

Model Klasifikasi Naive Bayes untuk Pemetaan Persepsi Publik Secara Real-Time pada Media Sosial: Studi Kasus RUU TNI 2025

Reza Rahmatulloh^{1*}, Mukhammad Iqbal Ibrahim², Maya Rini Handayani¹, Khotibul Umam¹, Nur Cahyo Hendro Wibowo¹

¹Program Studi Teknologi Informasi, UIN Walisongo Semarang, Indonesia.

²Program Studi Hukum Keluarga Islam, UIN Walisongo Semarang, Indonesia.

Artikel Info

Kata Kunci:

Naïve Bayes;
RUU TNI 2025;
Sentimen Masyarakat;
TF-IDF.

Keywords:

Naïve Bayes;
2025 TNI Bill;
Public Sentiment;
TF-IDF.

Riwayat Artikel:

Submitted: 17 Mei 2025

Accepted: 17 Juli 2025

Published: 23 Juli 2025

Abstrak: Pengesahan Revisi Undang-Undang TNI 2025 memicu respons polarisasi di media sosial X (Twitter), terutama terkait kekhawatiran kebangkitan doktrin dwifungsi ABRI era Orde Baru. Penelitian ini bertujuan menganalisis sentimen publik, mengidentifikasi kecenderungan opini (dukungan/penolakan), dan mengevaluasi kinerja model klasifikasi. Data diperoleh melalui *crawling* 531 cuitan menggunakan kata kunci strategis ("dwifungsi ABRI", "demokrasi", "supremasi sipil"), diproses dengan *preprocessing* (*cleaning*, *case folding*, *tokenization*, *stopword removal*, *stemming*) dan pelabelan manual. Fitur diekstraksi menggunakan pembobotan TF-IDF, lalu diklasifikasikan dengan Naïve Bayes melalui lima skema pembagian data latih-uji (90%:10% hingga 50%:50%). Hasil menunjukkan dominasi sentimen negatif (81%/434 cuitan) dibanding positif (19%/97 cuitan), dengan model terbaik (skema 90%:10%) mencapai akurasi 85,19% dan presisi 100%, namun *recall* sangat rendah (4,35%) dan F1-score 8,33%. *Word cloud* mengungkap kata kunci negatif seperti "tolak", "DPR", dan "Dwifungsi ABRI" yang merefleksikan trauma historis. Simpulan penelitian mengindikasikan penolakan publik terhadap RUU TNI 2025, dengan kelemahan kritis pada deteksi sentimen positif akibat *class imbalance* dan keterbatasan Naïve Bayes menangkap konteks linguistik kompleks. Rekomendasi mencakup teknik *oversampling* (SMOTE) dan algoritma berbasis konteks (LSTM) untuk optimasi model.

Abstract: The ratification of the 2025 Indonesian National Armed Forces (TNI) Bill Revision triggered polarized responses on social media platform X (Twitter), primarily concerning fears of reviving the New Order-era Dwifungsi ABRI doctrine. This study aimed to analyze public sentiment, identify opinion trends (support/opposition), and evaluate classification model performance. Data were collected by *crawling* 531 tweets using strategic keywords ("dwifungsi ABRI", "democracy", "civil supremacy"), processed through *preprocessing* (*cleaning*, *case folding*, *tokenization*, *stopword removal*, *stemming*) and manual labeling. Features were extracted using TF-IDF weighting, then classified with Naïve Bayes across five train-test split schemes (90%:10% to 50%:50%). Results revealed dominant negative sentiment (81%/434 tweets) versus positive (19%/97 tweets), with the best model (90%:10% split) achieving 85.19% accuracy and 100% precision, but critically low recall (4.35%) and F1-score 8.33%. *Word cloud* analysis highlighted negative keywords (e.g., "reject", "parliament", "Dwifungsi ABRI") reflecting historical trauma. Findings indicate public rejection of the 2025 TNI Bill,

with critical weaknesses in detecting positive sentiment due to class imbalance and Naïve Bayes' limitations in capturing complex linguistic context. Recommendations include oversampling techniques (SMOTE) and context-sensitive algorithms (LSTM) for model optimization.

Corresponding Author:

Reza Rahmatulloh

Email: 2208096005@student.walisongo.ac.id

PENDAHULUAN

Pada 20 Maret 2025, Revisi Undang-Undang (RUU) TNI mengubah UU Nomor 34 Tahun 2004 disahkan oleh DPR RI dalam Rapat Paripurna ke-15 (Akashi, 2025). Pengesahan ini dilakukan secara resmi ketika DPR melaksanakan Rapat Paripurna ke-15 Masa Persidangan II Tahun Sidang 2024-2025 yang digelar di gedung DPR (Dirgantara, 2025). Pengesahan ini memicu debat publik di platform X (sebelumnya Twitter), dengan dua kutub utama: (1) dukungan atas perluasan peran TNI dalam pembangunan nasional, dan (2) kekhawatiran akan pengulangan Dwifungsi ABRI—konsep Orde Baru yang mengizinkan intervensi militer dalam ranah sipil (Putra & Siregar, 2023; Zhang dkk, 2021).

Dwifungsi ABRI, yang dikukuhkan melalui Seminar Angkatan Darat 1965 dan Resolusi MPRS 1966, menciptakan era represi di bawah otoritas militer (Anwar, 2020). Doktrin ini meski berkedok "pembangunan", justru membatasi kebebasan sipil melalui sensor media dan kebijakan darurat. Namun, penelitian ini tidak berfokus pada analisis historis, melainkan pada respons kontemporer di media sosial terhadap RUU TNI 2025 (Hasudungan, 2021). Pro dan Kontra terhadap Pengesahan RUU TNI, menghasilkan berbagai reaksi berlebihan. Sentimen merupakan pandangan yang didasarkan pada emosi yang muncul secara berlebihan terhadap sesuatu (Gunawan dkk., 2018).

