

# Analisis Sentimen Kepuasan Masyarakat terhadap Aplikasi “INFO BMKG” menggunakan Naive Bayes, SVM, dan KNN

## *Sentiment Analysis of Public Satisfaction with the 'INFO BMKG' Application using Naive Bayes, SVM, and KNN*

<sup>1</sup>Natasya Aditiya\*, <sup>2</sup>Pratomo Setiaji, <sup>3</sup>Supriyono

<sup>1,2,3</sup>Program Studi Sistem Informasi, Fakultas Teknik, Universitas Muria Kudus

<sup>1,2,3</sup>Jl. Lkr. Utara, Kayuapu Kulon, Gondangmanis, Kudus

\*e-mail: [202153032@std.umk.ac.id](mailto:202153032@std.umk.ac.id), [pratomo.setiaji@umk.ac.id](mailto:pratomo.setiaji@umk.ac.id), [supriyono.si@umk.ac.id](mailto:supriyono.si@umk.ac.id)

(received: 14 April 2025, revised: 19 April 2025, accepted: 20 April 2025)

### Abstrak

Penelitian ini bertujuan untuk menganalisis sentimen kepuasan masyarakat terhadap aplikasi Info BMKG di *Google Play Store*. Dengan semakin banyaknya pengguna aplikasi berbasis informasi, memahami bagaimana pengguna menilai aplikasi ini menjadi hal penting. Penelitian menggunakan tiga algoritma klasifikasi: *Naive Bayes*, *Support Vector Machine (SVM)*, dan *K-Nearest Neighbors (KNN)*, yang diterapkan untuk mengklasifikasikan ulasan pengguna ke dalam kategori sentimen positif, netral, atau negatif. Dataset diperoleh melalui scraping dari *Google Play Store*, mencakup data *username*, tanggal, *rating* bintang, dan komentar pengguna. Tahap *preprocessing* dilakukan untuk membersihkan data dan mempersiapkannya sebelum analisis. Penelitian ini juga mencakup pengembangan *program data mining* berbasis *web* yang mempermudah pengolahan data dan visualisasi hasil analisis. Hasil penelitian diharapkan dapat memberikan gambaran distribusi sentimen positif dan negatif terhadap aplikasi Info BMKG, serta membantu pengembang memahami faktor-faktor yang memengaruhi kepuasan pengguna. Selain itu, penelitian ini memberikan kontribusi pada pengembangan ilmu di bidang analisis sentimen dan teknologi informasi, khususnya pada aplikasi berbasis informasi bencana. Berdasarkan evaluasi model, algoritma *Naive Bayes* menunjukkan performa terbaik dengan akurasi 79.84%, presisi 60%, recall 58%, dan runtime tercepat 0.19 detik. KNN memiliki akurasi 74.35% dengan recall terendah 44%, sedangkan SVM mencapai akurasi 72.26% namun membutuhkan runtime terlama yaitu 611 detik. Validasi AUC memperkuat keunggulan *Naive Bayes* dengan nilai tertinggi pada seluruh kategori sentimen. Dengan demikian, *Naive Bayes* terbukti paling optimal untuk analisis sentimen pada studi ini, sementara KNN dan SVM memiliki keterbatasan masing-masing terutama dalam efisiensi dan ketepatan klasifikasi.

**Kata kunci:** sentimen, info bmkg, naive bayes, svm, knn

### Abstract

*This study aims to analyze public sentiment regarding the Info BMKG application on the Google Play Store. With the increasing number of users of information-based applications, understanding how users perceive and evaluate such applications has become essential. This research employs three classification algorithms—Naive Bayes, Support Vector Machine (SVM), and K-Nearest Neighbors (KNN)—to categorize user reviews into positive, neutral, or negative sentiments. The dataset was obtained through web scraping from the Google Play Store, consisting of usernames, dates, star ratings, and user comments. Data preprocessing was conducted to clean and prepare the data for analysis. Additionally, a web-based data mining application was developed to facilitate data processing and result visualization. The study aims to present the distribution of sentiment (positive, neutral, and negative) toward the Info BMKG app and help developers understand the factors that influence user satisfaction. Moreover, it contributes to the field of sentiment analysis and information technology, particularly in disaster-related information applications. Based on model evaluation, the Naive Bayes algorithm demonstrated the best performance with an accuracy of 79.84%, precision of 60%, recall of 58%, and the fastest runtime at 0.19 seconds. KNN achieved an accuracy of 74.35% with the lowest recall at 44%, while SVM had an accuracy of 72.26% but required the longest runtime at 611 seconds. AUC validation further confirmed the superiority of Naive Bayes, with the highest*

<http://sistemasi.ftik.unisi.ac.id>

scores across all sentiment categories. Thus, Naive Bayes is shown to be the most optimal for sentiment analysis in this study, whereas KNN and SVM showed certain limitations, particularly in efficiency and classification accuracy.

**Keywords:** sentiment, info bmkg, naive bayes, svm, knn

## 1 Pendahuluan

Dalam era digital yang semakin maju, penggunaan aplikasi *mobile* telah menjadi bagian integral dari kehidupan sehari-hari masyarakat. Salah satu aplikasi yang memiliki peran penting adalah aplikasi "Info BMKG," yang dirancang untuk memberikan informasi cuaca, gempa bumi, dan peringatan dini kepada pengguna. Aplikasi ini tidak hanya berfungsi sebagai sumber informasi, tetapi juga berperan dalam meningkatkan kesadaran dan kewaspadaan masyarakat terhadap potensi bencana alam [1]. Namun, seiring dengan meningkatnya jumlah pengguna, muncul berbagai ulasan yang mencerminkan kepuasan dan ketidakpuasan masyarakat terhadap aplikasi ini.

Masyarakat, sebagai pengguna, memiliki harapan tinggi terhadap aplikasi yang mereka gunakan, terutama yang berkaitan dengan keselamatan dan keamanan. Dalam banyak kasus, terdapat kesenjangan antara harapan pengguna akan keakuratan dan kemudahan penggunaan aplikasi dengan kenyataan yang mereka alami. Beberapa ulasan di *Google Play Store* menunjukkan adanya keluhan terkait fitur yang tidak berjalan dengan baik, lambatnya pembaruan informasi, dan kurangnya respons dari pihak pengembang terhadap masukan pengguna [2]. Hal ini menimbulkan pertanyaan penting tentang seberapa efektif aplikasi "Info BMKG" dalam memenuhi kebutuhan informasi masyarakat.

Dari perspektif pengembangan ilmu pengetahuan dan teknologi (IPTEK), analisis sentimen merupakan alat yang sangat berguna untuk memahami pandangan dan opini masyarakat terhadap produk dan layanan [3]. Dengan menggunakan metode analisis sentimen, peneliti dapat mengidentifikasi pola dan tren dalam ulasan pengguna, yang dapat memberikan wawasan berharga bagi pengembang aplikasi. Meskipun telah banyak dilakukan penelitian dalam bidang ini, analisis sentimen yang fokus pada aplikasi khusus seperti "Info BMKG" masih jarang dibahas, sehingga ada kesenjangan dalam literatur yang ada.

## 2 Tinjauan Literatur

*Naive Bayes* merupakan metode klasifikasi berbasis probabilitas yang sering digunakan dalam analisis teks karena kemampuannya dalam menangani data dengan dimensi tinggi secara efisien. Metode ini mengasumsikan bahwa setiap fitur dalam data bersifat independen, sehingga meskipun sederhana, sering kali memberikan hasil yang cukup baik dalam klasifikasi teks [4]. Di sisi lain, *Support Vector Machine (SVM)* bekerja dengan mencari *hyperplane* terbaik yang memisahkan data dalam ruang vektor, menjadikannya unggul dalam menangani data yang tidak terdistribusi secara linear [5]. Metode ini dikenal memiliki performa tinggi dalam tugas klasifikasi, terutama dengan penggunaan kernel yang tepat. Sementara itu, *K-Nearest Neighbors (KNN)* merupakan metode berbasis *instance-based learning* yang mengklasifikasikan data berdasarkan kedekatan dengan data latih terdekatnya [6]. Meskipun metode ini mudah dipahami, performanya sangat bergantung pada pemilihan jumlah tetangga terdekat ( $k$ ) dan dapat menjadi lambat saat menangani dataset besar.

