

Analisis Sentimen Pengguna TikTok tentang Pembangunan IKN Menggunakan Algoritma Naive Bayes dan Decision Tree

Andre Cahya¹, Sauqi Zamani², Oknel³, Yamin Nuryamin⁴ ,Ade Priyatna⁵

^{1,2,3,4,5}Program Studi S1 Teknologi Informasi, Universitas Bina Sarana Informatika

¹andrecahya22@gmail.com, ²sauqizamani2004@gmail.com*, ³oknel0987@gmail.com, ⁴yamin.yny@bsi.ac.id,

⁵ade.aeq@bsi.ac.id

Corresponding Author: Andre Cahya

ABSTRACT

The construction of the capital of the archipelago, or IKN, is again a hot chat in the Indonesian community, especially on social media like TikTok. Opinions of people vary, there are pros and there are cons. This study wanted to check how public sentiment about this project, as well as compare the performance of Naive Bayes algorithm and Decision Tree for text classification. We used a dataset from Kaggle titled "IKN sentiment analysis" in CSV format. We prepare the Data first through the stages of preprocessing, starting from making all small letters, cleaning the text, breaking it into words, removing common words that are not important, and reducing it to basic forms using literary libraries. The model was trained using TF-IDF techniques to convert text into vectors, with data divided 80% for exercises and 20% for tests. From the results, negative sentiment is more dominant, almost 52.1%, while the positive is only 46.2%. Naive Bayes algorithm got 84.04% accuracy, and it turned out to be better than Decision Tree. In essence, Naive Bayes is more suitable for sentiment analysis of Indonesian texts, especially about IKN development. This research can also help the development of ways of sentiment analysis in our language for important social problems in this country.

Keywords: Sentiment Analysis, Ibu Kota Nusantara, Naive Bayes, Decision Tree, TikTok, TF-IDF.

ABSTRAK

Pembangunan ibu kota nusantara atau IKN kembali menjadi perbincangan hangat di masyarakat Indonesia khususnya di media sosial seperti TikTok. Pendapat orang berbeda-beda, ada pro dan ada kontra. Penelitian ini ingin mengetahui bagaimana sentimen publik terhadap proyek ini, serta membandingkan kinerja algoritma Naive Bayes dan Pohon Keputusan untuk klasifikasi teks. Kami menggunakan kumpulan data dari Kaggle berjudul "analisis sentimen IKN" dalam format CSV. Kami menyiapkan Data terlebih dahulu melalui tahapan preprocessing, mulai dari membuat semua huruf kecil, membersihkan teks, memecahnya menjadi kata-kata, menghilangkan kata-kata umum yang tidak penting, dan mereduksinya menjadi bentuk-bentuk dasar menggunakan pustaka sastra. Model dilatih menggunakan teknik TF-IDF untuk mengubah teks menjadi vektor, dengan data dibagi 80% untuk latihan dan 20% untuk pengujian. Dari hasil tersebut, sentimen negatif lebih dominan, hampir 52,1%, sedangkan positif hanya 46,2%. Algoritma Naive Bayes mendapat akurasi 84,04%, dan ternyata lebih baik dari Decision Tree. Intinya, Naive Bayes lebih cocok untuk analisis sentimen teks bahasa Indonesia khususnya tentang perkembangan IKN. Penelitian ini juga dapat membantu pengembangan cara analisis sentimen dalam bahasa kita untuk masalah sosial penting di negara ini.

Kata Kunci: Analisis Sentimen, Ibu Kota Nusantara, Naive Bayes, Decision Tree, TikTok, TF-IDF.

1. Pendahuluan

Perkembangan teknologi informasi dan komunikasi di era digital telah mengubah pola interaksi masyarakat secara fundamental. Media sosial, yang awalnya hanya berfungsi sebagai sarana hiburan, kini telah bertransformasi menjadi arena publik bagi masyarakat untuk menyuarakan aspirasi terkait isu sosial, ekonomi, hingga politik [6]. Salah satu topik nasional yang menyita perhatian publik di berbagai platform, termasuk TikTok, adalah proyek pemindahan Ibu Kota Nusantara (IKN). Proyek ini dirancang pemerintah sebagai upaya



Lisensi

Lisensi Internasional Creative Commons Attribution-ShareAlike 4.0.