X merupakan jejaring sosial dengan ciri khas *tweet* 140 karakter dan sistem *hashtag*, yang memfasilitasi ekspresi opini secara *real-time* (Bouazizi & Ohtsuki, 2019). Studi Adwan dkk, (2020), menunjukkan bahwa X menjadi pilihan ideal untuk analisis sentimen karena sifatnya yang terbuka dan viralitas konten. Data diambil menggunakan *data mining* dengan kata kunci seperti 'dwifungsi abri', 'demokrasi', dan 'hak asasi manusia', kemudian difilter berdasarkan indikator 'mendukung' dan 'menolak' (Ordila dkk., 2020). Data hasil pemrosesan dari *Data Mining* akan memberikan sejumlah informasi yang dapat dianalisis dalam jumlah yang besar sehingga data tersebut menghasilkan kesimpulan terkait suatu keputusan dimasa mendatang (Eko dkk., 2024).

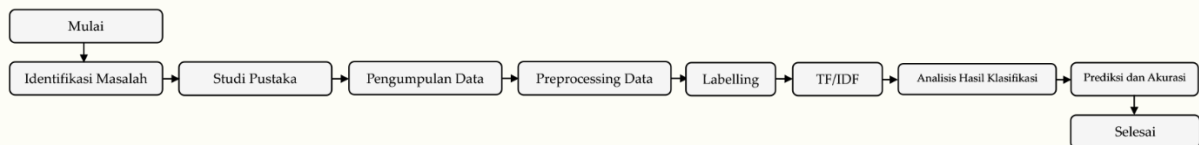
Naïve Bayes dipilih sebagai metode utama dalam penelitian ini karena efektivitasnya mengolah data teks singkat dan dinamis serta efisiensi komputasinya yang unggul pada dataset terbatas. Berbeda dengan *Logistic Regression* yang rentan *outlier* dan memerlukan asumsi linearitas, *Decision Tree/Random Forest* yang cenderung *overfitting* pada data *noisy* (seperti slang/emoji), *SVM* yang komputasinya berat untuk data berdimensi tinggi, *K-NN* yang tidak efisien untuk data besar, dan *Neural Networks* yang membutuhkan sumber daya intensif, Naïve Bayes mampu menangani fitur tinggi (kata kunci, *hashtag*) secara stabil, tahan terhadap fitur tidak relevan, dan memungkinkan iterasi berulang untuk peningkatan akurasi tanpa tuntutan komputasi besar—sesuai karakteristik *real-time* dan heterogen data media sosial.

Pemrosesan data menggunakan Naïve Bayes hingga lima kali guna meningkatkan akurasi sehingga menghasilkan nilai yang presisi. Penelitian berfokus guna menganalisis sentimen-sentimen publik terhadap pengesahan RUU TNI 2025 di Indonesia pada bulan Maret 2025. Penelitian ini bertujuan mengklasifikasi persepsi publik terkait RUU TNI 2025 di media sosial X guna memberikan wawasan mendalam tentang ragam reaksi masyarakat dan bahan refleksi kebijakan konstitusional pencegah konflik.

METODE

Penelitian ini dirancang sebagai studi prediktif terhadap sentimen masyarakat di media sosial X terkait RUU TNI dengan pendekatan kuantitatif. Data dikumpulkan melalui instrumen penelitian, diolah secara kuantitatif, dan dianalisis menggunakan klasifikasi Naïve Bayes untuk menghasilkan kategori sentimen (mendukung dan menolak) (Darman & Zila Razilu, 2024).

Dalam penelitian ini diterapkan teknik *text mining* untuk mengelompokkan sentimen publik pada unggahan Twitter ke dalam dua klasifikasi yaitu positif dan negatif. Data dikumpulkan melalui proses *crawling* tweet dengan kata kunci tertentu menggunakan Twitter API, di mana token autentikasi digunakan untuk mengakses *endpoint* pencarian. Seluruh hasil *crawling* kemudian dibersihkan, diproses, dan dimuat ke dalam basis data untuk keperluan analisis selanjutnya.



Gambar 1. Diagram Alir Metode Penelitian

Gambar 1 merupakan alur proses pengolahan data teks (*text mining*) secara sistematis, dimulai dari identifikasi masalah, studi pustaka tentang pengumpulan data (*crawling & scraping*), dilanjutkan dengan tahap *preprocessing data* yang mencakup enam langkah teknis (*cleaning, case folding, normalization, stopword removal, tokenization, dan stemming*), serta diakhiri dengan proses pelabelan menggunakan metode TF/DF dan analisis hasil klasifikasi untuk prediksi, pengukuran akurasi, dan penyelesaian proyek. Keseluruhan proses menunjukkan pendekatan komprehensif dalam transformasi data mentah menjadi informasi siap analisis untuk tujuan seperti klasifikasi atau analisis sentimen.

Identifikasi Masalah

Identifikasi Masalah ialah proses mengamati yang intensif sehingga ditemukan sebuah permasalahan yang terjadi dan merupakan permasalahan yang banyak dibahas (Artanto, 2024). Dalam tahapan ini, ditemukan adanya kegaduhan pada media sosial X mengenai keberadaan cuitan berisi sentimen maupun *argument* dari fenomena aktual yang sedang terjadi di Indonesia mengenai pengesahan RUU TNI yang sedang *trend* sehingga memicu reaksi publik yang beragam. Diperlukan penelitian yang menganalisa pola perasaan masyarakat Indonesia sehingga dapat diklasifikasi 'menerima' atau 'menolak' kebijakan tersebut.

