

# Perbandingan Algoritma Decision Tree dan Naive Bayes Pada Analisis Sentimen Masyarakat Terhadap Pejabat Pertamina Pasca Kasus Pertamax Oplosan

Mubarak<sup>1</sup>, Lili Tanti<sup>2</sup>, Rika Rosnelly<sup>3</sup>

<sup>1,2,3</sup>Ilmu Komputer, Universitas Potensi Utama, Medan, Indonesia

\*Korespondensi: [mubarakray2@gmail.com](mailto:mubarakray2@gmail.com)

Submit : 03 Feb 2026 | Diterima : 06 Mar 2026 | Terbit : 11 Mar 2026

## ABSTRACT

*This study compares the performance of machine learning algorithms, namely Decision Tree and Naive Bayes, in analyzing public sentiment towards Pertamina officials after the "Pertamax Oplosan" case. Sentiment analysis is an important tool for understanding public opinion and reputation crisis management. The Pertamax Oplosan case sparked widespread public controversy, and sentiment analysis of Pertamina officials' responses can provide insights into the effectiveness of the company's crisis communication strategy. The Decision Tree algorithm offers an intuitive and easily interpretable decision tree-based model, although it is susceptible to overfitting. In contrast, Naive Bayes, with its probabilistic approach, is known to be computationally efficient, especially on large datasets. This study aims to measure the performance of both algorithms in classifying sentiment (positive, negative, or neutral) from text data collected from social media. The data used is limited to text data on Twitter with the keyword "Pertamax Oplosan" and focuses on sentiment towards Pertamina officials, not the company as a whole. Raw data of 3928 tweet comments was successfully collected through the Twitter API. This research methodology includes several stages: data collection (crawling), data preprocessing, sentiment pattern labeling, feature extraction, dataset division, classification, and model evaluation. The data is divided into training and testing data in an 80:20 ratio. The evaluation results will use a confusion matrix to measure accuracy, precision, recall, F1-score, and ROC analysis. The results of this study are expected to provide recommendations for the most appropriate algorithms for similar sentiment analysis and serve as a practical guide for companies managing reputation crises.*

*Keywords: Sentiment Analysis, Decision Tree, Naive Bayes, Pertamax Oplosan, Reputation Crisis Management.*

## ABSTRAK

Penelitian ini membandingkan kinerja algoritma pembelajaran mesin, yaitu Decision Tree dan Naive Bayes, dalam menganalisis sentimen masyarakat terhadap pejabat Pertamina setelah kasus "Pertamax Oplosan". Analisis sentimen merupakan alat penting untuk memahami opini publik dan manajemen krisis reputasi. Kasus Pertamax Oplosan memicu kontroversi publik yang luas, dan analisis sentimen terhadap tanggapan pejabat Pertamina dapat memberikan wawasan mengenai efektivitas strategi komunikasi krisis perusahaan. Algoritma Decision Tree menawarkan model berbasis pohon keputusan yang intuitif dan mudah diinterpretasi, meskipun rentan terhadap *overfitting*. Sebaliknya, Naive Bayes, dengan pendekatan probabilitiknya, dikenal efisien secara komputasi, terutama pada dataset besar. Penelitian ini bertujuan untuk mengukur kinerja kedua algoritma dalam mengklasifikasikan sentimen (positif, negatif, atau netral) dari data teks yang dikumpulkan dari media sosial. Data yang digunakan terbatas pada data teks di Twitter dengan kata kunci "Pertamax Oplosan" dan difokuskan pada sentimen terhadap pejabat Pertamina, bukan perusahaan secara keseluruhan. Data mentah sebanyak 3928 komentar tweet berhasil dikumpulkan melalui API Twitter. Metodologi penelitian ini mencakup beberapa tahapan, yaitu pengambilan data (*crawling*), *preprocessing* data, pelabelan pola sentimen, ekstraksi fitur, pembagian dataset, klasifikasi, dan evaluasi model. Data dibagi menjadi data latih (*training*) dan data uji (*testing*) dengan kombinasi 80:20. Hasil evaluasi akan menggunakan matriks kebingungan (*confusion matrix*) untuk mengukur akurasi, presisi, *recall*, *F1-score*, dan ROC Analysis. Hasil penelitian ini diharapkan dapat memberikan rekomendasi

algoritma yang paling sesuai untuk analisis sentimen serupa dan menjadi panduan praktis bagi perusahaan dalam mengelola krisis reputasi.