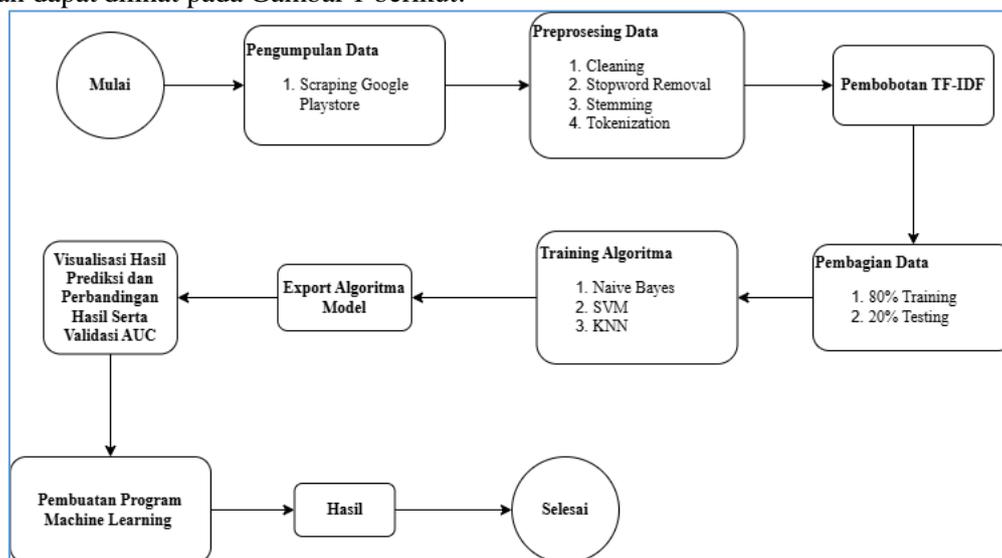
Berbagai studi telah mengimplementasikan metode-metode tersebut secara individual dalam analisis sentimen. *Naive Bayes* sering dipilih karena kesederhanaannya, tetapi kurang optimal jika fitur-fitur dalam dataset memiliki hubungan yang kuat. *SVM*, dengan keunggulannya dalam menangani data tidak linear, telah terbukti memberikan hasil akurat dalam berbagai penelitian, tetapi memerlukan tuning parameter yang lebih kompleks. *KNN* juga telah diterapkan dalam beberapa penelitian, meskipun memiliki kelemahan dalam efisiensi komputasi karena perlu membandingkan setiap data uji dengan seluruh data latih [7].

Meskipun penelitian sebelumnya telah mengkaji metode-metode tersebut dalam analisis sentimen, masih jarang ditemukan studi yang secara langsung membandingkan ketiganya dalam konteks aplikasi layanan kebencanaan di Indonesia, khususnya aplikasi Info BMKG. Selain itu, perbedaan dalam preprocessing teks dan ekstraksi fitur dalam penelitian sebelumnya sering kali membuat hasil perbandingan tidak konsisten.

Penelitian ini berkontribusi dengan menerapkan dan mengevaluasi tiga metode *machine learning* *Naive Bayes*, *SVM*, dan *KNN* dengan pendekatan preprocessing yang seragam untuk memastikan perbandingan yang lebih objektif. *Scraping* otomatis dari *Google Play Store* diterapkan guna memperoleh dataset yang lebih relevan dan terkini. Selain itu, penelitian ini tidak hanya menampilkan hasil akurasi masing-masing metode, tetapi juga membandingkan metrik evaluasi lainnya, seperti *precision*, *recall*, *F1-score*, serta *confusion matrix* serta validasi menggunakan *AUC*. Dengan demikian, penelitian ini bertujuan untuk mengisi celah penelitian sebelumnya dengan membandingkan efektivitas tiga metode klasifikasi dalam analisis sentimen terhadap aplikasi *Info BMKG*.

### 3 Metode Penelitian

Penelitian ini mencakup beberapa tahapan proses yang meliputi pengumpulan data, *preprocessing*, pembobotan *TF-IDF*, pembagian data, pelatihan model *machine learning*, evaluasi hasil, serta visualisasi analisis guna memastikan bahwa data yang digunakan dalam analisis sentimen ini memiliki tingkat prediksi dan keakuratan yang baik, untuk lebih jelasnya alur tahapan yang dilakukan dapat dilihat pada Gambar 1 berikut:



Gambar 1 Metode penelitian

#### 3.1 Pengumpulan Data

Tahapan pertama dalam penelitian ini adalah pengumpulan data, di mana data ulasan aplikasi "INFO BMKG" diperoleh dari *Google Play Store* menggunakan teknik *web scraping*. Proses *scraping* dilakukan secara otomatis di *Google Colab* menggunakan bahasa pemrograman *Python*, dengan hasil data disimpan dalam format *CSV* sebelum diproses lebih lanjut.

#### 3.2 Preprocessing Data

Setelah data diperoleh, tahap *preprocessing* dilakukan untuk memastikan bahwa teks yang digunakan dalam pelatihan model berada dalam format yang bersih dan siap untuk diproses lebih lanjut [8]. Proses ini dilakukan di *Google Colab* dan mencakup beberapa langkah penting yang bertujuan untuk meningkatkan kualitas data sebelum dianalisis lebih lanjut [9].

##### 3.2.1 Cleaning Simbol dan Karakter Khusus

Langkah pertama dalam *preprocessing* adalah *cleaning*, yaitu menghilangkan simbol, karakter khusus, angka, serta mengubah seluruh teks menjadi huruf kecil agar konsistensi data terjaga [10].

##### 3.2.2 Stopword Removal

Selanjutnya dilakukan penghapusan *stopword* untuk menghilangkan kata-kata umum yang tidak memiliki kontribusi signifikan dalam analisis sentimen, seperti "dan", "yang", "di", dan sebagainya [9].

### 3.2.3 Stemming

Setelah itu dilakukan *stemming* yaitu proses mengubah kata ke bentuk dasarnya agar variasi kata dengan makna yang sama tidak dianggap berbeda oleh model [7].

### 3.2.4 Tokenization

Kemudian teks melalui proses *tokenization* yang membagi teks menjadi kata-kata atau unit-unit yang lebih kecil untuk mempermudah analisis [11].

### 3.3 Pembobotan TF-IDF

Setelah tahap preprocessing selesai, dilakukan pembobotan *Term Frequency-Inverse Document Frequency (TF-IDF)* untuk mengubah teks menjadi representasi numerik berdasarkan frekuensi kemunculannya dalam dokumen [12].

Secara matematis, TF-IDF dihitung dengan persamaan (1) berikut:

$$TF - IDF(t, d) = TF(t, d) \times IDF(t) \quad (1)$$

Di mana:

TF (*Term Frequency*) mengukur seberapa sering kata  $t$  muncul dalam dokumen  $d$ . Dihitung dengan persamaan (2) berikut:

$$TF(t, d) = \frac{\text{jumlah kemunculan kata } t \text{ dalam dokumen } d}{\text{total kata dalam dokumen } d} \quad (2)$$

IDF (*Inverse Document Frequency*) mengukur seberapa unik suatu kata dalam keseluruhan korpus dokumen. Dihitung dengan persamaan (3) berikut:

$$IDF(t) = \log \frac{N}{DF(t)} + 1 \quad (3)$$

Di mana  $N$  adalah jumlah total dokumen dalam korpus, dan  $DF(t)$  adalah jumlah dokumen yang mengandung kata  $t$ . Semakin jarang sebuah kata muncul di seluruh dokumen, semakin besar nilai IDF-nya, sehingga kata tersebut dianggap lebih bermakna.

### 3.4 Pembagian Data

Data yang telah diproses kemudian dibagi menjadi dua bagian, yaitu 80% untuk data latih dan 20% untuk data uji.

### 3.5 Training Data

Pada tahap berikutnya, data digunakan untuk melatih model *machine learning* menggunakan tiga algoritma, yaitu *Naïve Bayes*, *Support Vector Machine (SVM)*, dan *K-Nearest Neighbors (KNN)*.

#### 3.5.1 Naïve Bayes

Algoritma *Naïve Bayes* dalam penelitian ini mengimplementasikan teknik klasifikasi probabilistik berdasarkan teorema Bayes dengan asumsi independensi antar fitur (kata). Prinsip utamanya adalah menghitung probabilitas posterior suatu dokumen termasuk dalam kelas tertentu berdasarkan frekuensi kemunculan kata-katanya [13].

Naive Bayes didasarkan pada Teorema Bayes, yang dapat dihitung dengan persamaan (4) berikut:

$$P(C|X) = \frac{P(X|C)P(C)}{P(X)} \quad (4)$$

Di mana:

$P(C|X)$  adalah probabilitas kelas  $C$  diberikan fitur  $X$  (*poterios probability*)

$P(X|C)$  adalah probabilitas fitur  $X$  diberikan kelas  $C$  (*likelihood*)

$P(C)$  adalah probabilitas awal dari kelas  $C$  (*prior probability*)

$P(X)$  adalah probabilitas dari fitur  $X$  (*evidence*), yang bisa diabaikan dalam klasifikasi karena sama untuk semua kelas.