Studi Pustaka

Literatur yang digunakan sebagai landasan memiliki parameter penelitian melingkupi judul sesuai dengan topik, database jurnal *scholar*, publikasi 5 tahun terakhir, dan indeks jurnal Sinta dan Scopus (Afuan dkk., 2025). Literatur lain yang digunakan demi menyusun penelitian adalah berita dari sumber terpercaya dan buku yang selaras mengenai *text mining*, Algoritma Naïve Bayes, dan Pengesahan RUU TNI 2025.

Pengumpulan Data

Pengumpulan data yang digunakan berbasis *text mining*, yang mana *text mining* adalah jenis analisis teks otomatis yang dilakukan oleh komputer untuk mengekstrak informasi berkualitas tinggi dari jenis teks yang tidak jelas dalam dokumen (Yunus dkk., 2023). Didalam *text mining* ini memerlukan 2 tahapan, yaitu *crawling* dan *scraping*. *Crawling* adalah proses yang digunakan untuk menghimpun dan menyusun konten dari internet, kemudiannya menyimpannya ke dalam basis *data search engine* (Zaman, 2024). Dengan menggunakan kata kunci 'RUU TNI', data cuitan dari aplikasi X yang relevan dengan kata kunci tersebut akan diambil (*Crawling*) dengan menggunakan bahasa pemrograman Node.js yang kemudian akan diindeks dan disebut sebagai daftar postingan. Selanjutnya dilaksanakan *scrapping* yaitu Ekstraksi data berupa informasi mengenai teks postingan, komentar, dan

respon pengguna dari daftar postingan yang sudah dihimpun sebelumnya (Setiawati dkk., 2024) menggunakan API Tweet Harvest versi 2.6.1 yang berjalan dibawah *framework* Node.js.

Preprocessing Data

Tahapan *Preprocessing* merupakan tahapan tingkat lanjut dimana data mentah yang dikumpulkan sebelumnya dipersiapkan dan diproses sehingga harus melalui beberapa prosedur sebelum dianggap layak dan siap digunakan. Prosedur sebagaimana disebutkan sebelumnya diantara lain:

- Cleaning* digunakan untuk membersihkan cuitan dari elemen-elemen yang tidak relevan dengan topik, seperti URL, emoji, tanda baca, dll (Pusean dkk., 2023).

```
[ ] sw_indo = stopwords.words('indonesian') + list(punctuation)

[ ] factory = StemmerFactory()
    stemmer = factory.create_stemmer()

def cleansing(text):
    text = emoji.demojize(str(text))
    text = text.lower()
    text = re.sub(r'^https?://.*[\r\n]*', '', text, flags=re.MULTILINE)
    word_list = word_tokenize(text)
    word_list = [word for word in word_list if word not in sw_indo]
    text = ' '.join(word_list)
    text = stemmer.stem(text)
    return text

[ ] df['cleaned_text'] = df['cleaned_text'].apply(cleansing)
    # Replace 'review' with 'cleaned_text' to clean the text in the column 'cleaned_text'

[ ] df.cleaned_text[3]

'guysss dampak ruu tni gambar yg kdm youth of may ya'

[ ] df.cleaned_text[100]

'kesah ruu tni demo puan bahas buka'
```

Gambar 2. Proses *Cleaning Data*

Gambar 2 menampilkan langkah untuk pemrosesan teks bahasa Indonesia yang tidak relevan dengan topik, seperti URL, emoji, tanda baca, dll. *Output*-nya menunjukkan hasil pembersihan teks yang masih mengandung beberapa kesalahan, seperti "poyess" dan "gumbor", mungkin akibat kesalahan penulisan atau stemming yang kurang akurat. Tujuannya adalah mempersiapkan data teks untuk analisis lebih lanjut.

- Case Folding* digunakan untuk memyamaratakan semua huruf di alfabet dari huruf 'a' hingga 'z' dalam data menjadi huruf kecil sebelum dilakukan pemrograman (Kurniadi Widodo dkk., 2022).

```
# Fungsi untuk menghitung skor sentimen
def calculate_sentiment(comment):
    # Bersihkan komentar (hapus tanda baca dan ubah ke lowercase)
    comment = re.sub(r'^[\w\s]', '', comment.lower())
```

Gambar 3. Proses *Case Folding*

Gambar 3 langkah untuk mendefinisikan fungsi `calculate_sentiment` untuk menghitung skor sentimen dari sebuah komentar. Langkah pertama adalah membersihkan teks dengan menghapus tanda baca menggunakan ekspresi reguler (`re.sub`) dan mengubah semua huruf menjadi *lowercase*, sehingga analisis sentimen menjadi lebih konsisten.

- Normalization* digunakan untuk mengubah dan menyesuaikan kata-kata yang ditulis dengan cara yang berbeda tetapi memiliki makna yang sama (Alamsyah dkk., 2023).

```

factory = StemmerFactory()
stemmer = factory.create_stemmer()

def cleansing(text):
    text = emoji.demojize(str(text))
    text = text.lower()
    text = re.sub(r'^https?://.*[\r\n]*', '', text, flags=re.MULTILINE)
    word_list = word_tokenize(text)
    word_list = [word for word in word_list if word not in sw_indo]
    text = ' '.join(word_list)
    text = stemmer.stem(text)
    return text

```

Gambar 4. Proses *Normalization*

Dalam Gambar 4 ini merupakan bagian mengubah teks mentah menjadi bentuk yang lebih seragam dan terstruktur untuk memudahkan analisis, misalnya "berlari" dan "lari" dianggap sama. Normalisasi seperti ini sangat krusial karena meningkatkan akurasi dalam analisis sentimen dan klasifikasi, dengan memastikan data konsisten serta bebas dari redundansi.