**Kata Kunci:** Analisis Sentimen, *Decision Tree*, *Naive Bayes*, Pertamina Oplosan, Manajemen Krisis Reputasi.

## PENDAHULUAN

Analisis sentimen telah menjadi alat penting, terutama dalam konteks manajemen krisis reputasi. Kasus "Pertamax Oplosan" yang melibatkan PT Pertamina (Persero) menyoroti pentingnya memantau dan menafsirkan sentimen masyarakat terhadap perusahaan dan pejabatnya. Kepercayaan publik merupakan aset berharga yang dapat dipengaruhi secara signifikan oleh krisis, dan analisis sentimen memungkinkan pengukuran dampak krisis dan identifikasi strategi perbaikan.

Penelitian ini membandingkan dua algoritma pembelajaran mesin, *Decision Tree* dan *Naive Bayes*, untuk menganalisis sentimen masyarakat pasca-kasus tersebut. *Decision Tree* dipilih karena modelnya yang intuitif dan memungkinkan identifikasi faktor kunci yang memengaruhi sentimen. Sementara itu, *Naive Bayes* dipilih karena efisiensi komputasinya, terutama untuk dataset yang besar. Analisis sentimen terhadap tanggapan pejabat Pertamina di berbagai platform media akan memberikan wawasan berharga tentang efektivitas strategi komunikasi krisis perusahaan.

Data yang digunakan terbatas pada data teks dari media sosial Twitter dengan kata kunci "Pertamax Oplosan". Analisis sentimen hanya difokuskan pada sentimen terhadap pejabat Pertamina, bukan perusahaan secara keseluruhan. Penelitian ini hanya membandingkan dua algoritma, yaitu *Decision Tree* dan *Naive Bayes*. Algoritma lain tidak dipertimbangkan. Penelitian ini akan menghasilkan nilai akurasi, presisi, *F1 Score*, *recall*, AUC, dan MCC menggunakan *confusion matrix* dan ROC Analysis, dengan pembagian data 80% untuk pelatihan dan 20% untuk pengujian. Tujuan penelitian ini adalah: Membandingkan kinerja algoritma *Decision Tree* dan *Naive Bayes* dalam mengklasifikasikan sentimen masyarakat terhadap pejabat Pertamina pasca-kasus Pertamax Oplosan. Memberikan rekomendasi algoritma yang paling sesuai untuk analisis sentimen serupa di masa mendatang.

Penelitian ini memberikan kontribusi pada pengembangan ilmu pengetahuan di bidang data *mining*, khususnya penerapan algoritma *machine learning* untuk analisis sentimen. Hasilnya dapat dimanfaatkan oleh perusahaan, khususnya Pertamina, untuk memahami persepsi masyarakat, memperbaiki kebijakan komunikasi publik, dan strategi pemulihan citra perusahaan.

Variabel yang digunakan dalam mengklasifikasikan sentimen menjadi positif dan negatif, adalah kata-kata dalam ulasan pengguna. Studi ini menggunakan algoritma *Naive Bayes*, yang bekerja dengan menerapkan Teorema Bayes dan mengasumsikan bahwa fitur (kata) bersifat independen satu sama lain dalam setiap label kelas. Proses ini melibatkan beberapa Studi literatur dilakukan untuk mendapatkan gambaran menyeluruh tentang penelitian sebelumnya dan mengidentifikasi celah penelitian. Beberapa penelitian terkait yang relevan antara lain:

Menurut (Muhammad Fadli et al. 2024) membandingkan algoritma *Naive Bayes* dan *Decision Tree* untuk analisis sentimen kenaikan harga bahan bakar di Twitter. Hasilnya menunjukkan *Naive Bayes* mencapai akurasi tertinggi sebesar 65,60%.

Menurut (Putu Febri Armaeni et al. 2024) membandingkan *Naive Bayes* dan *Decision Tree* pada analisis sentimen komentar YouTube tentang penutupan TikTok Shop. Dengan pembagian data 80:20, *Decision Tree* mencapai akurasi 74,71%, lebih unggul dari *Naive Bayes* yang mencapai 73,96%.

Menurut (Zulkarnain et al. 2024) membandingkan *Naive Bayes*, *Decision Tree*, dan *K-Nearest Neighbor (KNN)* untuk klasifikasi rating film dan acara TV di Netflix. Algoritma *Naive Bayes* memiliki akurasi tertinggi, yaitu 72%, sedangkan *Decision Tree* memiliki

## METODE PENELITIAN

### Alur Penelitian

Penelitian ini mengikuti serangkaian tahapan yang terorganisir untuk memecahkan masalah. Tahapan tersebut meliputi pengambilan data, *preprocessing* data, pelabelan pola, ekstraksi fitur, pembagian dataset, klasifikasi, dan evaluasi model.