Dalam *Multinomial Naive Bayes*, setiap dokumen diwakili sebagai vektor dari frekuensi kata-kata, dan probabilitas suatu kata muncul dalam kelas tertentu dihitung dengan persamaan (5) berikut:

$$P(W|C) = \frac{\text{count}(w,C) + a}{\sum_{w1} \text{count}(w1,C) + aV} \quad (5)$$

Di mana:

$\text{count}(w, C)$  adalah jumlah kemunculan data  $w$  dalam kelas  $C$

$a$  adalah nilai *smoothing (Laplace Smoothing)*, biasanya  $a=1$

$V$  adalah jumlah total kota unik dalam seluruh dokumen

### 3.5.2 SVM

Dalam proses pelatihan model *Support Vector Machine (SVM)* menggunakan algoritma pembelajaran mesin, bobot ( $w$ ) dan bias ( $b$ ) diperbarui berdasarkan data latih [14]. Algoritma ini mengikuti pendekatan pembaruan bobot menggunakan metode pembelajaran berbasis gradien [15]. Rumus yang digunakan dalam pembaruan bobot dan bias ditunjukkan pada persamaan (6) berikut:

$$\hat{f}(x) = w \cdot x + b \quad (6)$$

Di mana:

$\hat{f}(x)$  adalah hasil prediksi

$w$  adalah vektor bobot

$x$  adalah vektor fitur input

$b$  adalah bias

Jika hasil prediksi berbeda dengan label yang seharusnya, maka dilakukan pembaruan bobot dan bias menggunakan rumus persamaan (7 dan 8) berikut:

$$w = w + \eta y x \quad (7)$$

$$b = b + \eta y \quad (8)$$

$b$  adalah bias

$\eta$  adalah laju pembelajaran (*learning rate*)

$y$  adalah label kelas target (+1 untuk positif, -1 untuk negatif)

$x$  adalah vektor fitur dari data latih

### 3.5.3 KNN

Algoritma *KNN* yang diimplementasikan dalam penelitian menggunakan pendekatan berbasis jarak untuk klasifikasi. Berbeda dengan *SVM*, *KNN* tidak memiliki fase training eksplisit yang menghasilkan model matematis. Sebaliknya, *KNN* menyimpan seluruh dataset training dan menggunakan perhitungan jarak saat melakukan prediksi [6]. Rumus perhitungan jarak antar dua vektor teks pada algoritma *KNN* dapat dilakukan menggunakan persamaan (9) berikut:

$$d(x_1, x_2) = \sqrt{\sum_{i=1}^n (V(x_1)_i - V(x_2)_i)^2} \quad (9)$$

Di mana:

$d(x_1, x_2)$  adalah jarak antara dua vektor teks  $x_1$  dan  $x_2$

$V(x_1)_i - V(x_2)_i$  adalah nilai vektor untuk kata  $i$  dalam masing masing teks

Dalam proses klasifikasi, algoritma menghitung jarak antara dokumen baru dengan seluruh dokumen *training*, kemudian mengurutkan dokumen training berdasarkan jaraknya (dari terdekat hingga terjauh). Algoritma mengambil  $k$  tetangga terdekat (dalam kode,  $k=3$ ) dan melakukan voting mayoritas untuk menentukan kelas prediksi. Kelas yang muncul paling sering di antara  $k$  tetangga terdekat akan dipilih sebagai hasil prediksi untuk dokumen baru tersebut.

Pendekatan ini memungkinkan model *KNN* bersifat non-parametrik dan sangat lokal dalam keputusan klasifikasinya, sehingga cocok untuk pola data yang kompleks namun memiliki struktur lokal yang jelas.

## 3.6 Evaluasi dan Visualisasi Kriteria serta Validasi AUC

Setelah model dilatih dan diuji, dilakukan evaluasi hasil prediksi dengan membandingkan metrik performa dari setiap algoritma, seperti akurasi, presisi, *recall*, dan *F1-score*, serta validasi AUC (*Area Under Curve*) untuk mengukur kemampuan model dalam membedakan kelas sentimen. Selain itu, hasil analisis divisualisasikan dalam bentuk diagram perbandingan metrik dan AUC untuk menunjukkan efektivitas masing-masing algoritma dalam mengklasifikasikan sentimen pengguna.

### 3.6.1 Akurasi

Akurasi adalah metrik evaluasi yang mengukur seberapa sering model melakukan prediksi yang benar [16]. Dihitung dengan membagi jumlah prediksi benar (TP + TN) dengan total prediksi (TP + TN + FP + FN) atau dapat dihitung menggunakan persamaan (10) berikut:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (10)$$

### 3.6.2 Presisi

Presisi mengukur seberapa akurat model dalam mengklasifikasikan data positif [17]. Dihitung dengan membagi jumlah prediksi positif yang benar (TP) dengan total prediksi positif (TP + FP) atau dapat dihitung dengan persamaan (11) berikut.

$$Precision = \frac{TP}{TP+FP} \quad (11)$$

### 3.6.3 Recall

Recall mengukur kemampuan model dalam menemukan semua data positif yang sebenarnya [18]. Dihitung dengan membagi jumlah prediksi positif yang benar (TP) dengan total data positif aktual (TP + FN) atau dapat dihitung dengan persamaan (12) berikut.

$$Recall = \frac{TP}{TP+FN} \quad (12)$$

### 3.6.4 F1-Score

F1-Score adalah metrik yang menggabungkan presisi dan recall dalam satu nilai harmonik untuk memberikan gambaran keseimbangan antara keduanya [19] yang dapat dihitung dengan persamaan (13) berikut:

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision+Recall} \quad (13)$$

### 3.6.5 Validasi Area Under Curve (AUC)

AUC merupakan metrik evaluasi yang mengukur sejauh mana model mampu membedakan antara kelas positif dan negatif maupun netral. Semakin tinggi nilai AUC, semakin baik model dalam mengklasifikasikan sentimen dengan benar [20]. Dapat dihitung dengan persamaan (14) berikut:

$$AUC = \sum \left( \frac{TPR_i + TPR_{i-1}}{2} \right) \times (FPR_i - FPR_{i-1}) \quad (14)$$

Di mana:

$$TPR \text{ (True Positive Rate)} = \frac{TP}{TP+FN}$$

$$FPR \text{ (False Positive Rate)} = \frac{FP}{FP+TN}$$

TP = Jumlah prediksi positif yang benar

FP = Jumlah prediksi positif yang salah

TN = Jumlah prediksi negatif yang benar

FN = Jumlah prediksi negatif yang salah

Model dengan nilai AUC mendekati 1 dianggap memiliki performa klasifikasi yang sangat baik, sementara nilai mendekati 0.5 menunjukkan bahwa model tidak lebih baik dari prediksi acak.

## 3.7 Pembuatan Program Machine Learning

Tahapan akhir dalam penelitian ini adalah pembuatan program *machine learning* analisis sentimen berbasis *web* dengan menggunakan bahasa pemrograman PHP dan *framework CodeIgniter* 3. Program ini memungkinkan pengguna untuk melakukan analisis sentimen secara otomatis berdasarkan input ulasan baru. Model *machine learning* yang digunakan dalam sistem ini terdiri dari algoritma *Naive Bayes*, *SVM*, dan *KNN*, yang diperoleh melalui pola-pola dari proses *training* yang telah dilakukan sebelumnya di *Google Colab*. Selain itu, terdapat juga proses *scraping* data ulasan dari *Google Play Store* yang dilakukan secara otomatis menggunakan *Python* dan terintegrasi dalam sistem. Dengan kombinasi teknologi ini, sistem dapat mengumpulkan, mengolah, dan menganalisis data secara efisien serta menampilkan hasil prediksi secara interaktif [21].

## 4 Hasil dan Pembahasan

Penulis melakukan serangkaian eksperimen dan analisis untuk mengukur performa model dalam mengklasifikasikan sentimen pada ulasan aplikasi "INFO BMKG". Proses penelitian mencakup tahap pengumpulan data, preprocessing, pembobotan fitur, pelatihan model menggunakan beberapa algoritma *machine learning*, serta evaluasi hasil prediksi. Pada bagian ini, akan dipaparkan secara rinci hasil yang diperoleh pada setiap tahapan, disertai pembahasan terkait performa masing-masing metode berdasarkan metrik evaluasi yang telah ditentukan.

#### 4.1 Pengumpulan Data

Penelitian ini mengumpulkan ulasan aplikasi "INFO BMKG" dari *Google Play Store* menggunakan teknik *web scraping* dengan *Python* dan *library google-play-scraper*. Data yang diambil mencakup username, tanggal ulasan, rating bintang, dan komentar, dengan total 3.100 ulasan yang disimpan dalam format CSV.

