

# Analisis Sentimen Ulasan Aplikasi GoTube Menggunakan *Naïve Bayes* Berbasis *Particle Swarm Optimization*

Maedelien Tiffany Kariesta Simatupang<sup>a1</sup>, I Putu Gede Hendra Suputra<sup>a2</sup>

<sup>a</sup>Program Studi Informatika, Fakultas Matematika dan Ilmu Pengetahuan Alam,  
Universitas Udayana  
Jalan Raya Kampus UNUD, Bukit Jimbaran, Kuta Selatan, Badung, Bali, Indonesia  
<sup>1</sup>adelsimatupang26@gmail.com  
<sup>2</sup>hendra.suputra@gmail.com

## Abstract

*This research employs the sentiment analysis of GoTube application reviews using Naïve Bayes based on Particle Swarm Optimization (PSO). The study focuses on addressing the challenge of efficiently managing and analyzing user comments in the development of the GoTube application. By implementing automated sentiment analysis using text mining techniques, developers can enhance user experience and save resources. The methodology involves data collection, preprocessing, feature extraction using TF-IDF, classification using Naïve Bayes, and evaluation with various parameters. Additionally, Particle Swarm Optimization is utilized for feature selection to enhance the performance of the Naïve Bayes Classifier. The study aims to contribute to the improvement of GoTube's service quality and user satisfaction.*

**Keywords:** Sentiment Analysis, Naïve Bayes, Particle Swarm Optimization, GoTube Application

## 1. Pendahuluan

Pada era di mana penggunaan internet semakin meluas dan konten video menjadi semakin populer, Media sosial telah menjadi elemen yang sangat penting dalam rutinitas sehari-hari penduduk Indonesia. Masyarakat menggunakan media sosial sebagai alat komunikasi yang beragam untuk berinteraksi dengan orang lain [1]. Munculah aplikasi GoTube salah satu cara berkomunikasi melalui video sebagai alternatif bebas iklan dari platform sejenis seperti YouTube. GoTube menawarkan pengalaman menonton yang bebas gangguan iklan, memungkinkan pengguna untuk menikmati konten video favorit tanpa gangguan yang mengganggu. Dengan fitur-fitur seperti mode malam, pemutaran *pop-up*, dan pembuatan daftar putar yang disesuaikan, GoTube menjadi pilihan utama bagi mereka yang mencari kenyamanan maksimal dalam menonton video secara online. Dalam pengembangan aplikasi GoTube, salah satu tantangan adalah mengelola dan menganalisis komentar pengguna secara efisien. Pengecekan secara manual kurang praktis karena membutuhkan banyak waktu dan sumber daya manusia. Untuk mengatasi hal ini, diperlukan solusi otomatis yang menggunakan teknik *text mining* dan analisis sentimen. Implementasi fitur ini akan memberikan manfaat bagi pengguna dalam mengeksplorasi komentar pengguna, sementara pengembang akan menghemat sumber daya dan meningkatkan kualitas layanan. Dengan fitur ini, GoTube akan menjadi pilihan utama bagi pengguna yang menghargai kualitas dan inovasi dalam menonton konten video online. Analisis sentimen adalah proses ekstraksi dan identifikasi opini atau sentimen dari teks, kemudian mengelompokkannya menjadi kategori sentimen positif atau negatif [2]. Pada penelitian sebelumnya menganalisis sentimen pada aplikasi Spotify, dengan judul "Peningkatan Layanan Aplikasi Spotify Menggunakan *Naïve Bayes*" [3]. Pada penelitian tersebut mendapatkan 61586 ulasan yang diperoleh dari *Kaggle*, dan menghasilkan akurasi sebesar 85%. Adapun penelitian dengan judul "Analisis Sentimen Transportasi Online Menggunakan *Support Vector Machine* Berbasis *Particle Swarm Optimization*" [4]. Penelitian tersebut mengkonfirmasi bahwa pendekatan SVM yang menggunakan optimasi berbasis PSO memberikan tingkat akurasi yang lebih baik. Jika hanya menggunakan metode SVM sebesar 95,46% sedangkan SVM-PSO mendapatkan akurasi sebesar

96,04%. Dalam penelitian ini, pemilihan *Naive Bayes* yang berbasis *Particle Swarm Optimization* (PSO) sebagai metode analisis sentimen di Gotube didasarkan pada hasil penelitian sebelumnya pada aplikasi lain. Penelitian menunjukkan bahwa metode ini dapat meningkatkan akurasi dibandingkan dengan pendekatan konvensional. Oleh karena itu, penggunaan *Naive Bayes* berbasis PSO diharapkan dapat meningkatkan kualitas layanan dan pengalaman pengguna di Gotube.