pemerataan pembangunan dan pengurangan beban infrastruktur di Jakarta. Namun, realisasinya IKN memicu polarisasi opini di tengah masyarakat. Sebagian pihak mendukung langkah ini sebagai strategi jangka panjang, sementara pihak lain mengkritisi urgensiya di tengah kondisi ekonomi yang menantang [1], [2]. TikTok, dengan basis pengguna yang masif di Indonesia, menjadi wadah strategis untuk memetakan sentimen publik. Melalui kolom komentar dan unggahan video, masyarakat secara bebas mengekspresikan pandangan mereka. Oleh karena itu, analisis terhadap komentar di TikTok dapat memberikan gambaran riil mengenai penerimaan masyarakat terhadap progres IKN. Penelitian ini menggunakan dataset sekunder yang bersumber dari Kaggle dengan judul “Analisis Sentimen Komentar TikTok Progres IKN” yang dipublikasikan oleh N. Rafifah pada tahun 2023. Data ini mencakup ribuan komentar yang telah dikategorikan ke dalam sentimen positif, negatif, dan netral [3]. Untuk mengolah data tekstual tersebut, penelitian ini menerapkan pendekatan data science melalui analisis sentimen. Metode ini merupakan bagian dari Natural Language Processing (NLP) yang bertujuan mengekstraksi dan mengklasifikasikan emosi dalam teks. Terdapat berbagai algoritma klasifikasi yang umum digunakan, seperti Support Vector Machine (SVM), Random Forest, Naive Bayes, dan Decision Tree [5]. Dalam studi ini, fokus perbandingan diarahkan pada dua algoritma, yaitu Naive Bayes dan Decision Tree, dikarenakan keduanya memiliki karakteristik yang kontras namun sering menghasilkan performa tinggi pada klasifikasi teks. Naive Bayes bekerja menggunakan prinsip probabilitas Teorema Bayes dengan asumsi bahwa setiap fitur bersifat independen. Metode ini dikenal efisien secara komputasi dan sering menunjukkan akurasi sempurna dalam klasifikasi sentimen sederhana [6], [9].

2. Tinjauan Pustaka

Penelitian mengenai analisis sentimen terhadap kebijakan publik dan kinerja pemerintah telah banyak dilakukan seiring dengan meningkatnya peran media sosial sebagai barometer opini masyarakat. Dalam konteks pemindahan Ibu Kota Nusantara (IKN), beberapa studi terdahulu telah mencoba memetakan respon publik menggunakan berbagai pendekatan algoritma.

Salah satu penelitian yang relevan dilakukan oleh Saputra, Wulandari, dan Lestari [1] yang menganalisis sentimen publik terhadap IKN. Studi ini membandingkan metode Support Vector Machine (SVM) dan Naive Bayes, di mana hasilnya menunjukkan bahwa opini masyarakat terpolarisasi cukup tajam. Sementara itu, fokus pada platform TikTok secara spesifik diangkat oleh Prasetyo dan Nugroho [2]. Mereka menyoroti bahwa TikTok memiliki karakteristik data yang unik dan menggunakan metode Random Forest untuk mengklasifikasikan komentar terkait progres IKN. Hasil penelitian mereka menunjukkan performa yang cukup baik, dengan akurasi mencapai 77%, yang mengindikasikan bahwa algoritma berbasis pohon keputusan (decision tree family) memiliki potensi besar dalam menangani data media sosial yang tidak terstruktur. Hal ini juga didukung oleh temuan Firmansyah dan Saputra [10] yang memanfaatkan Decision Tree untuk menganalisis sentimen pembangunan infrastruktur secara umum, di mana metode ini dinilai mampu memberikan interpretasi model yang jelas. Di sisi lain, perdebatan mengenai algoritma mana yang paling efektif untuk klasifikasi teks bahasa Indonesia masih terus berkembang. Siregar, Manurung, dan Harahap [5] melakukan studi komparatif langsung antara algoritma Naive Bayes dan Decision Tree untuk menilai sentimen kinerja pemerintah. Hasil studi mereka menjadi landasan penting bagi penelitian ini karena menunjukkan bahwa kedua metode tersebut memiliki keunggulan masing-masing tergantung pada karakteristik datasetnya. Temuan menarik dipaparkan oleh Ayudya et al. [7] dalam analisis sentimen sistem E-Tilang. Penelitian tersebut menemukan bahwa meskipun kedua algoritma bekerja sangat baik dengan akurasi di atas 95%, Decision Tree unggul tipis dengan



Lisensi

Lisensi Internasional Creative Commons Attribution-ShareAlike 4.0.

akurasi mencapai 99% pada pengujian tertentu, terutama ketika menggunakan pembobotan TF-IDF. Keunggulan Decision Tree juga dikonfirmasi oleh Pattiha dan Hanif [8] yang membandingkannya dengan K-Nearest Neighbor (K-NN) dan Naive Bayes pada data opini terkait PT PAL Indonesia. Mereka menyimpulkan bahwa struktur pohon keputusan mampu menangani variasi data opini dengan lebih presisi dibandingkan metode probabilitas sederhana. Namun, Naive Bayes tidak bisa dipandang sebelah mata. Alfi, Reynaldhi, dan Sibaroni [6] dalam penelitiannya mengenai ulasan film menemukan anomali positif di mana Naive Bayes justru mampu mencapai akurasi sempurna (100%) dan mengungguli metode hybrid lainnya pada kondisi data tertentu. Keandalan Naive Bayes dalam konteks program pemerintah juga diperkuat oleh penelitian Ramadhan dan Pratiwi [9] serta Widodo dan Nugraha [11], yang menilai bahwa metode ini sangat efisien secara komputasi dan efektif untuk klasifikasi teks berskala besar meskipun menggunakan asumsi independensi fitur yang sederhana. Berdasarkan paparan studi terdahulu di atas, terlihat adanya variasi hasil performa antara Naive Bayes dan Decision Tree tergantung pada domain topik dan karakteristik data. Meskipun dataset IKN dari Kaggle yang disusun oleh Rafifah [3] telah tersedia, belum banyak penelitian yang secara spesifik membandingkan performa Naive Bayes dan Decision Tree pada dataset tersebut untuk melihat konsistensi akurasi kedua algoritma ini dalam membedah isu IKN di platform TikTok. Oleh karena itu, penelitian ini hadir untuk mengisi celah tersebut dengan menguji efektivitas kedua metode ini secara head-to-head.