Untuk menghindari deteksi spam, diterapkan jeda satu detik setiap permintaan. *Scraping* dilakukan dalam batch berisi 100 ulasan per permintaan, dimulai dari yang terbaru. Sistem juga memeriksa duplikasi berdasarkan kombinasi *username* dan komentar. Jika token pengambilan data habis, proses akan berhenti otomatis. Data hasil *scraping* ini akan digunakan untuk tahap *preprocessing* dan analisis sentimen, sebagaimana ditunjukkan pada Gambar 2 berikut:

index	Username	Tanggal	Bintang	Komentar
0	Antok Plontos	2025-03-03	1	Beda dgn aplikasi yg lama,,yg ini hampir gak prnh dpt notifikasi..cm sekali2 aja dpt notif gempu,pdhl hampir tiap HR ada gempa di wilayah indonesia.
1	Azraqi Warhangan	2025-03-03	2	Tambah susah baca perkiraan gelombang laut Kembalikan ke model lama ya pak
2	maryadi amer	2025-03-02	2	Habis update mlah ngk bisa ganti lokasi
3	Wawax Wawax	2025-03-02	5	Bgus
4	angky	2025-03-01	1	Perubahan Cuaca bukannya di update setiap saat, tapi hanya lewat pemberitahuan saja.. Kalah jauh sama Aplikasi serupa yg bisa diatur updatenya setiap saat.. Ayo maju lagi BMKG.. Jangan mau kalah sama yg free.. Wkwkwk...
5	Aris Kurniawan	2025-02-28	1	Setela update malah semakin banyak bug. Gps tidak bisa deteksi,citra satelit pun susah untuk dibuka.
6	Adneil's Project	2025-02-27	2	Ini Gimana om.. lokasi ke lock Gunung Sahari... Jadi ngaco lokasi.. ganti ganti tetap patokan gunung Sahari .. tolong kembalikan posisi GPS nya yang bener..
7	aziz azhari	2025-02-27	5	Kenapa citra satelit himawari tidak muncul lagi ya, selalu loading saat dibuka, padahal sangat berguna buat saya tau arah angin membawa awan hujan, mudah2an bisa diperbaiki oleh BMKG
8	Botol KecapS Tech	2025-02-27	2	Tidak terlalu akurat
9	Elangdirga Nuswantara	2025-02-25	3	Mohon beri tambahan ada vibration meter, agar wilayah lain dapat mengukur getaran di daerah yang belum ter-cover

Gambar 2. Data hasil scraping ulasan google playstore

#### 4.2 Preprocessing Data

Langkah pertama dalam *preprocessing* adalah membaca *dataset* yang telah disimpan dalam format CSV. Setiap ulasan kemudian diperiksa untuk memastikan tidak kosong atau hanya mengandung simbol. Jika ulasan tidak memiliki teks yang bermakna, maka akan langsung dikategorikan berdasarkan jumlah bintang yang diberikan oleh pengguna. Kategori sentimen ditentukan dengan aturan berikut: bintang 1 dan 2 dikategorikan sebagai negatif, bintang 3 sebagai netral, sedangkan bintang 4 dan 5 sebagai positif.

Selanjutnya, dilakukan proses *cleaning* dengan mengubah semua teks menjadi huruf kecil (*lowercase*), menghapus simbol, angka, serta karakter khusus yang tidak diperlukan. Setelah pembersihan, teks akan melalui tahap tokenisasi, yaitu pemisahan teks menjadi kata-kata individu menggunakan pustaka *Natural Language Toolkit (NLTK)*. Kata-kata hasil tokenisasi kemudian disaring untuk menghapus stopwords, yaitu kata-kata umum yang tidak memiliki makna signifikan dalam analisis sentimen.

Setelah proses tokenisasi dan penghapusan *stopwords*, dilakukan tahap *stemming*, yaitu mengubah kata-kata ke dalam bentuk dasarnya menggunakan pustaka *Sastrawi*. Proses *stemming* ini membantu dalam menyederhanakan variasi kata sehingga model dapat mengenali pola dengan lebih baik. Hasil akhir dari *preprocessing* mencakup dua kolom baru dalam dataset, yaitu "Komentar\_Cleaned" yang berisi teks hasil stemming dan "Komentar\_Tokenized" yang berisi daftar kata-kata setelah tokenisasi.

Dataset yang telah diproses kemudian disimpan dalam file CSV baru. Data hasil *preprocessing* ini akan digunakan dalam tahap selanjutnya, yaitu pelatihan model *machine learning* untuk analisis sentimen. Struktur data hasil *preprocessing* ditunjukkan pada Gambar 3 berikut:

index	Username	Tanggal	Bintang	Komentar	Komentar_Cleaned	Komentar_Tokenized	Label
0	Antok Plontos	2025-03-03	1	Beda dgn aplikasi yg lama.,yg ini hampir gak prnh dpt notifikasi. cm sekali2 aja dpt notif gempapdhl hampir tiap HR ada gempap di wilayah indonesia.	beda dgn aplikasi yg lamayg hampir gak prnh dpt notifikasicm sekali aja dpt notif gempapdhl hampir tiap hr ada gempap wilayah indonesia	['beda', 'dgn', 'aplikasi', 'yg', 'lamayg', 'hampir', 'gak', 'prnh', 'dpt', 'notifikasicm', 'sekali', 'aja', 'dpt', 'notif', 'gempapdhl', 'hampir', 'tiap', 'hr', 'ada', 'gempap', 'wilayah', 'indonesia']	negatif
1	Azraqi Warhangan	2025-03-03	2	Tambah susah baca perkiraan gelombang laut Kembalikan ke model lama ya pak	tambah susah baca kira gelombang laut kembali model lama ya pak	['tambah', 'susah', 'baca', 'kira', 'gelombang', 'laut', 'kembali', 'model', 'lama', 'ya', 'pak']	negatif
2	maryadi amer	2025-03-02	2	Habis update mlah ngk bisa ganti lokasi	habis update mlah ngk bisa ganti lokasi	['habis', 'update', 'mlah', 'ngk', 'bisa', 'ganti', 'lokasi']	negatif
3	Wawax Wawax	2025-03-02	5	Bgus	bgus	['bgus']	positif
4	angky	2025-03-01	1	Perubahan Cuaca bukannya di update setiap saat, tapi hanya lewat pemberitahuan saja.. Kalah jauh sama Aplikasi serupa yg bisa diatur update nya setiap saat.. Ayo maju lagi BMKG. Jangan mau kalah sama yg free.. Wkwkwk...	ubah cuaca bukan update tiap saat tapi hanya lewat pemberitahuan saja kalah jauh sama aplikasi rupa yg bisa atur update nya tiap saat ayo maju lagi bmgk jangan mau kalah sama yg free wkwkwk	['ubah', 'cuaca', 'bukan', 'update', 'tiap', 'saat', 'tapi', 'hanya', 'lewat', 'pemberitahuan', 'saja', 'kalah', 'jauh', 'sama', 'aplikasi', 'rupa', 'yg', 'bisa', 'atur', 'update nya', 'tiap', 'saat', 'ayo', 'maju', 'lagi', 'bmgk', 'jangan', 'mau', 'kalah', 'sama', 'yg', 'free', 'wkwkwk']	negatif
5	Aris Kurniawan	2025-02-28	1	Setela update malah semakin banyak bug. Gps tidak bisa deteksi.citra satelit pun susah untuk dibuka.	setela update malah makin banyak bug gps tidak bisa deteksicitra satelit pun susah buka	['setela', 'update', 'malah', 'makin', 'banyak', 'bug', 'gps', 'tidak', 'bisa', 'deteksicitra', 'satelit', 'pun', 'susah', 'buka']	negatif
6	Adneif's Project	2025-02-27	2	Ini Gimana om.. lokasi ke lock Gunung Sahari.. Jadi ngaco lokasi. ganti ganti tetap patokan gunung Sahari .. tolong kembalikan posisi GPS nya yang bener..	gimana om lokasi lock gunung sahari jadi ngaco lokasi ganti ganti tetap patok gunung sahari tolong kembali posisi gps nya bener	['gimana', 'om', 'lokasi', 'lock', 'gunung', 'sahari', 'jadi', 'ngaco', 'lokasi', 'ganti', 'ganti', 'tetap', 'patok', 'gunung', 'sahari', 'tolong', 'kembali', 'posisi', 'gps', 'nya', 'bener']	negatif
7	aziz azhari	2025-02-27	5	Kenapa citra satelit himawari tidak muncul lagi ya, selalu loading saat dibuka, padahal sangat berguna buat saya tau arah angin membawa awan hujan, mudah2an bisa diperbaiki oleh BMKG	kenapa citra satelit himawari tidak muncul lagi ya selalu loading saat buka padahal sangat guna buat saya tau arah angin bawa awan hujan mudah bisa baik oleh bmgk	['kenapa', 'citra', 'satelit', 'himawari', 'tidak', 'muncul', 'lagi', 'ya', 'selalu', 'loading', 'saat', 'buka', 'padahal', 'sangat', 'guna', 'buat', 'saya', 'tau', 'arah', 'angin', 'bawa', 'awan', 'hujan', 'mudah', 'bisa', 'baik', 'oleh', 'bmgk']	positif
8	Botol KecapS Tech	2025-02-27	2	Tidak terlalu akurat	tidak terlalu akurat	['tidak', 'terlalu', 'akurat']	negatif
9	Elangdirga Nuswantara	2025-02-25	3	Mohon beri tambahan ada vibration meter, agar wilayah lain dapat mengukur getaran di daerah yang belum ter-cover	mohon beri tambah ada vibration meter agar wilayah lain dapat ukur getar daerah belum tercover	['mohon', 'beri', 'tambah', 'ada', 'vibration', 'meter', 'agar', 'wilayah', 'lain', 'dapat', 'ukur', 'getar', 'daerah', 'belum', 'tercover']	netral