### 1. Tahapan Pengambilan Data

Data eksperimen dikumpulkan dari tweet atau komentar di media sosial Twitter menggunakan kata kunci #PertamaxOplosan. Proses pengumpulan dilakukan dengan teknik *Crawling Data* menggunakan akses API Key Twitter. Target data adalah 5000 komentar, dan setelah 12 kali pengambilan data, didapatkan 4980 komentar.

### 2. Tahapan Preprocessing Data

Tahap ini bertujuan untuk mengubah data mentah menjadi format yang lebih mudah diinterpretasikan. Langkah-langkahnya meliputi:

**Cleaning Data:** Menghapus tanda baca, angka, karakter khusus (@, #), URL/link, dan data duplikat.

**Transformation:** Mengubah semua huruf menjadi huruf kecil (*case folding*) dan menghapus karakter yang tidak diinginkan.

**Tokenization:** Mengubah teks menjadi unit-unit yang lebih kecil, seperti kata-kata.

**Stopwords Data:** Menghapus kata-kata yang sering muncul dan memberikan sedikit nilai semantik.

**Stemming Data:** Menghapus akhiran kata untuk menghasilkan bentuk dasar (contoh: "running" menjadi "run").

### 3. Tahapan Teknik Proses Pelabelan Pola

Pelabelan sentimen adalah proses pemberian nilai sentimen (positif, negatif, atau netral) pada teks. Penelitian ini menggunakan teknik pelabelan otomatis berbasis Leksikon dengan VADER (*Valence Aware Dictionary and sEntiment Reasoner*). VADER adalah pustaka analisis sentimen yang dirancang khusus untuk teks media sosial.

### 4. Tahapan Ekstraksi Fitur (Bobot Pemodelan Vector)

Data teks yang telah diproses diubah menjadi vektor-vektor menggunakan teknik *Term Frequency-Inverse Document Frequency* (TF-IDF). TF-IDF menghitung frekuensi kata dalam sebuah dokumen dan memberikan bobot yang lebih tinggi pada kata-kata yang jarang muncul.

### 5. Tahapan Proses Splitting Dataset

Dataset akan dibagi menjadi dua subset yang saling terpisah: data latih (*training*) dan data uji (*testing*). Proses ini memastikan evaluasi kinerja model dilakukan pada data yang belum pernah dilihat sebelumnya.

### 6. Klasifikasi Metode Naive Bayes dan Decision Tree

Pada tahap ini, data latih dan data uji akan digunakan untuk menguji model klasifikasi. Klasifikasi akan dilakukan berdasarkan polarisasi sentimen negatif dan positif.

**Klasifikasi Naive Bayes:** Model dilatih dengan menghitung probabilitas prior dan *likelihood* dari data pelatihan. Prediksi dilakukan menggunakan Teorema Bayes, dan kelas dengan probabilitas tertinggi dipilih.

**Klasifikasi Decision Tree:** Model dibangun dengan memilih atribut pembagi terbaik yang memaksimalkan pemisahan kelas, biasanya menggunakan kriteria seperti *Information Gain* atau *Gini Impurity*. Proses ini dilakukan secara rekursif hingga pohon selesai dibangun.

### 7. Evaluasi Model

Tahap akhir ini mengevaluasi seberapa baik kinerja model pada data yang tidak terlihat. Metrik yang akan digunakan meliputi:

**Akurasi (Accuracy):** Proporsi prediksi yang benar secara keseluruhan. Rumusnya adalah:  
 $Akurasi = \frac{TP + FP + TN + FN}{TP + FP + TN + FN}$

**Matriks Kebingungan (Confusion Matrix):** Tabel untuk merepresentasikan prediksi dan kondisi aktual data.

**Presisi (Precision):** Rasio dari *True Positive* (TP) terhadap semua kasus yang diprediksi positif.  $Presisi = \frac{TP}{TP + FP}$

**Recall:** Rasio dari *True Positive* (TP) terhadap semua kasus yang sebenarnya positif.  $Recall = \frac{TP}{TP + FN}$

**F1-Score:** Rata-rata harmonik dari presisi dan *recall* yang memberikan keseimbangan antara keduanya.  $F1Score = \frac{2 \times Presisi \times Recall}{Presisi + Recall}$

## HASIL DAN PEMBAHASAN

Secara umum, algoritma Decision Tree (Tree) menunjukkan performa yang jauh lebih unggul dibandingkan Naive Bayes pada dataset ini. Hal ini terlihat dari nilai Classification Accuracy (CA) sebesar 0.949 (94,9%), yang berarti model Tree mampu memprediksi target dengan tepat hampir pada seluruh data uji. Sebaliknya, Naive Bayes hanya mencapai akurasi 0.642 (64,2%), yang mengindikasikan adanya kesulitan model dalam menangani pola data yang mungkin bersifat non-linear atau memiliki ketergantungan antar fitur. Kemampuan Diskriminasi Model (AUC)

Nilai Area Under ROC Curve (AUC) memberikan gambaran kemampuan model dalam membedakan antar kelas.