## 2. Metode Penelitian

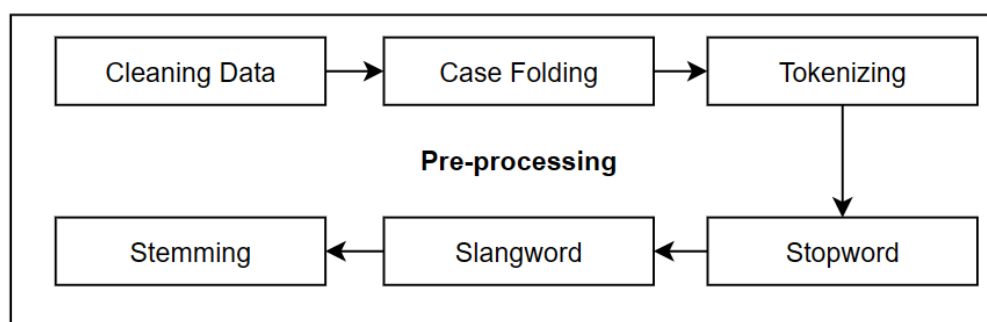
Pada tahap ini, akan diuraikan jalur penelitian yang akan diambil dalam analisis sentimen ulasan aplikasi GoTube menggunakan *Naive Bayes* berbasis *Particle Swarm Optimization* (PSO). Data yang dianalisis dalam penelitian ini terdiri dari evaluasi dan ulasan tentang aplikasi GoTube. Informasi akan dipisahkan menjadi dua segmen, yakni satu untuk proses pelatihan dan yang lainnya untuk proses pengujian, yang akan diklasifikasikan sebagai ulasan yang bersifat positif dan negatif. Output yang diharapkan adalah evaluasi hasil yang meliputi kinerja algoritma *Naive Bayes* tanpa melakukan seleksi fitur, dan kinerja algoritma *Naive Bayes* setelah dilakukan pemilihan fitur menggunakan PSO. Alur penelitian ini terdiri dari dua jalur yang berbeda: Pada jalur pertama, proses dimulai dengan pengumpulan data, diikuti oleh tahap preprocessing untuk membersihkan dan mempersiapkan data. Langkah berikutnya melibatkan ekstraksi fitur menggunakan metode TF-IDF untuk mengonversi teks menjadi representasi vektor. Setelah itu, dilakukan klasifikasi menggunakan algoritma *Naive Bayes*. Terakhir, model akan diuji menggunakan berbagai parameter untuk evaluasi performa., sedangkan jalur kedua menambahkan langkah seleksi fitur menggunakan *Particle Swarm Optimization* sebelum menjalankan proses klasifikasi dengan menggunakan algoritma *Naive Bayes*, dan kemudian juga dievaluasi menggunakan parameter yang sama.

### 2.1 Pengumpulan Data

Studi ini memanfaatkan dataset berupa ulasan aplikasi GoTube dari PlayStore. Ulasan diambil secara acak menggunakan metode web scraping dengan bantuan alat bawaan Google, yaitu Scraper, yang dijalankan menggunakan bahasa pemrograman Python. Data yang digunakan akan diberi label positif dan negatif berdasarkan rating pada ulasan produk.

### 2.2 Pre-Processing Data

Dalam kegiatan penambangan teks, Dokumen teks perlu melewati tahap persiapan terlebih dahulu sebelum dapat digunakan untuk analisis lebih lanjut. Tahapan ini dikenal sebagai *pre-processing* teks, dimana tujuan utamanya adalah mengkonversi data teks yang tidak terstruktur menjadi data yang terstruktur. Dengan *pre-processing*, dataset bisa dibersihkan dari gangguan, ketidakkonsistenan, sehingga proses pemrosesan dokumen lebih efisien. [5]. Tahapan preprocessing ini biasanya dilakukan secara berurutan, sesuai dengan gambaran yang diberikan dalam Gambar 1.