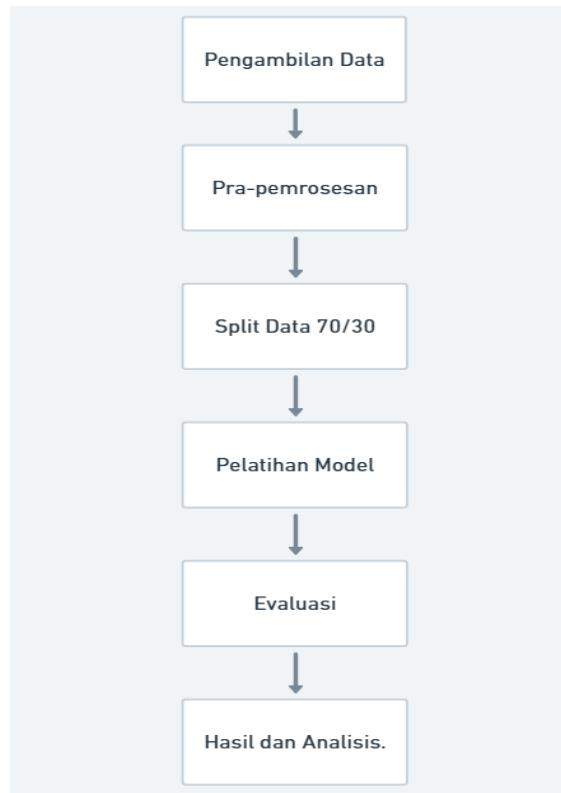
3. Bahan & Metode

Penelitian ini pakai pendekatan kuantitatif lewat metode text mining buat nganalisis sentimen komentar pengguna TikTok soal pembangunan Ibu Kota Nusantara (IKN). Tahapan metodologinya ngikutin beberapa penelitian sebelumnya yang relevan, kayak yang dilakukan Saputra et al. [1], Prasetyo dan Nugroho [2], sama Siregar et al. [5], yang pakai algoritma klasifikasi macam Naïve Bayes, Decision Tree, dan metode banding lainnya buat data opini publik dari media sosial. Data yang dipake di sini itu data publik, diambil dari dataset Kaggle judul “Analisis Sentimen Komentar TikTok Progres IKN” yang dibuat sama N. Rafifah [3]. Datasetnya dalam format CSV, isinya komentar pengguna TikTok tentang proyek IKN yang udah dilabeli manual jadi dua kategori sentimen: positif dan negatif. Tahapan penelitian ini terbagi menjadi beberapa tahap sebagaimana ditunjukkan pada Gambar 1 sebagai berikut.



Lisensi

Lisensi Internasional Creative Commons Attribution-ShareAlike 4.0.



Gambar 1. Tahapan Penelitian

3.1. Pengambilan Data

Dataset yang dipakai pada penelitian ini diambil dari platform Kaggle bertajuk “Data Komentar Tiktok Progres IKN” (<https://www.kaggle.com/datasets/najmarafifah/analisis-sentimen-komentar-tiktok-progres-ikn>). Dataset ini berisi 1471 data komentar dengan 2 atribut positif dan negatif.

Detail		Compact	Column
text	label		
engga ada namanya suatu pembangunan ga berguna selama niat nya baik buat masa depan negara itu sendiri...	positif		
aku yg dikalsel aja bangga deket ibukota wlp beda provinsi dan jarak yg jauh hehe	positif		
terlihat damai dan aman dari pendemo	positif		

Gambar 2. Sample dataset

```
# Load CSV
df_tweets = pd.read_csv('dataset_ikn.csv')

print(f"✅ Dataset berhasil dimuat!")
print(f"📊 Total data: {len(df_tweets)} baris")

# Tampilkan info kolom
print(f"\n📋 Kolom yang tersedia: {list(df_tweets.columns)}")
```

Gambar 3. Code memuat dataset



Lisensi

Lisensi Internasional Creative Commons Attribution-ShareAlike 4.0.

3.2. Lingkungan dan Alat Penelitian

Seluruh tahapan implementasi dan eksperimen dalam penelitian ini dilakukan menggunakan perangkat lunak Visual Studio Code (VSCode). Pemilihan Integrated Development Environment (IDE) ini didasarkan pada kinerjanya yang ringan, manajemen ekstensi yang fleksibel, serta kemampuannya dalam menangani pengembangan kode berbasis Python di lingkungan lokal secara efisien.