- d. *Stopword Removal/Filtering* digunakan untuk menghapus kata-kata yang intens muncul namun tidak penting dan tidak relevan (Aldiansyah Poetra dkk., 2023).

```

from nltk.tokenize import word_tokenize
from nltk.corpus import stopwords
import re, emoji
from string import punctuation
from sklearn.feature_extraction.text import TfidfTransformer, CountVectorizer
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory

[ ] sw_indo = stopwords.words('indonesian') + list(punctuation)

```

Gambar 5. Proses *Stopword Removal/Filtering*

Gambar 5 merupakan langkah untuk menghapus kata-kata yang intens muncul dalam cuitan, namun tidak relevan seperti "yang", "si", dan banyak kata kata lainnya yang tidak relevan dengan penelitian ini.

- e. *Tokenization* digunakan untuk memecah suatu kalimat menjadi kata-kata atau bagian yang lebih kecil yang disebut sebagai token. Token dapat berupa kata, frasa, ataupun tanda baca.

```

[ ] from nltk.tokenize import word_tokenize
from nltk.corpus import stopwords
import re, emoji
from string import punctuation
from sklearn.feature_extraction.text import TfidfTransformer, CountVectorizer
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory

[ ] sw_indo = stopwords.words('indonesian') + list(punctuation)

factory = StemmerFactory()
stemmer = factory.create_stemmer()

def cleansing(text):
    text = emoji.demojize(str(text))
    text = text.lower()
    text = re.sub(r'^https?://.*[\r\n]*', '', text, flags=re.MULTILINE)
    word_list = word_tokenize(text)
    word_list = [word for word in word_list if word not in sw_indo]
    text = ' '.join(word_list)
    text = stemmer.stem(text)
    return text

```

Gambar 6. Proses *Tokenization*

Gambar 6 merupakan langkah tokenization, penggunaan kode menggunakan `word_tokenize` dari NLTK untuk memecah teks menjadi kata-kata individual (`word_list`). Contohnya, kalimat "Saya setuju RUU TNI" akan diubah menjadi daftar ["Saya", "setuju", "RUU", "TNI"].

- f. *Stemming* digunakan untuk proses pembalikan kata-kata ke bentuk dasar atau bentuk kata dasar mereka (Imam Riadi dkk., 2024).

```
[ ] from nltk.tokenize import word_tokenize
    from nltk.corpus import stopwords
    import re, emoji
    from string import punctuation
    from sklearn.feature_extraction.text import TfidfTransformer, CountVectorizer
    from Sastrawi.Stemmer.StemmerFactory import StemmerFactory

sw_indo = stopwords.words('indonesian') + list(punctuation)

factory = StemmerFactory()
stemmer = factory.create_stemmer()

def cleansing(text):
    text = emoji.demojize(str(text))
    text = text.lower()
    text = re.sub(r'https?://.*[\r\n]*', '', text, flags=re.MULTILINE)
    word_list = word_tokenize(text)
    word_list = [word for word in word_list if word not in sw_indo]
    text = ' '.join(word_list)
    text = stemmer.stem(text)
    return text
```

Gambar 7. Proses *Stemming*

Pada gambar 7, langkah ini menggunakan kode *StemmerFactory* dari library *Sastrawi* untuk mengubah kata-kata menjadi bentuk dasarnya. Setelah teks dibersihkan dan di-tokenisasi, fungsi *stemmer.stem(text)* dipanggil untuk melakukan stemming pada setiap kata. Misalnya, kata "berlari" akan diubah menjadi "lari" dan "memakan" menjadi "makan".

Pelabelan

Pelabelan data digunakan untuk menetapkan kategori atau penanda spesifik untuk tiap unit data dalam kumpulan data, dengan maksud untuk mengidentifikasi atribut atau karakteristik tertentu yang hendak dievaluasi atau diteliti (Alvia Wirayani dkk., 2025).

```
import pandas as pd
import re

# Baca file CSV
df = pd.read_csv('cleaned_ruu_tni.csv')

# Daftar kata positif dan negatif (diperluas)
positive_words = [
    'dukung', 'setuju', 'pro', 'mendukung', 'baik', 'maju', 'kuat', 'damai', 'reformasi', 'demokrasi', 'transparansi', 'partisipasi publik', 'profesionalisme',
    'hak asasi manusia', 'supremasi sipil', 'aman', 'tepat waktu', 'responsif', 'memuaskan', 'recommended', 'profesional', 'sempurna'
]

negative_words = [
    'tolak', 'menolak', 'kontra', 'bahaya', 'ancaman', 'represif', 'arogan', 'kekerasan', 'militarisasi', 'dwifungsi abri', 'orde baru', 'pembungkaman',
    'korupsi', 'pelanggaran HAM', 'ketidakadilan', 'ketidaktransparanan', 'oligarki', 'penindasan', 'lama', 'buruk', 'jelek', 'telat', 'stuck', 'komplain', 'kecewa',
    'masalah', 'susah', 'parah', 'lelet', 'overload', 'tai', 'nyasar', 'malak', 'hilang', 'ngaret', 'bobrok', 'mengecewakan', 'gagal', 'ketipu', 'bot', 'ngambek',
    'payah', 'kacau', 'menyedihkan', 'menjengkelkan', 'menyebalkan', 'error', 'rusak', 'tertahan', 'terlambat', 'tidak jelas', 'tidak responsif', 'malas'
]
```

Gambar 8. Proses Pelabelan

Pelabelan secara umum akan mengkategorikan teks dan mengklasifikasikannya sebagai *positive* atau *negative* dengan mempertimbangkan keberadaan kata-kata kunci dan parameter lainnya.