Gambar 1. Tahap Pre-Processing Data

**a. Cleaning Data**

Langkah ini dilakukan untuk menghilangkan data duplikat, nilai yang hilang, data yang tidak valid, atau noise dari data [6]. Selain itu, atribut-atribut yang tidak relevan untuk klasifikasi, seperti mention, hashtag, dan link, akan dihapus. Tahapan ini dibutuhkan untuk memastikan bahwa data yang digunakan dalam analisis adalah bersih dan relevan.. Proses ini penting karena data yang tidak terstruktur atau mengandung noise dapat mempengaruhi hasil analisis. Oleh karena itu, dengan membersihkan dan mempersiapkan data dengan baik, hasil klasifikasi dapat menjadi lebih akurat dan dapat diandalkan.

**b. Case Folding**

Langkah selanjutnya dikenal sebagai proses case folding, di mana seluruh karakter dalam teks diubah huruf kecil. Tujuannya adalah untuk menjaga konsistensi dalam pemrosesan teks, sehingga tidak ada perbedaan dalam analisis berdasarkan perbedaan huruf besar dan kecil.

**c. Tokenizing**

Langkah selanjutnya adalah tokenisasi yang bertujuan untuk memecah teks menjadi potongan-potongan kalimat atau kata-kata terpisah. Proses ini penting karena memungkinkan analisis lebih lanjut terhadap setiap unit teks secara terpisah, sehingga mempermudah pemrosesan dan pemahaman terhadap isi teks secara keseluruhan.

**d. Stopword**

Setelah tahap tokenizing, setiap kata berdiri sendiri. Namun, ada kata-kata seperti "ini", "itu", "adalah", yang tidak memberi makna penting dalam menentukan ciri dokumen. Kata-kata ini disebut stop word. Proses menghapus stop word disebut stopword removal, yang dilakukan untuk meningkatkan kualitas analisis dan mengurangi fitur yang tidak relevan dalam model.

**e. Slangword**

Langkah berikutnya adalah slangword, yang bertujuan untuk menggantikan kata-kata slang dengan bentuk yang sebenarnya. Ini dilakukan untuk memastikan konsistensi dan kejelasan dalam data, serta untuk menghindari ambiguitas yang dapat mempengaruhi analisis.

**Tabel 1.** Contoh Slangword

<b>Sebelum Slangword</b>	<b>Susudah Slangword</b>
bgss	bagus
mntp	mantap
benerin	benarkan
tdk	tidak

**f. Stemming**

Tahap terakhir dalam proses preprocessing adalah stemming, yang memiliki tujuan untuk menghapus sufiks, prefiks, dan konflik dari kata-kata dalam teks, mengubahnya menjadi bentuk dasarnya. Dalam Bahasa Indonesia, stemming penting sebelum *text mining* karena Bahasa Indonesia memiliki afiks yang membuat kata dasar bervariasi. Contohnya, sufiks seperti "-an", prefiks seperti "per-", dan konflik seperti "ke-...an". Ini membuat pencarian kata dasar sulit.

**2.3 Pembobotan Kata**

Pemberian bobot kepada setiap kata adalah bagian dari proses term weighting yang bertujuan untuk meningkatkan kemampuan analisis sentimen dalam kegiatan penambangan teks. [4]. TF-IDF, sebagai metode pembobotan term yang sering digunakan, kerap dipilih sebagai alternatif

yang umum. Dalam skema TF-IDF, bobot term dalam sebuah dokumen dihitung berdasarkan frekuensi kemunculan kata (Term Frequency, TF) dan inversenya dari frekuensi kemunculan kata tersebut di seluruh dokumen (Inverse Document Frequency, IDF). Dalam tahap TF, nilai kemunculan kata dalam dokumen dihitung, sementara pada tahap IDF, nilai kemunculan kata dalam seluruh dokumen dihitung. Nilai IDF berbanding terbalik dengan nilai TF, sehingga semakin sering suatu kata muncul dalam seluruh dokumen, semakin kecil nilai IDF-nya. Persamaan untuk menghitung TF-IDF adalah sebagai berikut: [6].