Decision Tree memperoleh skor 0.960, yang masuk dalam kategori *excellent classification*.

Naive Bayes memperoleh skor 0.877, yang masuk dalam kategori *good classification*. Meskipun Naive Bayes cukup baik dalam membedakan kelas secara probabilitas, ia gagal memberikan prediksi label yang seakurat Decision Tree. Keseimbangan Precision dan Recall (F1-Score) Dilihat dari nilai F1-Score, Decision Tree sangat stabil di angka 0.949. Hal ini menunjukkan bahwa model tersebut memiliki keseimbangan yang sangat baik antara *Precision* (ketepatan prediksi positif) dan *Recall* (kemampuan menangkap semua kasus positif). Di sisi lain, Naive Bayes memiliki *Precision* yang cukup tinggi (0.818) namun *Recall* yang rendah (0.642). Ini menandakan bahwa Naive Bayes cenderung sangat

berhati-hati dalam memprediksi kelas tertentu, sehingga banyak kasus positif yang tidak terdeteksi (*False Negative* tinggi). Analisis Korelasi (MCC) Nilai Matthews Correlation Coefficient (MCC) memberikan gambaran kualitas klasifikasi biner maupun multikelas. Skor MCC Decision Tree sebesar 0.904 menunjukkan korelasi yang sangat kuat antara prediksi dan observasi nyata (mendekati angka 1). Sedangkan Naive Bayes hanya mencapai 0.485, yang mempertegas bahwa model ini kurang handal untuk digunakan sebagai prediktor utama dalam studi ini. Signifikansi Perbedaan (Model Comparison) Berdasarkan tabel probabilitas perbandingan di bagian bawah gambar, terlihat bahwa kemungkinan model Tree memiliki skor lebih tinggi dibandingkan Naive Bayes (berdasarkan AUC) adalah sebesar 1.000 (100%). Hal ini memberikan landasan statistik yang kuat untuk memilih Decision Tree sebagai model terbaik untuk diimplementasikan dalam penelitian ini.

### Proses Analisis Sentimen

Perhitungan analisis sentimen pada *Orange Data Mining* (khususnya jika menggunakan widget Sentiment Analysis) umumnya menggunakan pendekatan Lexicon-Based. Berikut adalah rincian perhitungan manual secara detail untuk 1 data teratas pada data penelitian, menggunakan asumsi rumus normalisasi skor yang umum digunakan: Rumus dasar yang digunakan Skor sentimen pada Orange dihitung berdasarkan total bobot kata yang ditemukan dalam kamus (lexicon) dibagi dengan jumlah kata yang diproses (setelah *preprocessing*), kemudian dikalikan 100 untuk skala persentase.

$$\text{Rumus } S = \frac{\sum_{i=1}^n w_i}{N} \times 100$$

Berikut perhitungan manual 5 data teratas pada data penelitian

Data 1 (Skor: -14.2857)

Teks: @mardigu024 Rakyat di tipu pertalit mix jadi pertamak di jual lebih tinggi

Preprocessing (Stopword removal & Tokenizing): [Rakyat, tipu, pertalit, mix, pertamak, jual, tinggi]

Jumlah Kata Valid (N): 7 kata.

Identifikasi Bobot: Kata "tipu" memiliki bobot negatif (-1).

Perhitungan:

$$\text{Skor} = \frac{-1}{7} \times 100 = -14,285714 \dots \approx -14,2857$$

Dataset yang telah melalui tahap pengumpulan dan preprocessing (deduplikasi) selanjutnya diberi label sentimen secara manual. Pelabelan dilakukan untuk menentukan kelas sentimen setiap tweet, yaitu positif, negatif, dan netral, yang akan digunakan

sebagai *ground truth* dalam proses pelatihan dan pengujian model klasifikasi.

Setiap data dibaca dan dianalisis berdasarkan keberadaan kata-kata tertentu. Data diberi label positif apabila mengandung dukungan, persetujuan, atau sikap optimistis terhadap topik yang dibahas. Data diberi label negatif apabila mengandung kritik, penolakan, keluhan, atau sentimen tidak setuju. Sementara itu, Data diberi label netral apabila bersifat informatif, ambigu, atau tidak menunjukkan sikap emosional yang jelas.