Untuk mendukung proses analisis sentimen mulai dari pra-pemrosesan teks hingga evaluasi model, penelitian ini memanfaatkan bahasa pemrograman Python dengan dukungan sejumlah pustaka (libraries) utama. Pustaka-pustaka tersebut dipilih berdasarkan fungsinya yang spesifik, antara lain:

1. Pandas dan NumPy Digunakan sebagai fondasi utama dalam memuat dataset.
2. NLTK dan Sastrawi Merupakan pustaka krusial untuk tahap text preprocessing.
3. NumPy untuk perhitungan numerik.
4. Matplotlib, Seaborn, dan WordCloud Ketiga pustaka ini digunakan untuk keperluan visualisasi data

```
import pandas as pd
import numpy as np
import re
import nltk
import Sastrawi
import warnings
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from Sastrawi.StemmerFactory import StemmerFactory
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.naive_bayes import MultinomialNB
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.svm import SVC
from sklearn.model_selection import GridSearchCV
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix, precision_recall_fscore_support
import matplotlib.pyplot as plt
from wordcloud import WordCloud
warnings.filterwarnings('ignore')
```

Gambar 4. Code Mengimpor Semua Library

3.3. Pra-pemrosesan Data

Tahapan pra-pemrosesan data bertujuan untuk mengubah data mentah yang tidak terstruktur menjadi format yang bersih dan siap dianalisis oleh model algoritma. Berdasarkan alur kerja sistem yang dibangun, proses ini meliputi langkah-langkah berikut: Pembersihan data, yaitu menghapus nilai kosong (missing values) dan duplikat.

1. Pembersihan Data Awal (Data Cleaning) Proses ini diawali dengan memvalidasi kualitas dataset, yaitu menghapus baris data yang memiliki nilai kosong (missing values) serta menyaring label sentimen agar hanya menyisakan kategori yang valid (positif dan negatif) untuk keperluan pemodelan biner.
2. Pembersihan Teks (Text Cleansing dan Case Folding) Setiap komentar atau tweet dibersihkan dari komponen yang tidak relevan (noise) menggunakan operasi Regular Expression (Regex). Langkah ini mencakup penghapusan URL, mention (@pengguna), tagar, angka, serta tanda baca. Selain itu, dilakukan penyeragaman format teks menjadi huruf kecil (lowercase) agar konsisten.
3. Tokenisasi dan Filtrasi (Stopword Removal) Teks dipecah menjadi potongan kata (token), kemudian dilakukan penyaringan untuk membuang kata-kata umum yang tidak memiliki makna sentimen signifikan (stopwords). Penelitian ini menggunakan daftar



Lisensi

Lisensi Internasional Creative Commons Attribution-ShareAlike 4.0.

stopwords standar bahasa Indonesia dari pustaka NLTK yang diperkaya dengan daftar kustom (seperti: "ikn", "yg", "dg") untuk meningkatkan relevansi konteks.

4. Stemming Mengubah setiap kata berimbahan menjadi kata dasarnya menggunakan pustaka Sastrawi. Proses ini penting untuk menyatukan variasi kata (misalnya: "membangun", "pembangunan", "dibangun") menjadi satu entitas fitur yang sama.
5. Ekstraksi Fitur (Feature Extraction) Mengubah data teks yang telah bersih menjadi representasi numerik menggunakan metode TF-IDF (Term Frequency-Inverse Document Frequency). Teknik ini memberikan bobot pada kata berdasarkan seberapa sering kata tersebut muncul dalam satu dokumen dibandingkan dengan kemunculannya di seluruh dataset, sehingga kata yang unik dan penting mendapatkan bobot lebih tinggi.

```
# Cek apakah ada missing values
print(f"\n❶ Checking missing values:")
print(df_tweets.isnull().sum())

# Hapus baris dengan missing values
df_tweets = df_tweets.dropna()
print(f"\n❷ Data setelah hapus missing values: {len(df_tweets)} baris")

# Standarisasi nama kolom
if 'comment' in df_tweets.columns and 'sentimen' in df_tweets.columns:
    df_tweets = df_tweets.rename(columns={'comment': 'tweet', 'sentimen': 'sentiment'})
elif 'text' in df_tweets.columns and 'label' in df_tweets.columns:
    df_tweets = df_tweets.rename(columns={'text': 'tweet', 'label': 'sentiment'})

# Standarisasi format sentimen (lowercase)
df_tweets['sentiment'] = df_tweets['sentiment'].str.lower()

# Tampilkan distribusi sentimen
print(f"\n❸ Distribusi Sentimen:")
print(df_tweets['sentiment'].value_counts())

# Cek apakah ada sentimen yang tidak valid
valid_sentiments = ['positif', 'negatif']
invalid_sentiments = df_tweets[~df_tweets['sentiment'].isin(valid_sentiments)]

if len(invalid_sentiments) > 0:
    print(f"\n❹ Ditemukan {len(invalid_sentiments)} data dengan sentimen tidak valid:")
    print(invalid_sentiments['sentiment'].value_counts())
    print("Akan dihapus dari dataset...")
    df_tweets = df_tweets[df_tweets['sentiment'].isin(valid_sentiments)]

print(f"\n❺ Data valid untuk analisis: {len(df_tweets)} baris")
```