$$W_{dt} = TF_{dt} \times IDF_{ft} \tag{1}$$

Keterangan:

$W_{dt}$  = Bobot dokumen ke-d terhadap kata ke-t  
 $TF_{dt}$  = Banyaknya kata yang dicari pada sebuah dokumen  
 $IDF_{ft}$  = Inverse Document Frequency  $\log \frac{N}{df}$

## 2.4 Particle Swarm Optimization (PSO)

James Kennedy dan Russ Eberhart menemukan *Particle Swarm Optimization* (PSO) pada tahun 1995. PSO mengambil inspirasi dari pola perilaku sosial yang teramati pada kawanan burung yang terbang secara berkelompok atau gerombolan ikan yang berenang bersama-sama. PSO adalah metode penelitian populasi yang memperbaharui individu-individu populasi pada setiap iterasi. Dalam PSO, setiap partikel mengarah ke dua posisi, yakni posisi terbaik sebelumnya (pbest) dan posisi global terbaik (gbest), dengan tujuan mencapai solusi optimal. Metode ini diterapkan untuk meningkatkan ketepatan atribut dalam metode Naïve Bayes. [5]

$$v_i(t) = v_i(t-1) + c_1 r_1 (X_{pbesti} - X_{i(t)}) + c_2 r_2 (X_{gbest} - X_{i(t)}) \tag{2}$$

$$X_i(t) = X_i(t-1) + v_i(t) \tag{3}$$

Keterangan:

$v_i(t)$  = Kecepatan partikel  $i$  saat iterasi  $t$   
 $X_i(t)$  = Posisi partikel  $i$  saat iterasi  $t$   
 $c_1$  dan  $c_2$  = Learning rates untuk kemampuan individu dan pengaruh social  
 $r_1$  dan  $r_2$  = Bilangan random yang berdistribusi uniformal dalam interval 0 dan 1  
 $X_{pbesti}$  = Posisi terbaik partikel  
 $X_{gbest}$  = Posisi terbaik global

## 2.5 Naïve Bayes

*Naïve Bayes* ialah sebuah metode klasifikasi yang sederhana dan efisien, terutama pada klasifikasi teks, dan sering digunakan karena performanya yang baik. Namun, metode ini memiliki Salah satu kelemahan PSO yaitu kepekaannya terhadap pemilihan fitur. Jumlah fitur yang berlebihan dapat meningkatkan waktu komputasi dan bahkan mengurangi akurasi klasifikasi. Klasifikasi Bayes, yang menggunakan Teorema Bayes sebagai dasar, merupakan sebuah teknik yang dapat mengklasifikasikan data dengan ketepatan golongan tinggi dan efisiensi baik, terutama saat diterapkan pada dataset yang besar. Di bawah ini adalah formula dari Teorema Bayes [1].

$$P(H|X) = \frac{p(X|H) \times p(H)}{p(X)} \tag{4}$$

$P(H|X)$  = Probabilitas hipotesis H berdasar kondisi X (*posteriori probability*)  
 $P(X|H)$  = Probabilitas X berdasar kondisi pada hipotesis H  
 $P(H)$  = Probabilitas hipotesis H (*prior probability*)  
 $P(X)$  = Probabilitas dari X

## 2.6 Performance Evaluation Measure (PEM)

Penilaian Kinerja (PEM) adalah langkah yang digunakan untuk mengevaluasi kinerja suatu sistem. Validasi dijalankan dengan membagi dataset menjadi dua bagian: data pelatihan dan data pengujian, menggunakan metode `train_test_split`. Untuk mengukur akurasi, confusion matrix digunakan seperti yang dijelaskan dalam Tabel 2.

