900	0.902	1438.000
weighted avg	0.905	0.903	0.902	1438.000

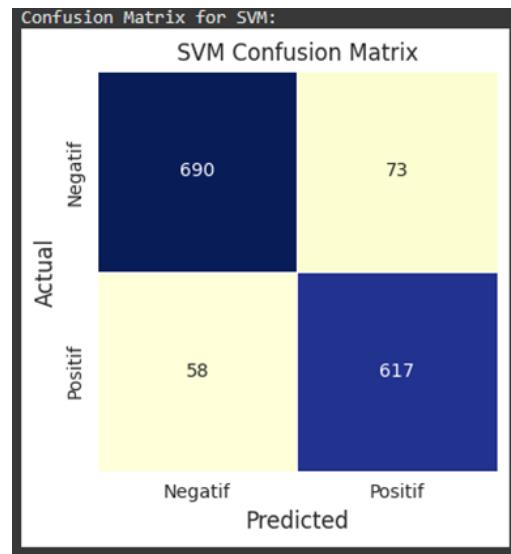
Gambar 7. Classification Report

Dengan nilai akurasi 0.909, *f1-score* kelas negatif 0,913, dan *f1-score* kelas positif 0,904, model SVM menunjukkan kinerja yang paling seimbang. Hal ini menunjukkan bahwa SVM mampu mengklasifikasikan kedua kelas secara konsisten dengan tingkat kesalahan yang rendah. Selain itu, nilai *macro average* dan *weighted average* yang tinggi mengindikasikan kestabilan model dalam menangani distribusi data.

Sementara itu, model KNN menghasilkan akurasi sebesar 0.860, dengan *f1-score* yang tidak seimbang, yakni 0.857 untuk kelas negatif dan 0.862 untuk kelas positif. Nilai *recall* pada kelas negatif cukup rendah (0.796), yang menunjukkan bahwa KNN kurang efektif dalam mengenali ulasan negatif, dan cenderung bias terhadap ulasan positif.

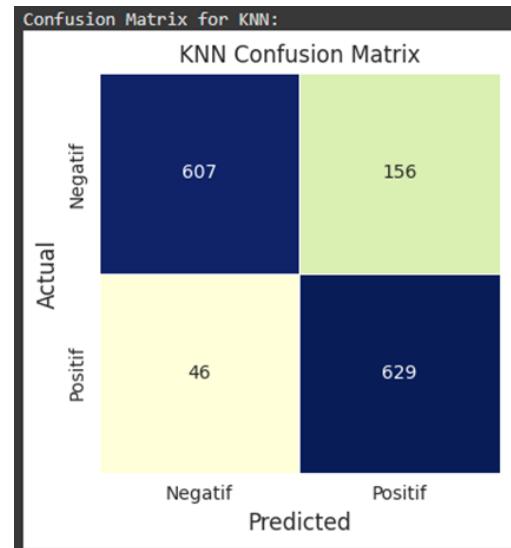
Model *Naive Bayes* juga sekaligus menunjukkan kompetitif ketepatan 0.903, dan *f1-score* sebesar 0.912 untuk kelas negatif dan 0.891 untuk kelas positif. Menariknya, *precision* untuk kelas positif pada *Naive Bayes* mencapai 0.936, tertinggi di antara ketiga model. Namun, nilai *recall*-nya tidak sebaik SVM, yang menandakan model ini sedikit kurang optimal dalam mendekripsi seluruh ulasan positif.