Gambar 3. Data hasil preprosesing ulasan google playstore

### 4.3 Pembobotan TF-IDF

Teks ulasan yang telah dibersihkan dan ditokenisasi selanjutnya diubah menjadi representasi numerik menggunakan metode *Term Frequency-Inverse Document Frequency (TF-IDF)*. Hasil transformasi TF-IDF ini ditunjukkan pada Gambar 4 berikut:

index	Term	TF	IDF	TF-IDF
0	aa	0.5102571633108025	8.346332738540504	4.258776067215759
1	aamiin	0.2870138077021461	8.346332738540504	2.3955127396375904
2	aamiin	0.2346171218101972	8.346332738540504	1.958192564786594
3	abc	0.5514190874597786	8.346332738540504	4.602327182321679
4	abdate	0.3792936189004869	8.346332738540504	3.165710748948639
5	abdetan	0.2834159931609803	8.346332738540504	2.3654841823454613
6	abgred	0.3634551629246652	8.346332738540504	3.033517725309706
7	abiis	0.5245302506279511	8.346332738540504	4.377904003170924
8	abis	2.666577169940515	7.247720449872394	19.326605885740726
9	accuweather	0.17685717293681824	8.346332738540504	1.4761088125282857

Gambar 4. Data hasil TF-IDF

Dalam implementasi ini, data ulasan yang telah melalui preprocessing dibaca dari file CSV, kemudian vektor TF-IDF dibuat menggunakan *TfidfVectorizer* dari pustaka *scikit-learn*. Proses ini menghasilkan tiga nilai utama: TF, IDF, dan TF-IDF untuk setiap kata unik dalam *dataset*. Setelah perhitungan selesai, hasilnya disimpan dalam bentuk tabel yang berisi daftar kata unik beserta nilai TF, IDF, dan TF-IDF-nya. Data ini kemudian diekspor ke dalam file CSV baru agar dapat digunakan untuk tahap selanjutnya dalam analisis sentimen menggunakan *machine learning*.

Setelah proses *preprocessing* dan perhitungan TF-IDF, langkah berikutnya adalah melakukan training dan evaluasi model *Naive Bayes* untuk analisis sentimen. Model yang digunakan dalam implementasi ini adalah *Multinomial Naive Bayes*, yang sering digunakan dalam pemrosesan teks karena efektif dalam menangani data diskrit seperti frekuensi kata.

### 4.4 Training Data

Setelah melalui tahap preprocessing, data yang telah dibersihkan dan diberi label kemudian digunakan sebagai training data untuk proses pelatihan model. Training data berperan penting dalam membentuk pola pembelajaran mesin agar mampu mengenali karakteristik teks ulasan berdasarkan

sentimen positif maupun negatif. Pada penelitian ini, beberapa metode pembelajaran mesin diterapkan untuk mengklasifikasikan data latih, seperti *Naïve Bayes*, *Support Vector Machine (SVM)*, dan *K-Nearest Neighbor (KNN)*. Hasil pelatihan dari masing-masing metode kemudian dievaluasi menggunakan metrik seperti akurasi, presisi, *recall*, dan *f1-score* untuk melihat performa klasifikasi sentimen terhadap ulasan aplikasi INFO BMKG.

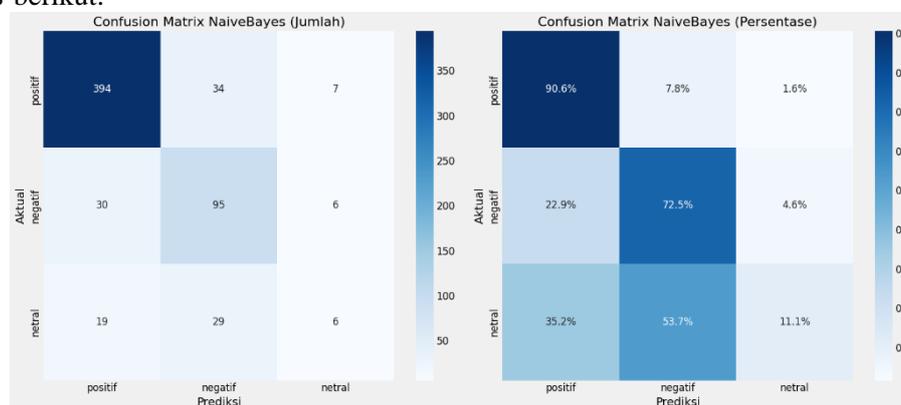
#### 4.4.1 Naïve Bayes

Setelah model *Naive Bayes* dilatih menggunakan *dataset* yang telah melalui tahap *preprocessing*, evaluasi dilakukan untuk mengukur performa model dalam mengklasifikasikan sentimen pengguna terhadap aplikasi Info BMKG. Hasil metriks evaluasi *Naive Bayes* ditampilkan pada Tabel 1 berikut:

**Tabel 1. Metriks metode naïve bayes**

No	Naïve Bayes Metriks	Jumlah
1.	Akurasi	79.84%
2.	Presisi	60.21%
3.	Recall	58.07%
4.	F1-Score	57.31%
5.	Runtime	0.1857 detik

Berdasarkan hasil pengujian, model mencapai akurasi sebesar 79.84%, yang berarti sekitar 78% prediksi yang dilakukan oleh model sesuai dengan label aslinya. Meskipun akurasi cukup tinggi, metrik lainnya seperti presisi 60.21%, *recall* 58.07%, dan *F1-score* 57.31% menunjukkan bahwa model masih mengalami kesulitan dalam membedakan beberapa kategori sentimen. Nilai *F1-score* yang relatif rendah mencerminkan bahwa terdapat ketidakseimbangan antara kemampuan model dalam mengenali kelas yang benar dan menghindari kesalahan klasifikasi. Model membutuhkan 0.1857 detik untuk melakukan proses pelatihan dan prediksi, yang menunjukkan bahwa model bekerja dengan cepat dan efisien. Untuk hasil yang lebih jelas ditunjukkan pada Gambar 5 *Confusion Matrix Naïve Bayes* berikut:



**Gambar 5. Confusion matrix naïve bayes**

Berdasarkan confusion matrix *Naive Bayes*, model memiliki performa terbaik dalam mengklasifikasikan sentimen positif dengan akurasi 90.6%, serta cukup baik dalam mengenali sentimen negatif 72.5%. Namun, model masih kesulitan mengklasifikasikan sentimen netral, yang sering salah dikategorikan sebagai positif 35.2% atau negatif 53.7%. Secara jumlah, 394 data positif diklasifikasikan dengan benar, sementara 34 salah sebagai negatif dan 7 sebagai netral. Untuk negatif, 95 sampel benar, tetapi 30 salah sebagai positif dan 6 sebagai netral. Sentimen netral paling sulit dikenali, dengan akurasi hanya 11.1%.

#### 4.4.2 SVM

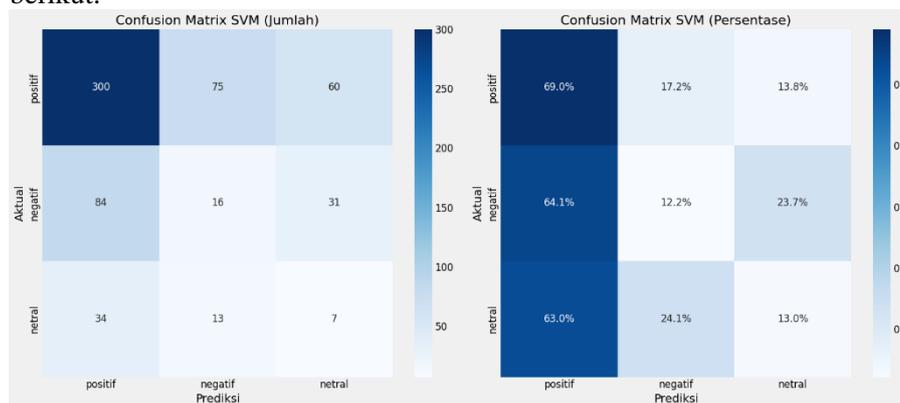
Model *SVM* yang telah dilatih pada dataset sentimen Info BMKG menunjukkan hasil evaluasi dengan akurasi sebesar 72.26%, yang lebih rendah dibandingkan dengan model *Naive Bayes*.