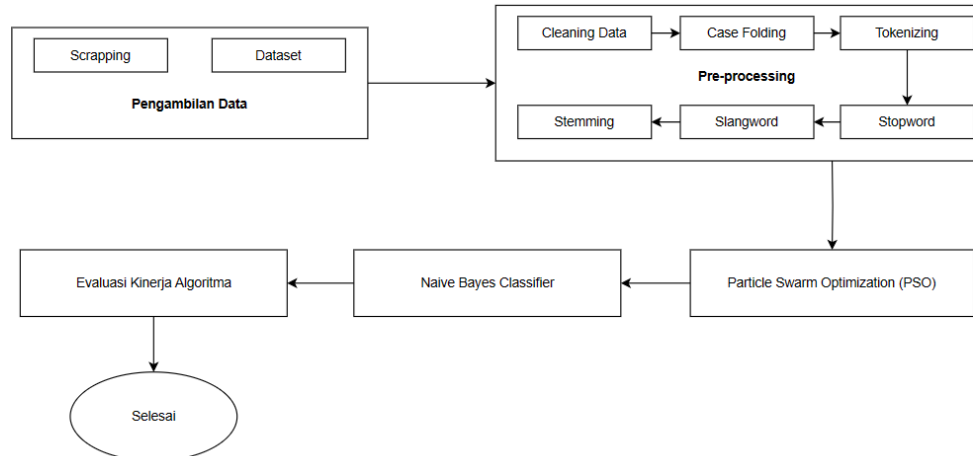
**Tabel 2. Confusion Matrix**

	Positif	Negatif
Positif	True Positif (TP)	False Negatif (FN)
Negatif	False Positif (FP)	True Negatif (TN)

$$\begin{aligned}
 \text{Akurasi} &= \frac{TP+TN}{TP+FP+TN+FN} \\
 \text{Precision} &= \frac{TP}{TP+FP} \\
 \text{Recall} &= \frac{TP}{TP+FN} \\
 \text{F1 - Score} &= 2 \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}
 \end{aligned}$$

## 3. Hasil dan Pembahasan

Alur penelitian ini dapat direpresentasikan dalam diagram alir, yang memberikan gambaran tentang tahapan yang akan dijalankan dari awal hingga selesai. Dalam diagram alir ini, setiap tahapan penelitian akan dijelaskan secara berurutan, memberikan pandangan umum tentang proses penelitian ini.



**Gambar 2.** Diagram Alir

### 3.1 Pengumpulan Data

Penelitian ini memanfaatkan jenis kualitatif yang berasal dari teks ulasan produk yang diselidiki. Data yang dianalisis dalam penelitian ini berdasarkan analisis ulasan aplikasi GoTube yang terdapat di platform *Google Play Store*. Melalui teknik web scraping, data yang dikumpulkan mencakup tanggal, nama pengguna, ulasan, dan rating. Total data ulasan yang berhasil diambil sebanyak 1.060 pada rentang waktu dari September 2023 hingga Mei 2024. Proses labeling dilakukan secara manual dengan membagi ulasan. Ulasan dikelompokkan menjadi dua kategori, yakni positif dan negatif, berdasarkan skala penilaian yang diberikan oleh pengguna, ulasan dengan rating 4-5 dikategorikan kelas positif, sedangkan rating 1-2 dikategorikan kelas negatif..

score	content	Label	
0	5	Bagus but ada bug kek tekeluar, pop up nya ila...	Positif
1	5	Gotube memang tidak mengecewakan mantap benar,...	Positif
2	2	Mengapa tampilan awal GoTube sekarang seperti ...	Negatif
3	5	Lebih asik dengerin music ,bebas iklan yg meng...	Positif
4	1	Aplikasi apa sih ini gak jelas banget katanya ...	Negatif
...	...	...	...
1055	1	Sangat kecewa sekali dengan perubahan gotube.....	Negatif
1056	5	Wow sangat mudah dan gampang.. Tidak ada kenda...	Positif
1057	1	Kenapa daftar putar video saya malah tidak mun...	Negatif
1058	1	Kenapa stiap putar lagu yg di pilih nanti nexn...	Negatif
1059	5	Bagus dan saya merasa terhibur serta memang ta...	Positif

1060 rows x 3 columns

**Gambar 3.** Ulasan GoTube

### 3.2 Pre-Processing

#### a. Cleaning Data

Proses ini melibatkan pembersihan data dari duplikat, nilai yang hilang, data tidak valid, atau noise, serta penghapusan atribut yang tidak relevan seperti mention, hashtag, dan link.

**Tabel 3.** Cleaning Data

Data Awal	Cleaning Data
Bagus banget kok, heran banget sama yang ga bilang bagus padahal pertama ak download ini apk tahun 2023 sampe skrg ga ad bedanya masih bagus bngt	Bagus banget kok heran banget sama yang ga bilang bagus padahal pertama ak download ini apk tahun sampe skrg ga ad bedanya masih bagus bngt

#### b. Case Folding

Selanjutnya merupakan *Case Folding*, yakni merubah huruf kapital ke huruf kecil.