Gambar 5. Code Data Diolah Oleh Algoritma Machine Learning

3.4. Pembagian Data (Data Splitting)

Untuk menguji generalisasi model, dataset dibagi menjadi dua subset independen menggunakan modul train_test_split dari pustaka Scikit-Learn. Proporsi pembagian yang diterapkan adalah 80% data latih (training set) untuk proses pembelajaran model, dan 20% data uji (testing set) untuk evaluasi performa. Pembagian ini menerapkan teknik stratified sampling guna memastikan proporsi sentimen positif dan negatif tetap seimbang di kedua subset, serta menggunakan parameter random state 42 untuk menjamin konsistensi hasil eksperimen.

```
# Encode label
df_model['sentiment_encoded'] = df_model['sentiment'].map({'positif': 1, 'negatif': 0})
df_model = df_model.dropna(subset=['sentiment_encoded'])
X = df_model['cleaned_tweet']
y = df_model['sentiment_encoded']
# Split data 80:20
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42, stratify=y)
print(f"\n❻ Data split (80:20):")
print(f" - Training: {len(X_train)} tweets ({len(X_train)/len(X)*100:.1f}%)")
print(f" - Testing: {len(X_test)} tweets ({len(X_test)/len(X)*100:.1f}%)")
```

Gambar 6. Code Dataset Dibagi Menjadi 80% Training Set Dan 20% Testing Set



Lisensi

Lisensi Internasional Creative Commons Attribution-ShareAlike 4.0.

3.5. Pelatihan Model

Proses pelatihan model dilakukan dengan menerapkan teknik Hyperparameter Tuning menggunakan metode GridSearchCV untuk mendapatkan performa yang optimal. Algoritma tidak sekadar dilatih menggunakan parameter default, melainkan diuji dengan berbagai kombinasi parameter (seperti nilai alpha pada Naive Bayes atau kedalaman pohon pada algoritma berbasis tree) untuk menemukan konfigurasi terbaik. Guna memvalidasi hasil pelatihan dan mencegah overfitting, diterapkan skema 3-Fold Cross-Validation, di mana data latih dibagi menjadi tiga bagian untuk pengujian iteratif selama proses pencarian model terbaik berlangsung.

```
models_to_tune = {
    'Naive Bayes': {
        'model': MultinomialNB(),
        'params': {
            'alpha': [0.1, 0.5, 1.0]
        }
    },
    'Decision Tree': {
        'model': DecisionTreeClassifier(random_state=42),
        'params': {
            'criterion': ['gini', 'entropy'],
            'max_depth': [None, 10, 20, 30]
        }
    }
}

trained_models = {}
print("Menjalankan GridSearchCV untuk setiap model...")

for name, config in models_to_tune.items():
    print("\n--- Tuning {} ---".format(name))
    grid_search = GridSearchCV(
        estimator=config['model'],
        param_grid=config['params'],
        cv=3,
        scoring='accuracy',
        n_jobs=-1,
        verbose=1
    )
    grid_search.fit(X_train_tfidf, y_train)
```

Gambar 7. Code Membangun Model terbaik

3.6. Evaluasi Model

Untuk mengukur kehandalan model dalam melakukan klasifikasi, penelitian ini menggunakan empat metrik evaluasi utama:



Lisensi

Lisensi Internasional Creative Commons Attribution-ShareAlike 4.0.

1. Akurasi (Accuracy), menghitung persentase ketepatan prediksi model secara keseluruhan terhadap data yang di uji.
2. Presisi (Precision), melihat seberapa akurat model saat menetapkan prediksi positif, untuk meminimalkan kesalahan saat deteksi.
3. Recall, mengukur daya tangkap model dalam mengenali keseluruhan data yang sebenarnya bersentimen positif.
4. F1-Score, nilai rata-rata harmonis yang menyeimbangkan presisi dan recall untuk memberikan gambaran performa yang lebih objektif.

```
y_pred_nb = nb_model.predict(X_test_tfidf)
y_pred_dt = dt_model.predict(X_test_tfidf)

print(f"\n\U0001f4c1 Prediksi model Naive Bayes:")
print(y_pred_nb)

print(f"\n\U0001f4c1 Prediksi model Decision Tree:")
print(y_pred_dt)
```

Gambar 8. Code Kinerja model diukur dengan beberapa metrik

3.7. Hasil dan Analisis

Tahap akhir penelitian berfokus pada analisis komparatif performa model yang telah dilatih. Skor akurasi, presisi, recall, dan F1-score disandingkan untuk menentukan algoritma mana yang paling andal dalam mengklasifikasikan sentimen komentar terkait IKN. Selain evaluasi metrik, analisis mendalam juga dilakukan terhadap fitur teks untuk memahami kata-kata apa yang menjadi penentu utama dalam prediksi sentimen.

```
feature_names = vectorizer.get_feature_names_out()
feature_importance = dt_model.feature_importances_

indices = np.argsort(feature_importance)[-20:][::-1]

for i, idx in enumerate(indices, 1):
    print(f" {i:2d}. {feature_names[idx]:25s} (importance: {feature_importance[idx]:.6f})")

plt.figure(figsize=(12, 8))
top_features = [(feature_names[i], feature_importance[i]) for i in indices[:15]]
words, importances = zip(*top_features)

plt.barh(range(len(words)), importances, color="#4CAF50", alpha=0.8)
plt.yticks(range(len(words)), words, fontsize=11)
plt.xlabel('Importance Score', fontsize=12, fontweight='bold')
```