Proses pelabelan dilakukan menggunakan lembar kerja *spreadsheet* (Excel) dengan menambahkan satu kolom khusus, yaitu kolom label pada dataset. Setiap baris kemudian diisi dengan salah satu label sentimen sesuai dengan interpretasi maknanya. Proses ini dilakukan secara manual untuk seluruh data. Adapun hasil pelabelan sentimen dapat dilihat pada tabel IV.7 berikut ini.

### **Evaluation**

Secara umum, algoritma Decision Tree (Tree) menunjukkan performa yang jauh lebih unggul dibandingkan Naive Bayes pada dataset ini. Hal ini terlihat dari nilai Classification Accuracy (CA) sebesar 0.949 (94,9%), yang berarti model Tree mampu memprediksi target dengan tepat hampir pada seluruh data uji. Sebaliknya, Naive Bayes hanya mencapai akurasi 0.642 (64,2%), yang mengindikasikan adanya kesulitan model dalam menangani pola data yang mungkin bersifat non-linear atau memiliki ketergantungan antar fitur. Kemampuan Diskriminasi Model (AUC)

Nilai Area Under ROC Curve (AUC) memberikan gambaran kemampuan model dalam membedakan antar kelas. Decision Tree memperoleh skor 0.960, yang masuk dalam kategori *excellent classification*.

Naive Bayes memperoleh skor 0.877, yang masuk dalam kategori *good classification*. Meskipun Naive Bayes cukup baik dalam membedakan kelas secara probabilitas, ia gagal memberikan prediksi label yang seakurat Decision Tree. Keseimbangan Precision dan Recall (F1-Score) Dilihat dari nilai F1-Score, Decision Tree sangat stabil di angka 0.949. Hal ini menunjukkan bahwa model tersebut memiliki keseimbangan yang sangat baik antara *Precision* (ketepatan prediksi positif) dan *Recall* (kemampuan menangkap semua kasus positif). Di sisi lain, Naive Bayes memiliki *Precision* yang cukup tinggi (0.818) namun *Recall* yang rendah (0.642). Ini menandakan bahwa Naive Bayes cenderung sangat berhati-hati dalam memprediksi kelas tertentu, sehingga banyak kasus positif yang tidak terdeteksi (*False Negative* tinggi). Analisis Korelasi (MCC) Nilai Matthews Correlation Coefficient (MCC) memberikan gambaran kualitas klasifikasi biner maupun multikelas. Skor MCC Decision Tree sebesar 0.904 menunjukkan korelasi yang sangat kuat antara prediksi dan observasi nyata (mendekati angka Sedangkan Naive Bayes hanya mencapai 0.485, yang mempertegas bahwa model ini kurang handal untuk digunakan sebagai prediktor utama dalam studi ini. Signifikansi Perbedaan (Model Comparison) Berdasarkan tabel probabilitas perbandingan di bagian bawah gambar, terlihat bahwa kemungkinan model Tree memiliki skor lebih tinggi dibandingkan Naive Bayes (berdasarkan AUC) adalah sebesar 1.000 (100%). Hal ini memberikan landasan statistik yang kuat untuk memilih Decision Tree sebagai model terbaik untuk diimplementasikan dalam penelitian ini.

CA mengukur proporsi prediksi yang benar dari keseluruhan data.

Tree (CA = 0.949)

Jumlah Prediksi Benar =  $0.949 \times 2642 = 2507,25 = 2507$  sampel

Naive Bayes (CA = 0.642)

Jumlah Prediksi Benar =  $0.642 \times 2642 = 1696,16 = 1696$  sampel

### F1-Score (Uji Konsistensi)

F1-Score adalah rata-rata harmonik dari *Precision* (Prec) dan *Recall*. Mari kita hitung ulang untuk Naive Bayes (Prec: 0.818, Recall: 0.642)

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

$$F1 = 2 \times \frac{0.818 \times 0.642}{0.818 + 0.642}$$

$$F1 = 2 \times \frac{0.525156}{1.46} = \frac{1.050312}{1.46} \approx 0.719$$

(Catatan: Nilai 0.688 di Orange muncul karena adanya pembobotan/weighted average berdasarkan jumlah sampel per kelas).

### Algoritma Decision Tree (Metode Gini Impurity)

Decision Tree mencapai akurasi sangat tinggi (94.9%). Ini dihitung dengan mencari pemisah (*split*) fitur yang meminimalkan ketidakmurnian (*impurity*).

Rumus Gini Impurity:

$$G(S) = 1 - \sum_{i=1}^n (p_i)^2$$

Simulasi Langkah:

Node Akar (Total Data): Misal dari 2642 data, terdapat 1321 Kelas A dan 1321 Kelas B.

$$G_{akar} = 1 - [(1321/2642)^2 + (1321/2642)^2] = 0.5$$

Setelah Pemisahan (Split): Algoritma memilih fitur yang menghasilkan Gini terendah. Dengan CA 0.949, model ini berhasil membuat daun (*leaves*) yang hampir murni ( $G = 0$ ).