**Tabel 4.** Case Folding

Cleaning Data	Case Folding
Bagus banget kok heran banget sama yang ga bilang bagus padahal pertama ak download ini apk tahun sampe skrg ga ad bedanya masih bagus bngt	bagus banget kok heran banget sama yang ga bilang bagus padahal pertama ak download ini apk tahun sampe skrg ga ad bedanya masih bagus bngt

#### c. Tokenizing

Berikutnya proses *Tokenizing* dimana membuat teks menjadi potongan kata.

**Tabel 5. Tokenizing**

<b>Case Folding</b>	<b>Tokenizing</b>
bagus banget kok heran banget sama yang ga bilang bagus padahal pertama ak download ini apk tahun sampe skrg ga ad bedanya masih bagus bngt	['bagus', 'banget', 'kok', 'heran', 'banget', 'sama', 'yang', 'ga', 'bilang', 'bagus', 'padahal', 'pertama', 'ak', 'download', 'ini', 'apk', 'tahun', 'sampe', 'skrg', 'ga', 'ad', 'bedanya', 'masih', 'bagus', 'bngt']

**d. Stopword**

Lalu proses *Stopword* yaitu menghilangkan kata yang tidak memiliki makna ataupun tidak berpengaruh pada hasil akurasi

**Tabel 6. Stopword**

<b>Tokenizing</b>	<b>Stopword</b>
['bagus', 'banget', 'kok', 'heran', 'banget', 'sama', 'yang', 'ga', 'bilang', 'bagus', 'padahal', 'pertama', 'ak', 'download', 'ini', 'apk', 'tahun', 'sampe', 'skrg', 'ga', 'ad', 'bedanya', 'masih', 'bagus', 'bngt']	['bagus', 'banget', 'heran', 'banget', 'ga', 'bilang', 'bagus', 'ak', 'download', 'apk', 'sampe', 'skrg', 'ga', 'ad', 'bedanya', 'bagus', 'bngt']

**e. Slangword**

Selanjutnya, terdapat tahap *slangword*, yang bertujuan mengubah kata-kata slang menjadi bentuk yang sebenarnya.

**Tabel 7. Slangword**

<b>Stopword</b>	<b>Slangword</b>
['bagus', 'banget', 'heran', 'banget', 'ga', 'bilang', 'bagus', 'ak', 'download', 'apk', 'sampe', 'skrg', 'ga', 'ad', 'bedanya', 'bagus', 'bngt']	['bagus', 'banget', 'heran', 'banget', 'enggak', 'bilang', 'bagus', 'aku', 'download', 'apk', 'sampai', 'sekarang', 'enggak', 'ada', 'bedanya', 'bagus', 'banget']

**f. Stemming**

Proses ini menstandarisasi kata-kata berimbuhan menjadi kata dasar sesuai dengan aturan tata Bahasa Indonesia.

**Tabel 8. Stemming**

<b>Slangword</b>	<b>Stemming</b>
['bagus', 'banget', 'heran', 'banget', 'enggak', 'bilang', 'bagus', 'aku', 'download', 'apk', 'sampai', 'sekarang', 'enggak', 'ada', 'bedanya', 'bagus', 'banget']	bagus banget heran banget enggak bilang bagus aku download apk sampai sekarang enggak ada beda bagus banget

**3.3 Splitting Data dan Pembobotan TF-IDF**

Setelah *pre-processing* selesai, langkah berikutnya adalah melakukan klasifikasi menggunakan metode Multinomial *Naive Bayes* dengan menggunakan library MultinomialNB. Data dibagi menjadi dua bagian, di mana 80% digunakan sebagai data latih dan 20% sebagai data uji. Sehingga, terdapat 848 data latih dan 212 data uji.