Gambar 9. Code Visualisasi Feature Importance

4. Hasil

Penelitian ini mengimplementasikan dua algoritma klasifikasi, yaitu Decision Tree dan Naive Bayes, untuk memetakan sentimen publik terhadap pemindahan Ibu Kota Nusantara (IKN). Data eksperimen bersumber dari platform Kaggle yang telah melalui tahapan pra-pemrosesan, termasuk pembobotan kata menggunakan TF-IDF (Term Frequency-Inverse



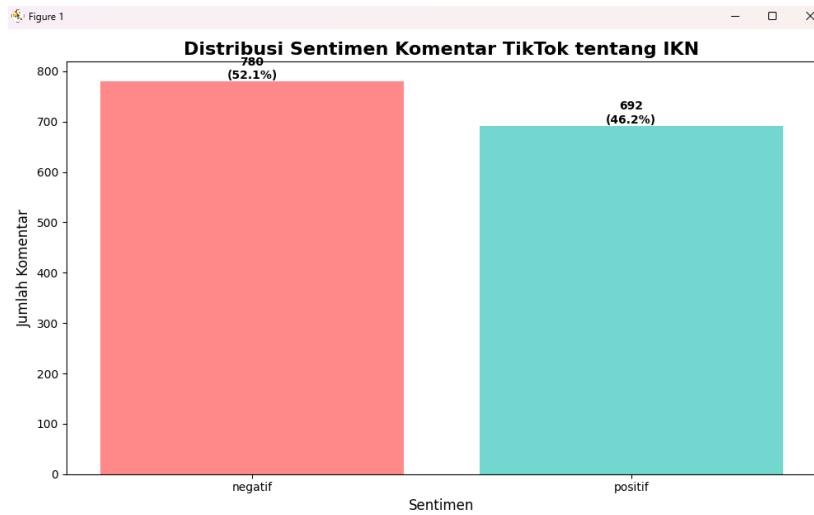
Lisensi

Lisensi Internasional Creative Commons Attribution-ShareAlike 4.0.

Document Frequency). Berbeda dengan penelitian sebelumnya yang mungkin menggunakan fitur numerik statis, penelitian ini mengolah fitur teks dinamis dari ribuan komentar pengguna.

Untuk menjamin validitas pengujian, dataset dibagi dengan proporsi 80% sebagai data latih dan 20% sebagai data uji. Seluruh proses komputasi dijalankan menggunakan lingkungan pengembangan Visual Studio Code (VSCode). Guna mendapatkan performa yang optimal, kedua model tidak hanya dijalankan dengan parameter standar, melainkan telah melalui proses optimasi (hyperparameter tuning) menggunakan GridSearchCV untuk menemukan konfigurasi terbaik.

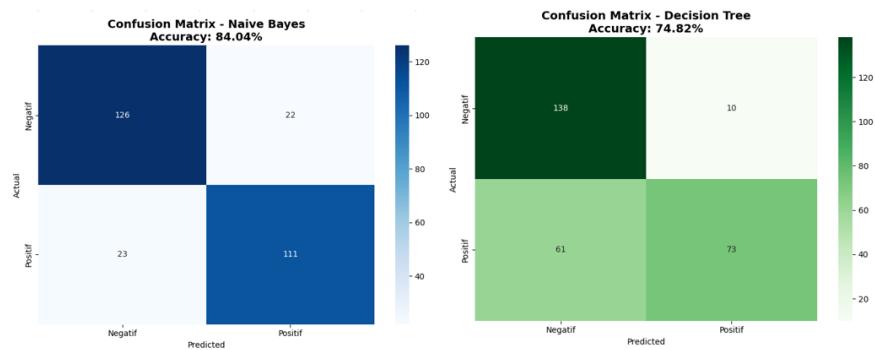
Hasil analisis menunjukkan bahwa sentimen negatif mendominasi dengan persentase 52,1%, sedangkan sentimen positif sebesar 46,2%. Model Naive Bayes mencapai akurasi sebesar 84,04% dengan keseimbangan antara precision dan recall. Decision Tree menghasilkan akurasi yang sedikit lebih rendah yaitu 74,82% namun lebih mudah diinterpretasikan. Word Cloud menunjukkan bahwa kata 'indonesia', 'bangga', dan 'jokowi' sering muncul dalam sentimen positif, ini bermakna Masyarakat menunjukkan kebanggaan dan harapan positif terhadap pembangunan IKN. sedangkan kata 'bangun', 'gak', dan 'pindah' sering muncul dalam sentimen negatif ini bermakna kekhawatiran tentang pemindahan ibu kota, proyek yang dianggap tidak prioritas, atau biaya besar.