#### 2) Confusion Matrix



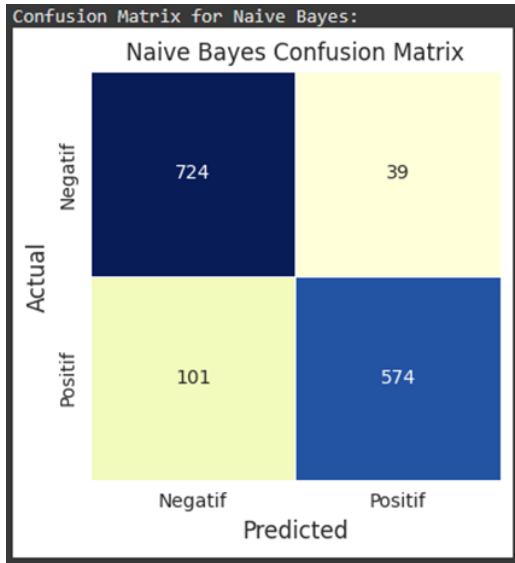
Gambar 8. Confusion Matrix SVM

Pada visualisasi di atas menunjukkan performa klasifikasi sentimen ulasan pengguna aplikasi *JKN Mobile*. Matrix tersebut menghasilkan berapa banyak tebakan tepat dan keliru pada dua kategori utama, yaitu Pro dan Kontra. Dari total data uji, sebanyak 690 data review kontra terkласifikasi dengan tepat (*True Negatives*), dan 617 data ulasan positif juga berhasil diprediksi dengan tepat (*True Positives*). Sementara itu, terdapat 73 ulasan kontra salah dikategorikan pro (*False Positives*) dan 58 review pro salah dikategorikan kontra (*False Negatives*).



Gambar 9. Confusion Matrix KNN

Berdasarkan hasil prediksi pada data uji, model KNN berhasil mengklasifikasikan 607 data ulasan negatif dengan benar (*True Negatives*) dan 629 data ulasan positif juga diprediksi dengan tepat (*True Positives*). Namun, terdapat 156 data kontra salah dikategorikan pro (*False Positives*) 46 data pro dikategorikan kontra (*False Negatives*).



Gambar 10. Confusion Matrix Naive Bayes

Berdasarkan *Output* tes terhadap *dataset* uji, model ini berhasil mengklasifikasikan 724 ulasan negatif dengan benar (*True Negatives*) dan 574 review pro juga dikategorikan tepat (*True Positives*). Namun demikian, terdapat 39 data ulasan kontra salah dikategorikan pro (*False Positives*), serta 101 ulasan pro salah dikategorikan kontra (*False Negatives*).

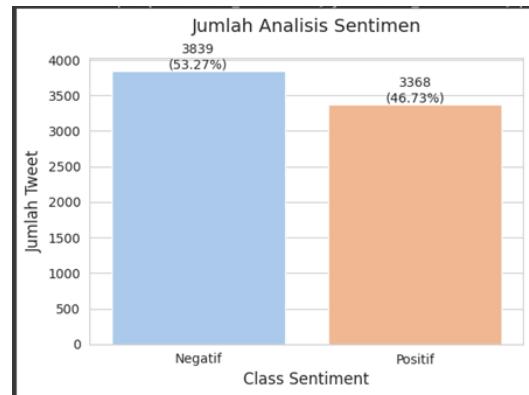
### 3) Komparasi Akurasi Model



Gambar 11. Komparasi Akurasi Model

Pada tahap evaluasi, melakukan perbandingan performa antara tiga algoritma, yaitu SVM, KNN, *Naive Bayes*. Hasil penelitian menunjukkan bahwa SVM memiliki akurasi tertinggi sebesar 90,9%, diikuti oleh *Naive Bayes* (90,3%) dan KNN (86%). Jika dibandingkan dengan penelitian sebelumnya, hasil ini sedikit lebih rendah dari studi Arsi dan Waluyo [3] yang mencapai 98,67%, namun lebih tinggi dibanding Anwar [5] dengan akurasi 73–81% dan Kusuma & Cahyono [7] dengan akurasi 82%. Secara umum, penelitian ini sejalan dengan temuan terdahulu bahwa SVM unggul dalam analisis sentimen teks, sedang kan *Naive Bayes* dan KNN tetap kompetitif, serta menunjukkan bahwa SVM efektif untuk data berdimensi tinggi seperti ulasan aplikasi Mobile JKN.