Meskipun akurasi cukup rendah, metrik lainnya seperti presisi 55.24%, *recall* 53.60%, dan *F1-score* 53.20% menunjukkan bahwa model masih mengalami kesulitan dalam mengklasifikasikan sentimen dengan baik. Nilai presisi yang sedikit lebih tinggi dibandingkan *recall* mengindikasikan bahwa model lebih selektif dalam memberikan prediksi positif, namun tetap memiliki tingkat kesalahan yang cukup tinggi. Hasil metrik evaluasi SVM ditampilkan pada Tabel 2 berikut:

**Tabel 2. Metriks metode SVM**

No	SVM Metriks	Jumlah
1.	Akurasi	72.26%
2.	Presisi	55.24%
3.	Recall	53.60%
4.	F1-Score	53.20%
5.	Runtime	611 detik

Salah satu faktor yang perlu diperhatikan dalam penerapan SVM adalah waktu komputasi. Dari hasil evaluasi, model SVM membutuhkan waktu 611 detik untuk menyelesaikan proses pelatihan dan prediksi, jauh lebih lama dibandingkan dengan model *Naive Bayes*. Waktu komputasi yang tinggi ini disebabkan oleh sifat SVM yang memerlukan perhitungan kompleks dalam menentukan hyperplane pemisah antar kelas. Untuk hasil yang lebih jelas ditunjukkan pada Gambar 6 *Confusion Matrix SVM* berikut:



**Gambar 6. Confusion matrix SVM**

Berdasarkan *confusion matrix SVM*, model lebih akurat dalam mengenali sentimen positif 72.26% dibanding negatif 12.2%. Namun, sentimen netral sering salah dikategorikan sebagai positif 63.0% atau negatif 24.1%, menunjukkan kesulitan model dalam membedakan kategori ini. Secara jumlah, 300 data positif diklasifikasikan dengan benar, tetapi 75 salah sebagai negatif dan 60 sebagai netral. Untuk negatif, hanya 16 sampel yang benar, sedangkan 84 salah sebagai positif dan 31 sebagai netral. Sentimen netral juga kurang dikenali, dengan hanya 7 sampel yang benar.

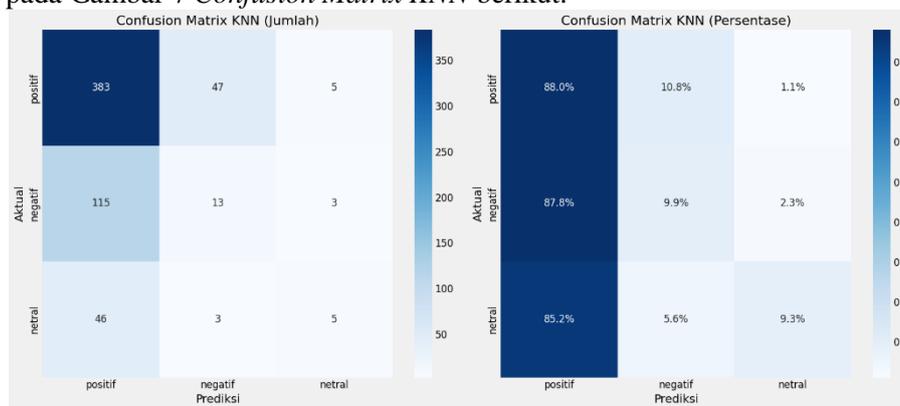
#### 4.4.3 KNN

Model *K-Nearest Neighbors (KNN)* yang digunakan dalam analisis sentimen aplikasi Info BMKG menghasilkan akurasi sebesar 74.35%, yang lebih rendah dibandingkan dengan model *Naive Bayes*, tetapi lebih baik daripada SVM. Dari segi metrik lainnya, presisi sebesar 52.90%, *recall* sebesar 43.52%, dan *F1-score* sebesar 44.55% menunjukkan bahwa model masih mengalami kesulitan dalam memberikan prediksi yang konsisten. Nilai *recall* yang lebih rendah mengindikasikan bahwa model sering gagal mengenali ulasan dengan benar, terutama untuk kelas sentimen yang lebih jarang muncul dalam *dataset*. Selain itu, runtime sebesar 10.3215 detik menunjukkan bahwa meskipun KNN memiliki performa yang lebih baik dibandingkan SVM dalam hal efisiensi waktu, model ini masih lebih lambat dibandingkan *Naive Bayes*. *F1-score* yang tidak terlalu tinggi juga menunjukkan bahwa keseimbangan antara presisi dan *recall* masih perlu ditingkatkan agar performa model lebih optimal. Hasil metrik evaluasi KNN ditampilkan pada Tabel 3 berikut:

**Tabel 3. Metriks metode KNN**

No	KNN Metriks	Jumlah
1.	Akurasi	74.35%
2.	Presisi	52.90%
3.	Recall	43.52%
4.	F1-Score	44.55%
5.	Runtime	10.32 detik

Dari sisi efisiensi, model *KNN* memiliki waktu eksekusi sebesar 10.32 detik, yang jauh lebih cepat dibandingkan dengan *SVM*, tetapi masih lebih lambat dibandingkan *Naive Bayes*. Hal ini disebabkan oleh sifat *KNN* yang membutuhkan perhitungan jarak untuk setiap titik data selama prediksi, yang bisa menjadi lebih lambat jika dataset semakin besar. Untuk hasil yang lebih jelas ditunjukkan pada Gambar 7 *Confusion Matrix KNN* berikut:



**Gambar 7. Confusion matrix KNN**

Berdasarkan confusion matrix *KNN*, model cukup baik dalam mengenali sentimen positif 88.0%, tetapi kurang optimal dalam membedakan negatif 9.9% dan netral 9.3%. Sebanyak 383 data positif diklasifikasikan dengan benar, namun 47 salah sebagai negatif dan 5 sebagai netral. Sentimen negatif memiliki kesalahan tinggi, dengan 115 data salah sebagai positif dan hanya 13 yang benar. Kategori netral juga sering diklasifikasikan sebagai positif 85.2%, menunjukkan bahwa model masih kesulitan membedakan sentimen ini.

#### 4.4.4 Evaluasi dan Perbandingan Hasil

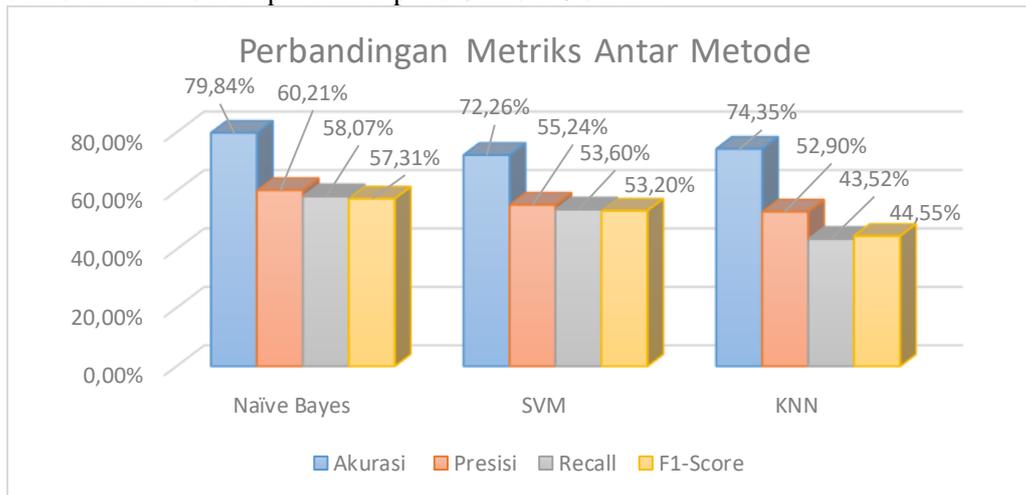
Evaluasi dilakukan untuk membandingkan performa model *Naive Bayes*, *SVM*, dan *KNN* dalam mengklasifikasikan sentimen. Hasil evaluasi ditampilkan pada Tabel 4 berikut:

**Tabel 4. Perbandingan hasil metriks masing-masing metode**

No	Metriks	Naive Bayes	SVM	KNN
1.	Akurasi	79.84%	72.26%	74.35%
2.	Presisi	60.21%	55.24%	52.90%
3.	Recall	58.07%	53.60%	43.52%
4.	F1-Score	57.31%	53.20%	44.55%
5.	Runtime	0.1857 detik	611 detik	10.32 detik