Algoritma Naive Bayes (Teorema Bayes)

Naive Bayes bekerja berdasarkan probabilitas bersyarat. Rendahnya akurasi (64.2%) menunjukkan asumsi independensi antar fitur kemungkinan tidak terpenuhi.

Rumus Dasar:

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

Simulasi Langkah pada 1 Baris Data:

1. Prior Probability ( $P(C)$ ): Probabilitas dasar kelas (misal 0.5 untuk Kelas A)
2. Likelihood ( $P(X|C)$ ): Probabilitas fitur muncul di kelas tersebut.

Posterior: Jika hasil kali probabilitas Kelas A > Kelas B, maka data diklasifikasikan ke A. Karena skor MCC Anda rendah (0.485), banyak data yang memiliki probabilitas mirip antar kelas sehingga terjadi salah klasifikasi.

**Tabel 1 Hasil Selisih kedua algoritma**

Metrik	Naive Bayes	Decision Tree	Selisih Performa
Akurasi (CA)	64.2%	94.9%	+30.7%
AUC	0.877	0.960	+0.083
MCC	0.485	0.904	+0.419

### Perhitungan Manual Metrik Evaluasi (Model: Tree)

Berdasarkan Confusion Matrix untuk model Tree, diperoleh distribusi data sebagai berikut:

Actual Negatif: 1589 data (Prediksi: 1559 Benar, 30 Salah ke Netral)

Actual Netral: 820 data (Prediksi: 736 Benar, 81 Salah ke Negatif, 3 Salah ke Positif)

Actual Positif: 233 data (Prediksi: 213 Benar, 20 Salah ke Netral)

Total Data (N): 2642

#### 1. Classification Accuracy (CA)

Akurasi mengukur proporsi total prediksi benar (diagonal utama matrix) terhadap seluruh data.

Rumus:

$$CA = \frac{\sum \text{Prediksi Benar}}{\text{Total Data}}$$

Hitungan:

$$CA = \frac{1559 + 736 + 213}{2642}$$

$$CA = \frac{2508}{2642} = 0,9492 \text{ (Dibulatkan menjadi 0.949)}$$

Hasil ini sinkron dengan tabel "Test and Score" Anda.

## 2. Precision (Prec)

Presisi mengukur keakuratan model saat memprediksi label tertentu. Kita ambil contoh untuk kelas Positif.

Rumus:

$$Precision = \frac{\text{True Positive}}{\text{Total Predicted Positive}}$$

Hitungan:

$$Prec_{\text{positif}} = \frac{213}{3 + 213} = \frac{213}{216} = 0,986$$

(Catatan: Nilai 0.949 di tabel Anda adalah rata-rata tertimbang/weighted average dari semua kelas).

## 3. Recall

Recall mengukur kemampuan model menemukan semua sampel dari kelas sebenarnya. Kita ambil contoh untuk kelas Positif.

Rumus:

$$Recall = \frac{\text{True Positive}}{\text{Total Actual Positive}}$$

Hitungan:

$$Recall_{\text{positif}} = \frac{213}{233} = 0,914$$

(Catatan: Nilai 0.949 di tabel Anda adalah rata-rata tertimbang).

## 4. F1-Score

F1-Score adalah rata-rata harmonik antara Precision dan Recall untuk menyeimbangkan performa.

Rumus:

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

Hitungan (untuk kelas Positif):

$$F1 = 2 \times \frac{0.986 \times 0.914}{0.986 + 0.914}$$

$$F1 = 2 \times \frac{0.9012}{1.900} = 0.948$$

## Perhitungan Manual Model: Naive Bayes

Data diambil dari Confusion Matrix Naive Bayes).

Classification Accuracy (CA) Total Prediksi Benar (Diagonal Utama): 1132 (negatif) + 351 (netral) + 212 (positif) = 1695