### 4) Perbandingan Sentimen



Gambar 12. Hasil Perbandingan

Berdasarkan hasil dari proses pelabelan, diketahui bahwa distribusi kelas sentimen cenderung tidak seimbang. Dari total 7.207 twitter yang dianalisis, sentimen negatif memiliki jumlah yang lebih dominan. Terdapat sebanyak 3.839 twitter atau 53.27% yang diklasifikasikan sebagai sentimen negatif, sementara 3.368 twitter sisanya 46.73% teridentifikasi sebagai sentimen positif. Jumlah 7.207 tweet merupakan hasil akhir setelah proses pembersihan dari total 10.000 review yang dikumpulkan, sehingga angka tersebut mencerminkan data valid yang digunakan dalam analisis. Hasil analisis menunjukkan bahwa mayoritas ulasan pengguna Mobile JKN bersentimen negatif (53,27%), yang mengindikasikan rendahnya kepuasan pengguna. Hal ini kemungkinan disebabkan oleh masalah teknis aplikasi, seperti eror fitur, kesulitan *login*, dan lambatnya sistem. Kondisi tersebut menjadi indikasi perlunya evaluasi dan perbaikan layanan digital BPJS Kesehatan, terutama pada aspek keandalan, kemudahan penggunaan, dan responsivitas aplikasi, agar kualitas pelayanan publik berbasis digital dapat meningkat.

## IV. KESIMPULAN

Berdasarkan hasil penelitian mengenai perbandingan algoritma klasifikasi sentimen pada ulasan pengguna aplikasi Mobile JKN, dapat disimpulkan bahwa keseluruhan proses pengolahan data, mulai dari tahap *crawling*, *preprocessing*, pelabelan sentimen, hingga pembagian dataset (*splitting*) berhasil dilakukan dengan baik menggunakan bahasa pemrograman Python dalam lingkungan Google Colab. Tiga algoritma pembelajaran mesin yang digunakan dalam penelitian ini, yaitu *Naive Bayes*, SVM, dan KNN, telah diimplementasikan dan diuji menggunakan berbagai *matrix* evaluasi seperti akurasi, *precision*, *recall*, *F1-score*, serta *confusion matrix*. Hasil evaluasi menunjukkan bahwa algoritma SVM memberikan performa terbaik dalam mengklasifikasikan sentimen ulasan, dengan tingkat akurasi mencapai 90,9%, nilai *F1-score* yang seimbang antara sentimen positif dan negatif, serta performa yang stabil. Sementara itu, algoritma *Naive Bayes* menunjukkan akurasi sebesar

90,3% dan memiliki keunggulan dalam *precision* untuk sentimen positif. Di sisi lain, algoritma KNN mencatat akurasi yang lebih rendah, yaitu sebesar 86%, serta menunjukkan kelemahan dalam mengidentifikasi sentimen negatif secara optimal. Berdasarkan temuan ini, dapat direkomendasikan bahwa algoritma SVM merupakan pendekatan yang paling efektif dan akurat untuk digunakan dalam analisis sentimen terhadap ulasan pengguna, khususnya pada aplikasi layanan publik seperti Mobile JKN.

Adapun saran untuk penelitian selanjutnya ialah dapat meningkatkan jumlah dan variasi data Untuk hasil yang lebih representatif dan generalisasi label, disarankan untuk memperluas jumlah data ulasan serta melibatkan ulasan dari aplikasi serupa atau kategori lainnya. Serta penambahan kategori sentimen seperti mempertimbangkan klasifikasi tiga kelas sentimen (positif, netral, negatif) untuk mendapatkan pemahaman yang lebih detail terhadap opini pengguna. Eksplorasi algoritma lain disarankan untuk mengeksplorasi algoritma *machine learning* lainnya seperti *Random Forest*, *Random Tree*, atau model *deep learning* seperti LSTM dan BERT untuk membandingkan performa yang lebih kompleks.