TF-IDF

Term Frequency – Inverse Document Frequency (TF-IDF) digunakan untuk menyebutkan setiap kata yang muncul berdasarkan frekuensi penyebutannya, semakin sering kata tersebut muncul, maka semakin tinggi bobot atau nilai yang dia miliki (TF), sekaligus kata tersebut juga harus dikompromikan dengan umum atau tidaknya kata tersebut muncul (IDF)(Prastyo dkk., 2020).


```
[ ] # Inisialisasi TF-IDF Vectorizer
vectorizer = TfidfVectorizer()

# Assuming 'df' is your DataFrame and 'cleaned_text' is the column with text data
dokumen = df['cleaned_text'] # Assign the 'cleaned_text' column to the variable 'dokumen'

# Hitung TF-IDF
tfidf_matrix = vectorizer.fit_transform(dokumen)

# Cetak hasil
print(tfidf_matrix)
```

Gambar 9. Proses TF-IDF

Penggunaan TF-IDF memudahkan kita dalam pengukuran terhadap kepentingan kata dalam sebuah dokumen sehingga menyortir beberapa kata dan menyortirnya dengan membedakan kelas-kelas yang berbeda berdasarkan seberapa penting informasi yang terkandung didalamnya.

HASIL DAN PEMBAHASAN

Hasil dan pembahasan akan menjelaskan temuan terkait pengesahan RUU TNI 2025 dengan menggunakan algoritma Naïve Bayes. Metode ini digunakan untuk menganalisis data secara sistematis guna mengidentifikasi respon masyarakat Indonesia terkait pengesahan undang-undang tersebut.

Data

Seluruh data diperoleh melalui pengambilan data menggunakan API Tweet Harvest versi 2.6.1, yang dijalankan dengan *framework* Node.js. Proses ini menggunakan *authentication token* untuk otentikasi dan otorisasi dalam mengakses data cuitan dari aplikasi X.

The screenshot shows a Jupyter Notebook with the following code:

```
import pandas as pd

# Specify the path to your CSV file
file_path = "raw_tweet.csv" # Pastikan path file benar

# Read the CSV file into a pandas DataFrame
df = pd.read_csv(file_path, delimiter=',')

# Display the DataFrame
print(df) # Gunakan print() jika tidak di Jupyter Notebook
# atau
display(df) # jika di Jupyter Notebook
```

The output is a table with 5 columns: conversation_id_str, created_at, favorite_count, full_text, id_str, image_url, in_reply_to_screen_name, lang, location, quote_count, reply_count. The table contains 4 rows of tweet data.

| conversation_id_str | created_at | favorite_count | full_text | id_str | image_url | in_reply_to_screen_name | lang | location | quote_count | reply_count |
|---------------------|---------------------|----------------|---|---------------------|--|-------------------------|------|--------------------|-------------|-------------|
| 0 | 1902710754106822938 | 49739 | Demi sang omega white bisa menang anggar ma... | 1902710754106822938 | https://pbs.twimg.com/media/GmfJ8embaAA1FME.png | NaN | in | Under your bed | 583 | 496 |
| 1 | 1902967121790800120 | 11444 | Demo Tidak RUU TNI kemaren 20 Maret 2025 ada... | 1902967121790800120 | https://pbs.twimg.com/amp/ly_video_thumb/1902... | NaN | in | Jakarta, Indonesia | 67 | 86 |
| 2 | 1902562863132131362 | 4080 | Yang ditakutkan saat krtu TNI dan RUU TNI ka... | 1902562863132131362 | https://pbs.twimg.com/text_hr_video_thumb/19025... | NaN | in | NaN | 1481 | 603 |
| 3 | 1903083512212673012 | 25076 | gimana darpah ma di bu gembarempa | 1903083512212673012 | https://pbs.twimg.com/media/GmkoXvXUAAyQWZ.jpg | NaN | in | Rules 🇮🇩 | 734 | 434 |

Gambar 10. Proses Crawling Data X

Pada gambar 10 menampilkan hasil pengambilan data (*Crawling*) yang dilakukan dengan memanfaatkan API Tweet Harvest. Proses ini dijalankan pada *framework* berbasis Node.js, yang berfungsi untuk mengakses dan mengumpulkan data dari platform X. Data yang diperoleh melalui metode ini mencakup berbagai informasi terkait tweet, seperti teks, waktu unggahan, jumlah favorit, serta media yang disertakan dalam *tweet* tersebut.

Preprocessing

Pada tahap ini, untuk memastikan data bersih dan siap digunakan dilakukan *cleansing* awal dengan menghapus URL, *mention* (@username), *hashtag* (#), karakter khusus, dan angka, serta menyeragamkan teks menjadi huruf kecil (*case folding*) (Garcia & Berton, 2021). Selanjutnya, teks yang sudah dibersihkan diberi label sentimen (*positive* dan *negative*) berdasarkan daftar kata kunci, dan *tweet* yang terdeteksi sebagai iklan dihapus. Tahap berikutnya melibatkan tokenisasi (pemecahan teks menjadi kata per kata), penghapusan *stopword* (kata umum bahasa Indonesia dan tanda baca), serta *stemming* menggunakan Sastrawi untuk mengubah kata berimbuhan menjadi bentuk dasar.

a) **Pembersihan Data**

Dalam tahap ini, dilakukan eliminasi elemen-elemen yang tidak diperlukan dalam teks untuk memastikan data lebih terstruktur dan siap untuk analisis lebih lanjut.