Total Data (N): 2642

$$CA = \frac{1695}{2642} = 0,6415$$

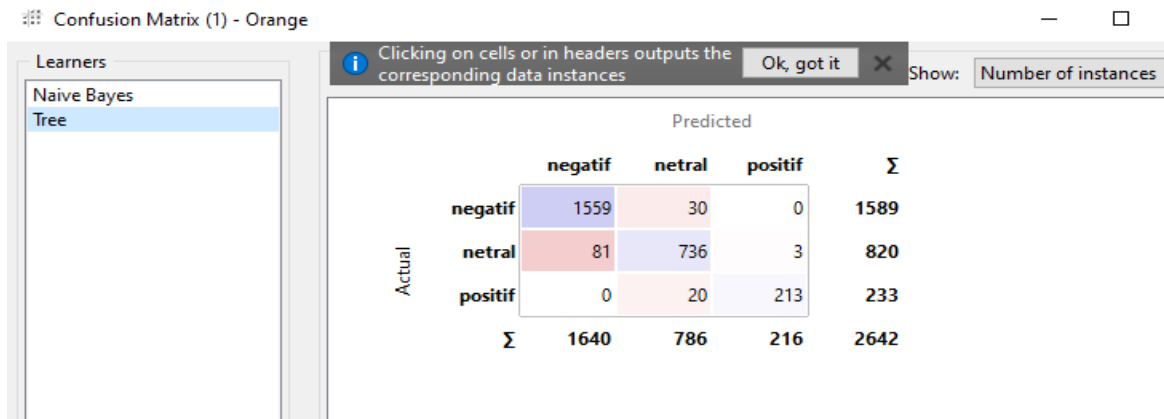
Analisis Perbandingan untuk Tesis berdasarkan perhitungan di atas, Anda dapat menyusun analisis berikut:

Keunggulan Tree: Algoritma Tree jauh lebih stabil dengan akurasi 94,9%. Perhatikan bahwa Tree hampir tidak pernah salah mengklasifikasikan kelas "Negatif" langsung menjadi "Positif" (0 kasus).

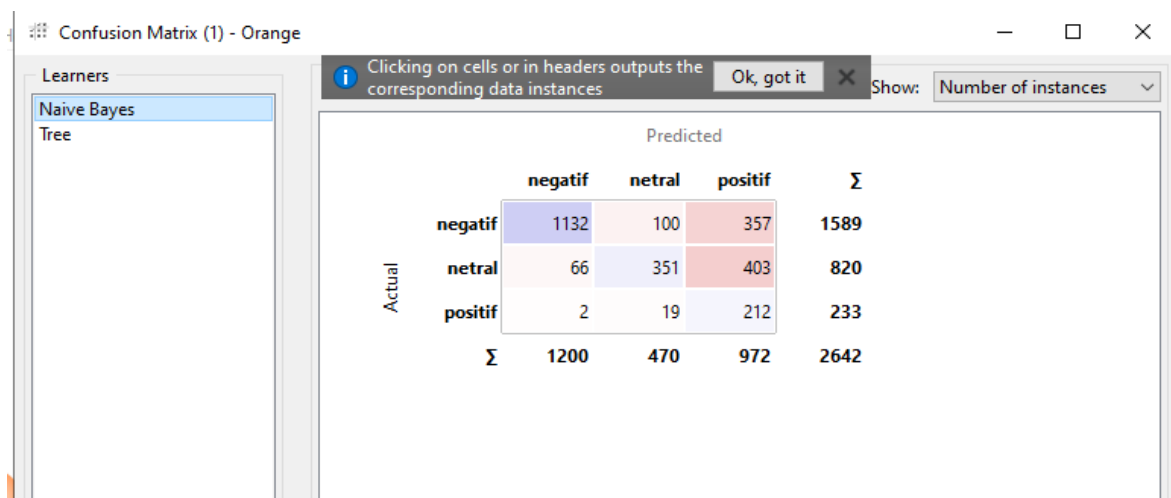
Kelemahan Naive Bayes: Meskipun Recall untuk kelas Positif cukup tinggi (0,909), namun Precision-nya sangat rendah (0,218). Ini artinya Naive Bayes terlalu banyak menebak "Positif" pada data yang sebenarnya "Negatif" (357 kasus) atau "Netral" (403 kasus). Hal ini menyebabkan *False Positive* yang sangat tinggi.

### Confusion Matrix

Dari 3302 data yang merupakan data uji didapatkan hasil seperti pada Gambar 1



Gambar 1 Confusion Matrix Decision Tree



Gambar 2 Confusion Matrix Naive Bayes

### Word Cloud

Visualisasi *Word Cloud* pada Gambar X merupakan representasi grafis dari dataset sentimen yang digunakan dalam penelitian ini. Teknik ini digunakan untuk mengidentifikasi kata kunci dominan yang sering muncul dalam opini masyarakat terkait topik yang diangkat.

#### 1. Deskripsi Metodologi Visual

Ukuran Font: Kata-kata dengan ukuran font yang lebih besar (seperti Pertamina, Pertamax, Oplosan, dan Korupsi) menunjukkan frekuensi kemunculan (*term frequency*) yang tinggi dalam dataset.