### 3.4 Performance Evaluation Measure (PEM)

Tabel 9 Data menunjukkan bahwa algoritma Naïve Bayes berhasil mencapai tingkat ketepatan sebesar 83,01%, sedangkan *Naïve Bayes* yang telah ditingkatkan dengan PSO berhasil mencapai tingkat ketepatan sebesar 85,37%. Dengan perbedaan 2,36% dalam akurasi, metode *Naïve Bayes-PSO* menunjukkan peningkatan yang signifikan dibandingkan dengan *Naïve Bayes* biasa. Penemuan ini mengindikasikan bahwa PSO dapat meningkatkan kinerja *Naïve Bayes*, karena PSO mampu mencari parameter terbaik dengan menyesuaikan diri terhadap *local best* serta memperhatikan posisi partikel terbaik secara global. Detail dari *confusion matrix* untuk kedua algoritma dapat ditemukan di Tabel 10 dan Tabel 11.

**Tabel 9.** Hasil pengujian data menggunakan *Naïve Bayes* dan *Naïve Bayes-PSO*

Algoritma	Accuracy
Naive Bayes	83,01%
Naive Bayes-PSO	85,375

**Tabel 10.** Presentase Hasil Pengujian Data Menggunakan *Naive Bayes*

	Precision	Recall	f1-score
Negatif	81%	75%	77%
Positif	84%	88%	86%
accuracy			83%
marco avg	82%	82%	82%
weighted avg	83%	83%	83%

**Tabel 11.** Presentase Hasil Pengujian Data Menggunakan *Naive Bayes-PSO*

	Precision	Recall	f1-score
Negatif	78%	83%	81%
Positif	90%	87%	88%
accuracy			85%
marco avg	84%	85%	84%
weighted avg	86%	85%	85%

## 4. Kesimpulan

Secara ringkas, penelitian ini sukses menerapkan analisis sentimen terhadap ulasan aplikasi GoTube memanfaatkan pendekatan Klasifikasi Naïve Bayes yang dikembangkan dengan menggunakan *Particle Swarm Optimization* (PSO). Hasil analisis menunjukkan bahwa penerapan Naïve Bayes dengan seleksi fitur PSO berhasil mencapai tingkat akurasi sebesar 85,37%, mengungguli Naïve Bayes standar yang hanya mencapai 83,01%. Peningkatan ini menegaskan keefektifan PSO dalam meningkatkan kinerja Naïve Bayes. Dengan demikian, peningkatan ini akan membantu memperbaiki pengalaman pengguna dan kualitas layanan dalam aplikasi GoTube.



### Daftar Pustaka

- [1] M. Christianto, J. Andjarwirawan, and A. Tjondrowiguno, "Aplikasi Analisa Sentimen Pada Komentar Berbahasa Indonesia Dalam Objek Video di Website YouTube Menggunakan Metode Naïve Bayes Classifier," 2020.
- [2] M. Hermiani *et al.*, "Analisis Ulasan Produk Menggunakan Metode *Naive Bayes Classifier*," *JNATIA*, vol. 2, no. 1, 2023.
- [3] E. Wahyuni, I. Ketut, G. Suhartana, and B. Jimbaran, "Peningkatan Layanan Aplikasi Spotify Menggunakan Naïve Bayes," 2022.
- [4] V. Kevin, S. Que: Analisis, S. Transportasi, A. Iriani, and H. D. Purnomo, "Analisis Sentimen Transportasi Online Menggunakan Support Vector Machine Berbasis *Particle Swarm Optimization* (Online Transportation Sentiment Analysis Using Support Vector Machine Based on Particle Swarm Optimization)," 2020. [Online]. Available: [www.tripadvisor.com](http://www.tripadvisor.com),
- [5] A. Santoso, A. Nugroho, and A. S. Sunge, "Analisis Sentimen Tentang Mobil Listrik Dengan Metode Support Vector Machine Dan Feature Selection Particle Swarm Optimization," 2022.
- [6] K. Sriwenda Putri, R. Setiawan, and A. Pambudi, "Analisis Sentimen Terhadap Brand Skincare Lokal Menggunakan Naïve Bayes Classifier," 2023.
- [7] M. Christianto, J. Andjarwirawan, and A. Tjondrowiguno, "Aplikasi Analisa Sentimen Pada Komentar Berbahasa Indonesia Dalam Objek Video di Website YouTube Menggunakan Metode Naïve Bayes Classifier," 2020.

Halaman ini sengaja dibiarkan kosong