Tabel 1. Hasil Pembersihan Data

| No. | Sebelum | Setelah |
|-----|--|---|
| 1. | "Demo Tolak RUU TNI kemaren 20 Maret 2025 akhirnya... #RUUTNI @user1 https://t.co/xxx" | "Demo Tolak RUU TNI kemaren Maret 2025 akhirnya" |

Pada Tabel 1 dilakukan dengan menghilangkan berbagai elemen tidak penting dalam teks tweet untuk mempersiapkan analisis sentimen yang lebih akurat. Beberapa tahap pembersihan data elemen-elemen seperti URL, *mention* (@username), hashtag (#), karakter khusus (tanda baca, emoji), dan angka dihilangkan karena tidak memberikan nilai signifikan dalam analisis sentimen.

b) **Case Folding**

Pada tahap ini, semua karakter dalam teks diubah menjadi huruf kecil (*lowercase*) untuk menselaraskan format teks.

Tabel 2. Case Folding

| No. | Sebelum | Setelah |
|-----|---|---|
| 1 | "Demo Tolak RUU TNI kemaren Maret 2025 akhirnya" | "demo tolak ruu tni kemaren maret 2025 akhirnya" |

Pada Tabel 2 mengubah huruf kapital menjadi huruf kecil untuk mempermudah analisis lebih lanjut dan mengidentifikasi kata selanjutnya.

c) **Tokenization**

Tokenization sangat penting dilakukan guna mengidentifikasi serta mengolah setiap kata secara terpisah.

Tabel 3. Tokenization

| | Sebelum | Setelah |
|---|---|--|
| 1 | "demo tolak ruu tni kemaren maret 2025 akhirnya" | ["demo", "tolak", "ruu", "tni", "maret", "2025" , "akhirnya"] |

Pada tahap *Tokenization* di mana setiap kata dan angka menjadi elemen terpisah. Hal ini bertujuan untuk memecah teks menjadi komponen terkecil (kata atau simbol) agar dapat diproses lebih lanjut.

d) **Stopword Removal**

Stopword removal membantu menyederhanakan teks dengan memfokuskan pada kata kunci yang relevan.

Tabel 4. Stopword Removal

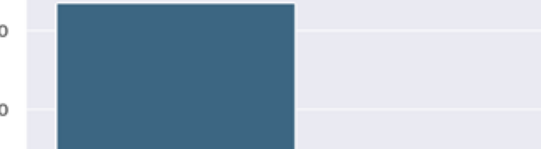
| No. | Sebelum | Setelah |
|-----|---|---|
| 1 | "demo tolak ruu tni kemaren maret 2025 akhirnya" | ["demo", "tolak", "ruu", "tni", "2025"] |

Pada Tabel 4 *Stopword removal* menggunakan daftar kata umum bahasa Indonesia (seperti "kemaren", "maret", "akhirnya") yang telah didefinisikan sebelumnya (misal dari `nlTK.corpus.stopwords` atau daftar custom yang dibuat sendiri).

Proses *stemming* mengubah kata berimbuhan menjadi bentuk dasarnya (*root word*) untuk menyamakan variasi kata yang memiliki makna serupa

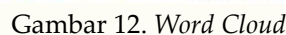
| No. | Sebelum | Setelah |
|-----|---|---|
| 1 | ["demo", "menolak", "ruu", "tni", "kebijakan", "pemerintah", "dipertanyakan", "dampaknya", "sangat", "berbahaya"] | ["demo", "tolak", "ruu", "tni", "bijak", "perintah", "tanya", "dampak", "sangat", "bahaya"] |

Pelabelan



| sentimen | count |
|----------|-------|
| negatif | 434 |
| positif | 97 |

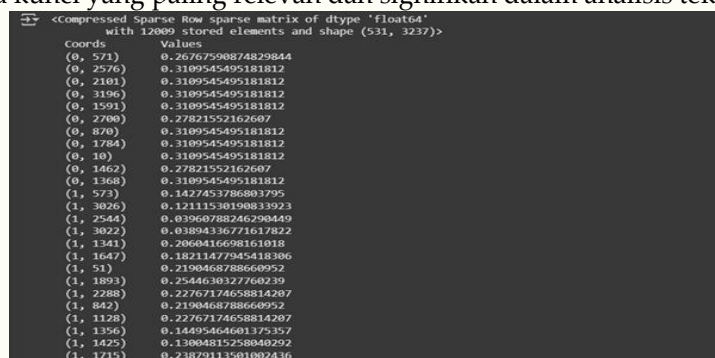
Berdasarkan diagram di atas, didapatkan bahwa sebanyak 434 komentar masyarakat memiliki sentimen *negative*, menunjukkan bahwa mayoritas masyarakat memiliki pandangan negative terkait pengesahan RUU TNI 2025. Sementara itu, hanya sebanyak 97 komentar ditemukan memiliki sentimen *positive*, menunjukkan bahwa sebagian kecil masyarakat memiliki pandangan yang lebih optimis atau mendukung pengesahan RUU TNI 2025. Hal ini mengindikasikan bahwa persepsi terhadap pengesahan RUU TNI 2025 cenderung kurang *positive* di kalangan masyarakat.