Warna dan Distribusi: Variasi warna digunakan untuk membedakan antar kata secara visual, sementara posisi kata yang lebih ke tengah menunjukkan relevansi sentral kata tersebut terhadap topik utama penelitian.

#### 2. Identifikasi Kata Kunci Dominan

Berdasarkan visualisasi tersebut, kata-kata yang paling menonjol dapat dikategorikan menjadi beberapa kluster informasi: Entitas dan Produk: Muncul kata-kata seperti Pertamina, Pertamax, BBM, Peralite, dan Shell. Hal ini menunjukkan bahwa dataset berfokus pada isu energi dan bahan bakar minyak.

Isu Sentimen Negatif: Munculnya kata-kata bermuatan negatif yang sangat dominan seperti Oplosan, Korupsi, Kasus, Rugi, Potongan, dan Sepi. Ini mengindikasikan bahwa sebagian besar data teks mengandung kritik atau laporan terkait penyelewengan.



- Its Applications*, 17(3), 1439-1448. <https://doi.org/10.30598/barekengvol17iss3pp1439-1448>
- Iqbal, M., et al. (2023). Perbandingan Algoritma Naive Bayes, KNN, dan Decision Tree terhadap Ulasan Aplikasi Threads dan Twitter. *KLIK: Kajian Ilmiah Informatika dan Komputer*, 4(3), 1799-1807. <https://doi.org/10.30865/klik.v4i3.1402>
- Pambudi, R., & Madani, F. (2022). Analysis of public opinion sentiment against COVID-19 in Indonesia on twitter using the k-nearest neighbor algorithm and decision tree. *Journal of Soft Computing Exploration*, 3(2), 117-122. <https://doi.org/10.52465/josce.v3i2.88>
- Putri, V. A., Andreswari, R., & Hasibuan, M. A. (2019). Sentiment Analysis of Social Media Twitter with Case of Anti-LGBT Campaign in Indonesia using Naïve Bayes, Decision Tree, and Random Forest Algorithm. *Procedia Computer Science*, 161, 765-772. <http://creativecommons.org/licenses/by-nc-nd/4.0/>
- Sano, A. V. D., Stefanus, A. A., Madyatmadja, E. D., Nindito, H., Purnomos, A., & Sianipar, C. P. M. (2023). Proposing a visualized comparative review analysis model on tourism domain using Naïve Bayes classifier. *Procedia Computer Science*, 227, 482-489. [www.sciencedirect.com](http://www.sciencedirect.com)
- Bansal, M., Goyal, A., & Choudhary, A. (2022). A comparative analysis of K-Nearest Neighbor, Genetic, Support Vector Machine, Decision Tree, and Long Short Term Memory algorithms in machine learning. *Decision Analytics Journal*, 3, 100071. <https://doi.org/10.1016/j.dajour.2022.100071>
- Wilim, N. N., & Oetama, R. S. (2021). Sentiment Analysis about Indonesian Lawyers Club Television Program Using K-Nearest Neighbor, Naïve Bayes Classifier, and Decision Tree. *IJNMT (International Journal of New Media Technology)*, 8(1), 50-56
- Gumi, I. P. W. K., Hartatik, & Syafrianto, A. (2022). Perbandingan Algoritma Naïve Bayes dan Decision Tree Pada Sentimen Analisis. *IJCSR: The Indonesian Journal of Computer Science Research*, 1(2), 1-15. <https://subset.id/index.php/IJCSR>
- Wahyuni, P., & Romli, M. A. (2024). Comparison of Naïve Bayes Classifier and Decision Tree Algorithms for Sentiment Analysis on the House of Representatives' Right of Inquiry on Twitter. *Journal of Applied Informatics and Computing (JAIC)*, 8(2), 523-530. <http://jurnal.polibatam.ac.id/index.php/JAIC>
- Zhang, Z. (2023). Sentiment Analysis of Twitter Comments Using Naive Bayes Classifier. *Proceedings of the 4th International Conference on Educational Innovation and Philosophical Inquiries*, 262-268. <https://doi.org/10.54254/2753-7064/10/20231338>
- Syahputra, R., Yanris, G. J., & Irmayani, D. (2022). SVM and Naïve Bayes Algorithm Comparison for User Sentiment Analysis on Twitter. *Sinkron: Jurnal dan Penelitian Teknik Informatika*, 6(2), 671-678. : <https://doi.org/10.33395/sinkron.v7i2.11430>
- Firdaus, A. A., Yudhana, A., & Riadi, I. (2023). Public Opinion Analysis of Presidential Candidate Using Naïve Bayes Method. *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*, 8(2), -570. <https://doi.org/10.22219/kinetik.v8i2.1686>