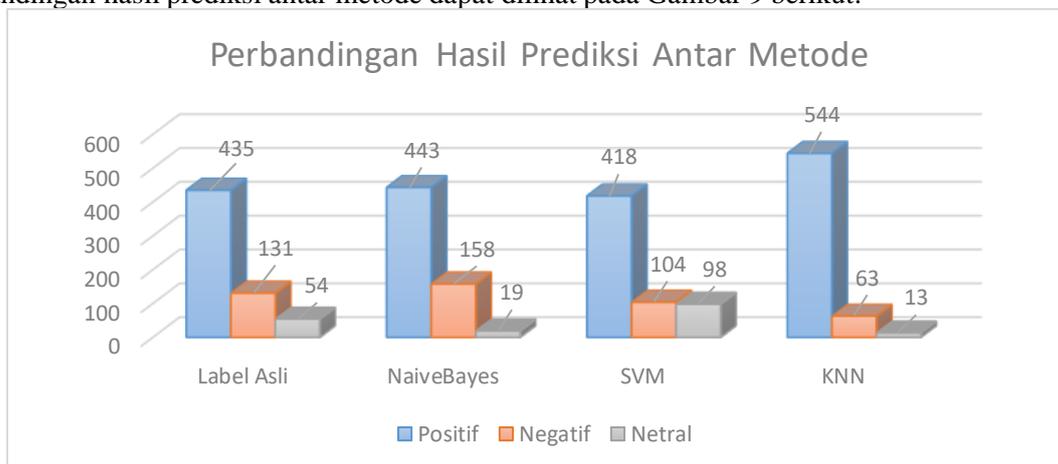
Berdasarkan hasil evaluasi tiga metode klasifikasi, *Naive Bayes* menunjukkan performa terbaik dalam analisis sentimen aplikasi Info BMKG. Model ini memiliki akurasi tertinggi sebesar 79.84%, mengungguli *KNN* 74.35% dan *SVM* 72.26%. Selain itu, presisi, recall, dan *F1-score* *Naive Bayes* juga lebih unggul, masing-masing sebesar 60.21%, 58.07%, dan 57.31%, dibandingkan *SVM*

55.24%, 53.60%, 53.20% dan KNN 52.90%, 43.52%, 44.55%. Hal ini menunjukkan bahwa *Naive Bayes* memiliki keseimbangan yang lebih baik antara presisi dan *recall*, menjadikannya metode paling akurat dalam mengklasifikasikan sentimen pengguna. Untuk lebih jelasnya mengenai perbandingan hasil metrik antar metode dapat dilihat pada Gambar 8 berikut:



**Gambar 8. Perbandingan metrik masing masing metode**

Dari sisi runtime, *Naive Bayes* juga yang paling efisien dengan waktu eksekusi hanya 0.1857 detik, jauh lebih cepat dibandingkan *KNN* 10.32 detik dan *SVM* 611 detik. Runtime yang lebih cepat membuat *Naive Bayes* lebih cocok untuk analisis skala besar seperti ulasan di *Google Play Store*. Sebaliknya, *SVM* memiliki runtime terlama karena kompleksitas optimasi bobotnya, sedangkan *KNN* membutuhkan waktu lebih lama akibat perhitungan jarak antar data yang besar. Untuk lebih jelasnya perbandingan hasil prediksi antar metode dapat dilihat pada Gambar 9 berikut:



**Gambar 9. Perbandingan hasil prediksi antar metode**

Data yang digunakan dalam eksperimen ini telah dibagi dengan skema 80% untuk *training* dan 20% untuk *testing*. Dari grafik terlihat bahwa semua metode cenderung mengklasifikasikan lebih banyak data ke dalam kategori positif dibandingkan dengan label aslinya.

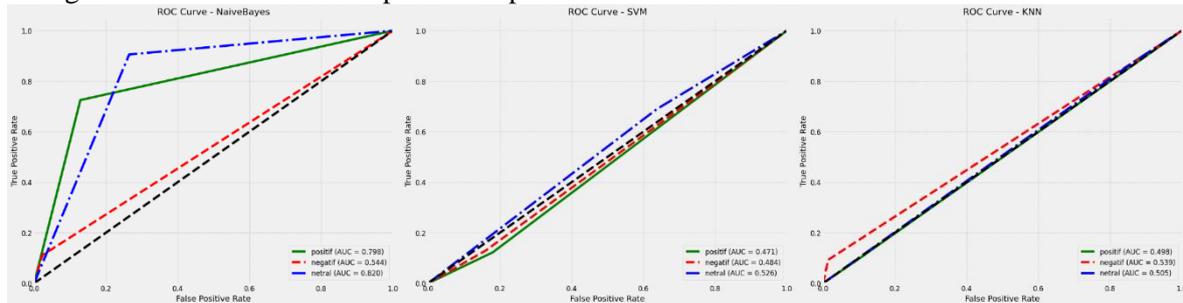
*Naive Bayes* memberikan hasil prediksi yang paling mendekati label asli, dengan 443 data positif, 158 negatif, dan 19 netral, menunjukkan peningkatan kecil pada prediksi negatif dan penurunan pada prediksi netral dibandingkan label asli.

*SVM* memiliki kecenderungan untuk meningkatkan prediksi negatif 104 dibandingkan 131 pada label asli namun lebih banyak mengklasifikasikan data sebagai netral 98 dibandingkan 54 pada label asli.

*KNN* menunjukkan bias yang signifikan terhadap kategori positif, dengan 544 prediksi positif, jauh lebih tinggi dibandingkan 435 pada label asli, sementara prediksi negatifnya 63 jauh lebih rendah dari label asli 131 dan kategori netralnya sangat sedikit 13.

#### 4.4.5 Validasi Area Under Curve (AUC)

Validasi model dilakukan menggunakan *Area Under Curve (AUC)* untuk mengukur seberapa baik masing-masing metode dalam membedakan antara sentimen positif, negatif, dan netral. Nilai AUC dihitung berdasarkan *ROC (Receiver Operating Characteristic) Curve* untuk setiap kelas dan dirata-ratakan menggunakan *macro average* guna memperoleh gambaran umum performa model. Hasil grafik *AUC ROC Curve* dapat dilihat pada Gambar 10 berikut:



**Gambar 10. AUC ROC curve naïve bayes, SVM, KNN**

Berdasarkan hasil validasi, metode *Naïve Bayes* menunjukkan performa kemampuan yang baik dalam membedakan kategori positif 0.798 dan netral 0.820, meskipun performanya pada kategori negatif 0.544 masih perlu ditingkatkan. Sementara itu metode *SVM* memiliki nilai AUC terendah, yang menunjukkan bahwa model ini kurang mampu membedakan antara kelas-kelas sentimen dengan baik. Hal ini terlihat dari nilai AUC yang rendah pada semua kategori, terutama untuk positif 0.471 dan negatif 0.484 serta netral hanya 0.526. Hasil perbandingan AUC dapat dilihat pada Tabel 5 berikut:

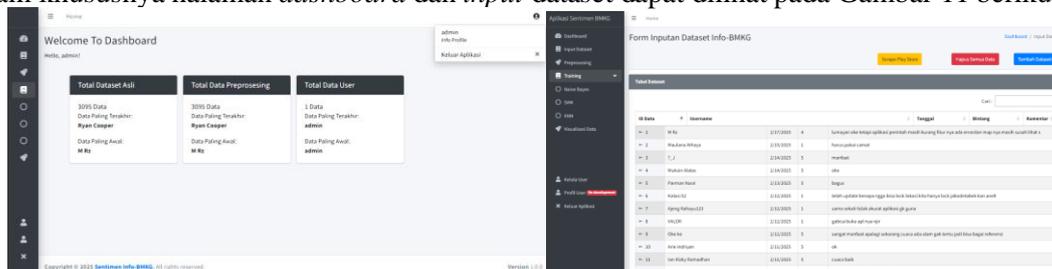
**Table 5. Perbandingan hasil AUC masing-masing metode**

No	Metode	Positif	Negatif	Netral
1.	Naïve Bayes	0.798	0.544	0.820
2.	SVM	0.471	0.484	0.526
3.	KNN	0.498	0.539	0.505

Metode *KNN* memiliki nilai sedikit lebih baik dari *SVM*, tetapi masih jauh di bawah *Naïve Bayes*. Meskipun performa *KNN* dalam membedakan kategori positif 0.498, negatif 0.539, netral 0.505 lebih baik daripada *SVM*, metode ini tetap menunjukkan kelemahan dalam menangani variasi data sentimen.

#### 4.4.6 Program Machine Learning

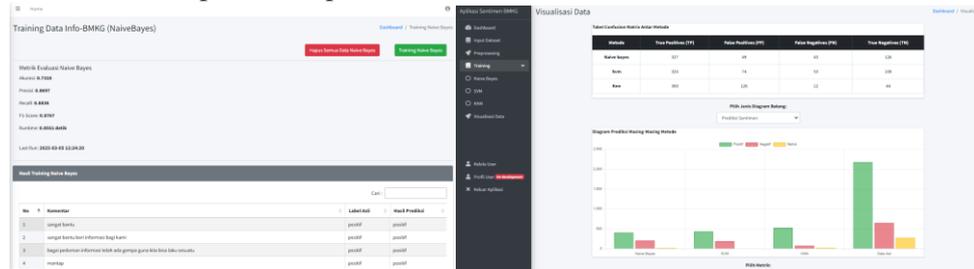
Program analisis sentimen ini dikembangkan menggunakan *PHP* dengan *framework CodeIgniter 3* dan *database SQL*. *Front-end web* dirancang dengan *AdminLTE* untuk tampilan yang responsif dan interaktif. Sistem memiliki fitur utama, termasuk login pengguna, *import dataset* melalui file *CSV* atau *scraping* otomatis menggunakan *script Python*. Data yang diperoleh kemudian diproses melalui tahap *preprocessing* sebelum dilakukan pelatihan model *machine learning*. Tampilan program khususnya halaman *dashboard* dan *input dataset* dapat dilihat pada Gambar 11 berikut:



**Gambar 11. Program halaman dashboard dan input dataset**

Terdapat 3 halaman untuk training metode *Naive Bayes*, *SVM*, dan *KNN*, Setelah *training* maka hasil prediksi ditampilkan dalam bentuk visualisasi grafik perbandingan metrik, serta tabel

*confusion matrix* untuk evaluasi performa setiap metode. Hasil tampilan program khususnya halaman training dan visualisasi dapat dilihat pada Gambar 12 berikut:



Gambar 12. Program halaman training dan visualisasi

## 5 Kesimpulan

Penelitian ini menganalisis sentimen kepuasan masyarakat terhadap aplikasi "INFO BMKG" di Play Store menggunakan algoritma *Naïve Bayes*, *SVM*, dan *KNN*. Hasil evaluasi menunjukkan bahwa *Naïve Bayes* memiliki performa terbaik dengan akurasi tertinggi 79,84%, diikuti oleh *KNN* sebesar 74,35% dan *SVM* sebesar 72,26%. *Naïve Bayes* juga mencatat presisi dan *recall* tertinggi masing-masing sebesar 60% dan 58%, serta *runtime* tercepat sebesar 0,19 detik dibandingkan *KNN* (10,32 detik) dan *SVM* (611 detik). Validasi menggunakan AUC menunjukkan keunggulan *Naïve Bayes* dalam membedakan sentimen positif (0,798), negatif (0,544), dan netral (0,820), sementara *SVM* mencatatkan nilai AUC terendah di seluruh kategori. Dengan demikian, *Naïve Bayes* terbukti sebagai metode paling optimal dalam klasifikasi sentimen pada studi ini. Namun, penelitian ini memiliki keterbatasan dalam jumlah data dan variasi ulasan yang dapat memengaruhi generalisasi model. Oleh karena itu, disarankan bagi peneliti selanjutnya untuk menggunakan dataset yang lebih besar dan beragam serta mengeksplorasi algoritma lain atau teknik ensemble untuk memperoleh hasil yang lebih akurat dan stabil.

## Referensi

- [1] D. Darwis, N. Siskawati, And Z. Abidin, "Penerapan Algoritma Naive Bayes untuk Analisis Sentimen Review Data Twitter BMKG Nasional," *Jurnal Tekno Kompak*, Vol. 15, No. 1, Pp. 131–145, 2021.
- [2] H. Firda, P. Putra, N. R. Oktadini, P. E. Sevtyuni, And A. Meiriza, "Comparison of Rating-based and Inset Lexicon-based Labeling in Sentiment Analysis using SVM (Case Study: Gobiz Application Reviews on Google Play Store)," *Sistemasi: Jurnal Sistem Informasi*, Vol. 14, No. 2, Pp. 516–528, 2025.
- [3] S. R. Putri, M. Arifin, And S. Supriyono, "Public Sentiment Analysis of Nadiem Makarim as Minister of Education, Culture, Research, and Technology using Support Vector Machine (SVM)," *Sistemasi: Jurnal Sistem Informasi*, Vol. 14, No. 2, Pp. 826–834, 2025.
- [4] J. Sanjaya, B. Priyatna, S. S. Hilabi, And Others, "Analisis Sentimen terhadap Opini Proyek Kereta Cepat menggunakan Metode Naive Bayes Classifier," *Jurnal Fasilkom*, Vol. 14, No. 1, Pp. 263–270, 2024.
- [5] Rojakul, Sumardianto, And A. Wibowo, "Analisis Sentimen Popularitas Capres dan Pilpres pada Media Sosial Twitter: Perbandingan Algoritma SVM, KNN, dan Naive Bayes.," *Techno. Com*, Vol. 23, No. 2, 2024.
- [6] C. Huda And M. B. Yel, "Analisa Sentimen tentang Ibu Kota Nusantara (IKN) dengan menggunakan Algoritma K-Nearest Neighbors (KNN) dan Naive Bayes," *Jurnal Ilmu Komputer dan Sistem Informasi (Jikoms)*, Vol. 7, No. 1, Pp. 126–130, 2024.
- [7] M. F. El Firdaus, Nurfaizah, And Sarmini, "Analisis Sentimen Tokopedia pada Ulasan di Google Playstore menggunakan Algoritma Naive Bayes Classifier dan K-Nearest Neighbor," Universitas Amikom Purwokerto, 2022.
- [8] R. Firdaus, R. Al Hariri, And H. F. Amran, "Sentimen Analisis Masyarakat tentang Penetapan Hari Raya Idul Adha Tahun 2023 pada Video Youtube menggunakan Algoritma Random Forest dan Support Vector Machine," *Jurnal Fasilkom*, Vol. 14, No. 1, Pp. 278–285, 2024.
- [9] V. Artanti, M. Faisal, And F. Kurniawan, "Klasifikasi Cardiovascular Diseases menggunakan Algoritma K-Nearest Neighbors (KNN).," *Techno. Com*, Vol. 23, No. 2, 2024.

- [10] M. Y. Hidayatulloh, A. Sunanto, A. Armansyah, M. F. A. Gevin, And D. D. Saputra, "Optimasi Sentimen Analisis Informatif dan tidak Informatif dari Tweet di BMKG menggunakan Algoritma Naive Bayes dan Metode Teknik Pengambilan Sampel Minoritas Sintetis," *J-Sakti (Jurnal Sains Komputer Dan Informatika)*, Vol. 7, No. 1, Pp. 1–12, 2023.
- [11] K. D. Pratama, D. W. Brata, And W. Purnomo, "Analisis Sentimen Ulasan Pengguna Aplikasi Info BMKG pada Google Play Store di Indonesia," *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, Vol. 7, No. 4, Pp. 1826–1834, 2023.
- [12] I. M. K. Karo, J. A. K. Karo, Y. Yuniarto, H. Hariyanto, M. Falah, And M. Ginting, "Analisis Sentimen Ulasan Aplikasi Info BMKG di Google Play menggunakan Tf-Idf dan Support Vector Machine," *J. Inf. Syst. Res*, Vol. 4, No. 4, Pp. 1423–1430, 2023.
- [13] R. Aziz, T. M. Fahrudin, And W. S. J. Saputra, "Analisis Sentimen Kepuasan Pengguna Oyo di Playstore dengan Multinoial Naive Bayes dan Chi-Square," *Jurnal Fasilkom*, Vol. 14, No. 1, Pp. 166–175, 2024.
- [14] P. Setiaji, K. Adi, And B. Surarso, "Development of Classification Method for Determining Chicken Egg Quality using GLCM-CNN Method," *Ingenierie Des Systemes D'information*, Vol. 29, No. 2, Pp. 397–407, Apr. 2024, Doi: 10.18280/Isi.290201.
- [15] S. Suryanto And W. Andriyani, "Sentiment Analysis of X Platform on Viral'fufufafa'account Issue in Indonesia using SVM," *Ijccs (Indonesian Journal Of Computing And Cybernetics Systems)*, Vol. 19, No. 1, 2024.
- [16] N. Sari, M. Jazman, T. K. Ahsyar, S. Syaifullah, And A. Marsal, "Implementation of Naive Bayes and Support Vector Machine Classification Algorithms for Sentiment Analysis of Bilingual Cyberbullying on X Application," *Sistemasi: Jurnal Sistem Informasi*, Vol. 14, No. 1, Pp. 211–224, 2025.
- [17] G. P. Insany, I. L. Kharisma, And R. Rosmawati, "Penerapan Algoritma Random Forest untuk menganalisis Ulasan Aplikasi Spotify pada Google Play," *Edumatic: Jurnal Pendidikan Informatika*, Vol. 8, No. 2, Pp. 369–378, 2024.
- [18] K. F. Ardika And J. D. Santoso, "Sentiment Analysis on Android Applications using Mediapipe for Text Classification," *Sistemasi: Jurnal Sistem Informasi*, Vol. 14, No. 1, Pp. 444–454, 2025.
- [19] G. Tamami, W. A. Triyanto, And S. Muzid, "Sentiment Analysis Mobile JKN Reviews using Smote based LSTM," *Ijccs (Indonesian Journal Of Computing And Cybernetics Systems)*, Vol. 19, No. 1, Pp. 13–24, 2025.
- [20] P. G. Aryanti And I. Santoso, "Analisis Sentimen pada Twitter terhadap Mobil Listrik menggunakan Algoritma Naive Bayes," *Ikra-Ith Informatika: Jurnal Komputer Dan Informatika*, Vol. 7, No. 2, Pp. 133–137, 2023.
- [21] F. F. Mulyandani, P. Setiaji, And Supriyono, "Optimasi Efisiensi Operasional UD Wijoo dengan Sistem Informasi Penjualan Mebel menggunakan Metode Multiple Step," *Jekin-Jurnal Teknik Informatika*, Vol. 5, No. 1, Pp. 445–458, 2025.