## REFERENSI

- [1] M. G. K. Lita, A. D. Mardhiyyah, I. G. A. N. S. Maharani, A. P. Mulia, and F. I. Maulana, "Sentiment Analysis of Tokopedia Product Reviews Using Naïve Bayes Algorithm," *Commun. Comput. Inf. Sci.*, vol. 2185 CCIS, no. 3, pp. 179–189, 2024.
- [2] B. S. Belay, "EFEKTIVITAS PENGGUNAAN APLIKASI MOBILE JKN DALAM MENGURANGI ANTRIAN," *Syntax Lit. J. Ilm. Indones.*, vol. 5, no. 8.5.2017, pp. 2003–2005, 2022.
- [3] P. Arsi and R. Waluyo, "Analisis Sentimen Wacana Pemindahan Ibu Kota Indonesia Menggunakan Algoritma Support Vector Machine (SVM)," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 8, no. 1, p. 147, 2021.
- [4] Normah, B. Rifai, S. Vambudi, and R. Maulana, "Analisa Sentimen Perkembangan Vtuber Dengan Metode Support Vector Machine Berbasis SMOTE," *J. Tek. Komput. AMIK BS*, vol. 8, no. 2, pp. 174–180, 2022.
- [5] K. Anwar, "Analisa sentimen Pengguna Instagram Di Indonesia Pada Review Smartphone Menggunakan Naive Bayes," *KLIK Kaji. Ilm. Inform. dan Komput.*, vol. 2, no. 4, pp. 148–155, 2022.
- [6] F. A. Larasati, D. E. Ratnawati, and B. T. Hanggara, "Analisis Sentimen Ulasan Aplikasi Dana dengan Metode Random Forest," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 6, no. 9, pp. 4305–4313, 2022.
- [7] I. H. Kusuma and N. Cahyono, "Analisis Sentimen Masyarakat Terhadap Penggunaan E-Commerce Menggunakan Algoritma K-Nearest Neighbor," *J. Inform. J. Pengemb. IT*, vol. 8, no. 3, pp. 302–307, 2023.
- [8] B. Ramadhani and R. R. Suryono, "Komparasi Algoritma Naïve Bayes dan Logistic Regression Untuk Analisis Sentimen Metaverse," *J. Media Inform. Budidarma*, vol. 8, no. 2, p. 714, 2024.
- [9] I. Cholissodin, Sutrisno, A. A. Soebroto, U. Hasanah, and Y. I. Febiola, "AI, Machine Learning & Deep Learning (Teori & Implementasi) 'from Basic Science to High Scientific Solution for Any Problem' Versi 1.01," p. 317, 2020.
- [10] I. Cholissodin and A. A. Soebroto, "AI , MACHINE LEARNING & DEEP LEARNING ( Teori & Implementasi )," no. July 2019, 2021.
- [11] U. Muhammadiyah, M. Aceh, and U. B. Nusantara, "Penggunaan Algoritma Support Vector Machine ( SVM ) Untuk Deteksi Penipuan pada Transaksi Online," vol. 13, pp. 1627–1632, 2024.
- [12] A. Mulyoto, *NAÏVE BAYES PADA GOOGLE COLABS*, 1st ed. EUREKA MEDIA AKSARA, 2024.
- [13] E. V Rahemadani, S. F. Pane, and N. H. Harani, *Algoritma C4.5 dan K-Nearest Neighbors (KNN) untuk Memetakan Matakuliah dan Keterlambatan Kelulusan Mahasiswa. in Ebook. Kreatif.*
- [14] A. Surahman, A. F. Octaviansyah, and D. Darwis, "Ekstraksi Data Produk E-Marketplace Sebagai Strategi Pengolahan Segmentasi Pasar Menggunakan Web Crawler," *Sistemasi*, vol. 9, no. 1, p. 73, 2020.
- [15] C. Carudin et al., *Buku Ajar Data Mining*. PT. Sonpedia Publishing Indonesia, 2024.