Visualisasi *word cloud* mengungkap pola linguistik yang merefleksikan dinamika sosial-politik masyarakat Indonesia. Dominasi kata "**tolak**", "**DPR**", dan "**undang-undang**" dalam sentimen negatif menunjukkan eratnya korelasi antara penolakan kebijakan dengan ketidakpercayaan terhadap institusi legislative (Putra & Siregar, 2023). Hal ini konsisten dengan aksi protes fisik di gedung DPR, membuktikan bahwa resistensi *online-offline* saling memperkuat. Sementara itu, kemunculan kata "**Dwifungsi ABRI**" dan "**Orde Baru**" dalam sentimen *negative* membuktikan trauma sejarah masih menjadi lensa utama masyarakat menilai kebijakan militer (Anwar, 2020). Masyarakat secara kolektif mengaitkan RUU TNI 2025 dengan praktik otoritarian masa lalu, di mana intervensi militer dalam sipil berujung pada represi. Di sisi lain, minimnya kata "**demokrasi**" dalam sentimen positif mengindikasikan dukungan terhadap kebijakan ini lebih bersifat pragmatis (misalnya harapan akan peran TNI dalam pembangunan infrastruktur) ketimbang didasari keyakinan terhadap nilai-nilai demokratis. Temuan ini menjadi *warning sign* bagi pemerintah bahwa legitimasi kebijakan tetap rapuh tanpa fondasi nilai demokrasi yang kokoh.

Pembobotan TF-IDF

Pembobotan TF-IDF (*Term Frequency-Inverse Document Frequency*) adalah metode statistik yang digunakan untuk menentukan tingkat pentingnya suatu kata dalam dokumen tertentu dengan mempertimbangkan frekuensi kemunculan kata tersebut dalam dokumen (TF) dan mengkompensasinya berdasarkan seberapa umum kata itu muncul di seluruh kumpulan dokumen (IDF) (Firdaus dkk., 2023), dimana kata yang sering muncul dalam dokumen tertentu namun jarang ditemukan di dokumen lain akan mendapatkan bobot lebih tinggi, sementara kata-kata umum yang muncul di banyak dokumen akan diberi bobot lebih rendah, sehingga teknik ini efektif untuk menyaring kata-kata kunci yang paling relevan dan signifikan dalam analisis teks.



| Coords | Values |
|-----------|---------------------|
| (0, 571) | 0.26767598874829844 |
| (0, 2576) | 0.3109545495181812 |
| (0, 2101) | 0.3109545495181812 |
| (0, 3196) | 0.3109545495181812 |
| (0, 1591) | 0.3109545495181812 |
| (0, 2780) | 0.27821552162607 |
| (0, 870) | 0.3109545495181812 |
| (0, 1784) | 0.3109545495181812 |
| (0, 10) | 0.3109545495181812 |
| (0, 1422) | 0.27821552162607 |
| (0, 1368) | 0.3109545495181812 |
| (1, 573) | 0.1427453786803795 |
| (1, 3026) | 0.12111530198833923 |
| (1, 2544) | 0.0396078824629849 |
| (1, 3022) | 0.03894136771617822 |
| (1, 1341) | 0.2060416698161018 |
| (1, 1647) | 0.18211477945418306 |
| (1, 51) | 0.2190468788660952 |
| (1, 1893) | 0.2544630327760239 |
| (1, 2288) | 0.22767174658814207 |
| (1, 842) | 0.2190468788660952 |
| (1, 1128) | 0.22767174658814207 |
| (1, 1356) | 0.14495464601375357 |
| (1, 1425) | 0.13004815258048292 |
| (1, 1715) | 0.23879113501002436 |

Gambar 13. Pembobotan TF-IDF

Dalam Gambar ini menunjukkan representasi matriks *sparse* dalam format *Compressed Sparse Row* (CSR) dengan tipe data *float 64*, berukuran 531 baris \times 3237 kolom dan menyimpan 12.009 elemen non-nol. Format CSR efisien untuk matriks yang didominasi nilai nol, dengan contoh nilai seperti (0, 571) = 0.267 atau (1, 573) = 0.142 yang merepresentasikan bobot atau hubungan tertentu. Matriks seperti ini sering digunakan dalam pemrosesan bahasa alami (misalnya TF-IDF), analisis data, atau komputasi ilmiah untuk menghemat memori dan optimasi operasi matriks. Elemen yang ditampilkan merupakan bagian kecil dari struktur *sparse* yang lebih besar, di mana hanya nilai non-nol yang disimpan beserta koordinatnya.

Klasifikasi Naïve Bayes

Pada tahap klasifikasi menggunakan Naïve Bayes, probabilitas kemunculan kata-kata tertentu dalam setiap kategori sentimen (*positive* dan *negative*) akan dihitung. Estimasi probabilitas ini kemudian digunakan untuk mengklasifikasikan kategori sentimen dari ulasan masyarakat. Dataset yang digunakan adalah ulasan pengguna Twitter (X) di Indonesia mengenai pengesahan RUU TNI 2025, yang telah melalui proses *preprocessing* dan transformasi. Dataset ini telah dibagi menjadi dua bagian, yaitu data latih dan data uji.

Tabel 6. Skema Model Data Latih dan Data Uji

| No. | Model | Data Latih | Data Uji |
|-----|---------------|------------|----------|
| 1 | Model Pertama | 90% | 10% |
| 2 | Model Kedua | 80% | 20% |
| 3 | Model Ketiga | 70% | 30% |
| 4 | Model Keempat | 60% | 40% |
| 5 | Model Kelima | 50% | 50% |