Gambar 20. Distribusi sentimen tentang IKN

4.1. Analisis Confusion Matrix

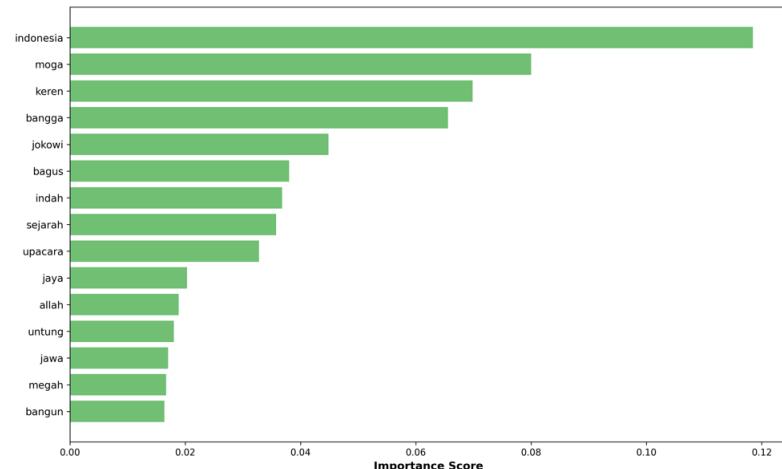
Berdasarkan visualisasi Confusion Matrix, terlihat jelas perbedaan pola prediksi antara kedua model. Naive Bayes tampil lebih unggul dengan akurasi 84,04% dan menunjukkan konsistensi yang tinggi. Model ini mampu memprediksi sentimen negatif (126 data) dan positif (111 data) dengan proporsi kesalahan yang relatif seimbang dan minim. Sebaliknya, Decision Tree (akurasi 74,82%) memperlihatkan ketimpangan yang cukup signifikan. Meskipun model ini sangat agresif dan akurat dalam mendeteksi sentimen negatif (mencapai 138 prediksi benar), ia justru "gagap" saat mengenali sentimen positif. Hal ini terbukti dari tingginya angka False Negative sebanyak 61 kasus, di mana komentar yang sebenarnya positif justru dianggap negatif oleh model. Kondisi ini menyimpulkan bahwa Naive Bayes jauh lebih stabil dan objektif dalam menangani variasi data dibandingkan Decision Tree yang cenderung bias ke kelas mayoritas (negatif).



Gambar 31. Grafik Perbandingan Akurasi Confusion Matrix Model Decision Tree dan Naive Bayes

4.2. Analisis Feature Importance

Analisis Feature Importance dilakukan untuk membedah variabel teks mana yang paling dominan mempengaruhi keputusan model dalam mengklasifikasikan sentimen. Berdasarkan hasil visualisasi, terlihat bahwa kata "indonesia" memiliki skor importance tertinggi, yang mengindikasikan bahwa konteks kebangsaan menjadi penentu utama dalam opini publik. Selain itu, deretan kata kunci teratas didominasi oleh kosakata bermuatan emosi positif dan harapan, seperti "moga", "keren", "bangga", dan "bagus". Kemunculan nama "jokowi" di posisi lima besar juga menegaskan bahwa figur kepemimpinan sangat erat kaitannya dengan sentimen yang terbangun. Hal ini menunjukkan bahwa model cenderung mendeteksi sentimen berdasarkan kata-kata sifat apresiatif dan aspirasi masyarakat terhadap kemajuan proyek IKN.



Gambar 42. Feature Importance Model Decision Tree dan Naive Bayes

5. Kesimpulan

Penelitian ini menyimpulkan bahwa algoritma Naive Bayes terbukti lebih efektif dibandingkan Decision Tree dalam memetakan sentimen publik terkait IKN di platform TikTok. Dengan capaian akurasi sebesar 84,04%, Naive Bayes mampu mengungguli Decision Tree yang tertinggal di angka 74,82%.

Keunggulan utama Naive Bayes terletak pada konsistensinya dalam memprediksi sentimen positif dan negatif secara seimbang. Sebaliknya, Decision Tree terbukti kurang stabil dan cenderung bias, seringkali gagal mengenali komentar positif dan menganggapnya sebagai negatif. Selain itu, analisis fitur menunjukkan bahwa opini masyarakat sebenarnya banyak



Lisensi

Lisensi Internasional Creative Commons Attribution-ShareAlike 4.0.

didorong oleh narasi optimisme dan kebanggaan nasional (seperti kata "moga", "keren", "bangga"), nuansa yang berhasil ditangkap dengan lebih baik oleh pendekatan probabilistik Naive Bayes. Oleh karena itu, metode ini sangat direkomendasikan untuk analisis data teks media sosial yang memiliki variasi bahasa tinggi.

REFERENSI

- [1] A. R. Saputra, D. Wulandari, dan M. F. Lestari, "Analisis Sentimen Publik terhadap Ibu Kota Nusantara (IKN) menggunakan Metode Support Vector Machine dan Naïve Bayes," Repository Universitas Putra Indonesia YPTK, Padang, Indonesia, 2023. [Online]. Available: <http://repository.upiypkt.ac.id/12625/>.
- [2] R. N. Prasetyo dan S. W. Nugroho, "Analisis Sentimen Pengguna TikTok tentang Progres Pembangunan IKN dengan Metode Random Forest," Jurnal Computer Science and Technology (JCSTech), Universitas Widya Dharma Klaten, vol. 6, no. 2, 2024. [Online]. Available: <https://journal.unwidha.ac.id/index.php/jcstech/article/view/345>.
- "Testing the Random Forest method on Google Colab showed an accuracy value of 77%, precision 78%, recall 77% and F1-score 77%. From these values, the Random Forest method is considered quite good in classifying Tiktok user sentiment in responding to the progress of the IKN relocation." (Rihastuti & Rosyidi, 2025, p. 22)
- [3] N. Rafifah, "Analisis Sentimen Komentar TikTok Progres IKN," Kaggle Dataset, 2023. [Online]. Available: <https://www.kaggle.com/datasets/najmarafifah/analisis-sentimen-komentar-tiktok-progres-ikn>
- "Dataset ini dikumpulkan dari dua unggahan TikTok yang membahas progres pembangunan Ibu Kota Negara (IKN) di Indonesia selama bulan Juni–Juli 2024. Dataset ini berisi 1.472 komentar yang telah diberi label sentimen positif, negatif, dan netral."
- [4] Komputika: Jurnal Sistem Komputer, "Vol. 14 No. 1 (2025): Komputika: Jurnal Sistem Komputer," Universitas Komputer Indonesia (UNIKOM), Bandung, Indonesia. [Online]. Available: <https://ojs.unikom.ac.id/index.php/komputika>.
- [5] S. A. Siregar, R. Manurung, dan F. Harahap, "Penerapan Algoritma Naïve Bayes dan Decision Tree untuk Analisis Sentimen Masyarakat Terhadap Kinerja Pemerintah di Media Sosial," Jurnal Informatika dan Sistem Informasi (JIFSI), vol. 10, no. 1, pp. 33-42, 2023. [Online]. Available: <https://jifsi.upbatam.ac.id/index.php/jifsi/article/view/812>.
- [6] M. Alfi, R. Reynaldhi, dan Y. Sibaroni, "Analisis Sentimen Review Film pada Twitter Menggunakan Metode Klasifikasi Hybrid SVM, Naïve Bayes, dan Decision Tree," e-Proceeding of Engineering, vol. 8, no. 5, pp. 10127-10137, 2021. [Online]. Available: <https://pdfs.semanticscholar.org/588d/d0dcc18cd5bef8f628c4f200dd9b359d8683.pdf>.
- "Pertumbuhan media sosial yang pesat memiliki pengaruh besar dan berperan penting dalam berbagai aspek kehidupan masyarakat di era digital."
- "Berdasarkan hasil pengujian menggunakan metode Naïve Bayes, diperoleh tingkat akurasi sebesar 100% yang menunjukkan metode ini mampu mengklasifikasikan sentimen dengan sangat baik."
- [7] S. Ayudya, A. Armand, M. Hafid, dan M. R. Muttaqin, "Analisis Sentimen Sistem E-Tilang pada Platform Twitter Menggunakan Metode Naïve Bayes dan Decision Tree," Jurnal Aplikasi Informatika dan Komputer (JAIC), vol. 8, no. 2, pp. 523-530, Nov. 2024. [Online]. Available: <https://jurnal.polibatam.ac.id/index.php/JAIC/article/view/8670>.
- "Hasil penelitian menunjukkan bahwa kedua algoritma tersebut memperlihatkan performa yang sangat baik dalam mengklasifikasikan sentimen, dengan rincian akurasinya 95% pada kedua dataset yang diuji."
- "Model ini menunjukkan akurasi sebesar 99% dengan pembobotan TF-IDF, yang mengindikasikan bahwa model Decision Tree mampu melakukan prediksi yang tepat terkait topik hak angket DPR."
- [8] F. S. Pattiha dan H. Hanif, "Perbandingan Metode K-NN, Naïve Bayes, Decision Tree untuk Analisis Sentimen Tweet Twitter Terkait Opini terhadap PT PAL Indonesia," Jurnal Riset Komputer, vol. 9, no. 2, pp. ..., 2022. [Online]. Available: <https://ulilalbabinstitute.id/index.php/JIM/article/download/4575/3597/9202>.



Lisensi

Lisensi Internasional Creative Commons Attribution-ShareAlike 4.0.

“Hasil penelitian menunjukkan bahwa algoritma Decision Tree memiliki tingkat akurasi tertinggi yaitu sebesar 99%, sehingga dapat disimpulkan bahwa penggunaan algoritma Decision Tree lebih baik dalam mengklasifikasikan data analisis sentimen komentar siswa.”

[9] A. F. Ramadhan dan D. Pratiwi, “Analisis Sentimen Masyarakat terhadap Program Pemerintah Menggunakan Algoritma Naïve Bayes,” *Jurnal Ilmiah Teknologi Informasi Terapan (JITTER)*, vol. 5, no. 3, pp. 212–219, 2023. [Online]. Available: <https://jitter.poltekegal.ac.id/index.php/jitter/article/view/574>.

[10] M. R. Firmansyah dan N. Saputra, “Analisis Sentimen pada Media Sosial Twitter terhadap Pembangunan Infrastruktur di Indonesia Menggunakan Metode Decision Tree,” *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)*, vol. 11, no. 4, pp. 715–724, 2024. [Online]. Available: <https://jtiik.ub.ac.id/index.php/jtiik/article/view/7685>.

[11] D. S. Widodo dan A. A. Nugraha, “Penerapan Text Mining dan Analisis Sentimen terhadap Opini Publik Menggunakan TF-IDF dan Naïve Bayes,” *Jurnal Sistem dan Teknologi Informasi (JUSTIN)*, vol. 10, no. 1, pp. 60–68, 2022. [Online]. Available: <https://journal.ubaya.ac.id/index.php/justin/article/view/5748>.



Lisensi

Lisensi Internasional Creative Commons Attribution-ShareAlike 4.0.