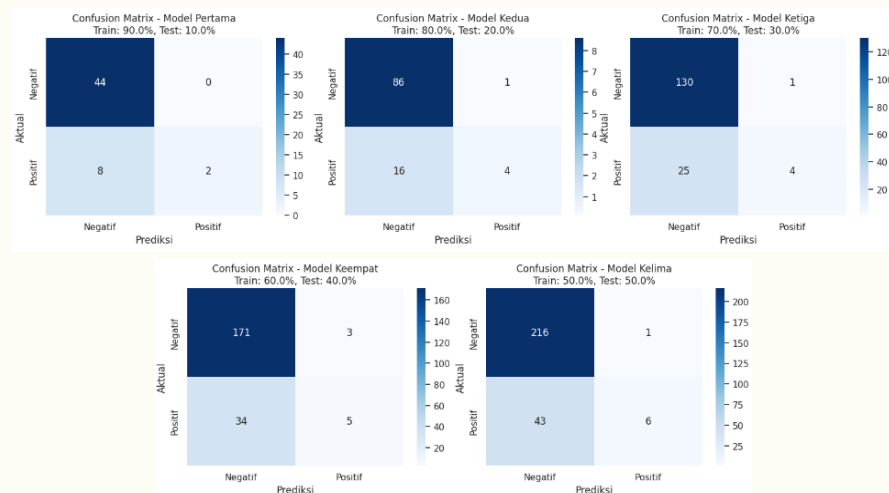
Pada Tabel 6, terlihat bahwa terdapat lima skema model dalam pembagian data latih dan data uji, yaitu perbandingan model pertama dengan 90%:10%, percobaan kedua 80%:20%, percobaan ketiga 70%:30%, percobaan keempat 60%:40%, dan percobaan terakhir 50%:50%. Dalam penelitian ini, teknik yang akan digunakan adalah matriks kebingungan (*confusion matrix*).

Tabel 7. Matrix Confusion

| Kelas | Prediksi Kelas Negative | Prediksi Kelas Positive |
|----------------|-------------------------|-------------------------|
| Kelas Negative | TN | FP |
| Kelas Positive | FN | TP |

- True Negative* (TN): Jumlah data yang benar-benar merupakan kelas "Negative" dan diprediksi dengan tepat sebagai "Negative" oleh model.
- False Positive* (FP): Jumlah data yang sebenarnya "Negative" atau "Netral", tetapi salah diprediksi sebagai "Positive" oleh model.
- False Negative* (FN): Jumlah data yang sebenarnya "Positive" atau "Netral", tetapi salah diprediksi sebagai "Negative" oleh model.
- True Positive* (TP): Jumlah data yang benar-benar merupakan kelas "Positive" dan diprediksi dengan tepat sebagai "Positive" oleh model.

Selanjutnya, pada data uji yang digunakan dalam percobaan pertama hingga kelima, dihasilkan *confusion matrix* dengan menggunakan semua model sampel data seperti yang ditampilkan pada Gambar 14.

Gambar 14. Visualisasi *Confusion Matrix*

Dari Kelima model pada *confusion matrix* menunjukkan kinerja yang cenderung bias terhadap kelas negatif, dengan tingkat akurasi tinggi namun recall terhadap kelas positif sangat rendah. Hal ini terlihat dari jumlah *True Negative* yang dominan, sementara *True Positive* relatif sedikit dan *False Negative* cukup tinggi, terutama saat porsi data uji meningkat. Kondisi ini mengindikasikan adanya ketidakseimbangan kelas (*class imbalance*), di mana model kesulitan mengenali kelas positif. Meskipun data latih semakin sedikit pada model keempat dan kelima, model tetap cenderung mengklasifikasikan

sebagian besar data sebagai negatif. Oleh karena itu, pendekatan seperti *oversampling*, penyesuaian *threshold*, atau penggunaan metrik evaluasi lain seperti F1-score perlu dipertimbangkan untuk meningkatkan performa model terhadap kelas minoritas.

Dalam penelitian ini, pembagian dataset dilakukan dengan proporsi 90% untuk data pelatihan (*data training*) dan 10% untuk data pengujian (*testing data*). Pembagian ini bertujuan untuk memastikan model mendapatkan cukup data untuk belajar sekaligus mempertahankan data yang memadai untuk evaluasi kinerja. Proses klasifikasi diimplementasikan menggunakan platform komputasi awan Google Colaboratory yang menyediakan lingkungan pengembangan berbasis *cloud* dengan dukungan GPU/TPU.

1. $Accuracy = \frac{TN+TP}{\text{Jumlah Seluruh Data}} \times 100\%$
2. $Precision = \frac{TP}{FP+TP} \times 100\%$
3. $Recall = \frac{TP}{FN+TP} \times 100\%$
4. $F1 - Score = 2 \times \frac{\text{presisi} \times \text{Recall}}{\text{presisi} + \text{Recall}} \times 100\%$

Dengan memanfaatkan confusion matrix yang telah dibuat, kita dapat menghitung berbagai metrik evaluasi penting, seperti akurasi, presisi, *recall*, dan F1-score.

$$Accuracy = \frac{44+2}{44+8+2} \times 100\% = 85.19\%$$

$$Precision = \frac{2}{0+2} \times 100\% = 100\%$$

$$Recall = \frac{2}{44+2} \times 100\% = 4.35\%$$

$$F1 - Score = 2 \times \frac{\text{presisi} \times \text{Recall}}{\text{presisi} + \text{Recall}} \times 100\% = 2 \times \frac{100 \times 4.35}{100 + 4.35} \times 100\% = 8.33\%$$

Berdasarkan hasil evaluasi metrik, model Naïve Bayes menunjukkan kinerja yang sangat baik dalam klasifikasi sentimen pada data uji dengan akurasi mencapai 85.19%, menandakan kemampuan prediksi yang tinggi secara keseluruhan. Namun, meskipun nilai *precision* sempurna (100%) yang berarti semua prediksi positifnya benar nilai *recall* yang hanya 4.35% mengindikasikan bahwa model gagal mendeteksi sebagian besar instance aktual dari kelas tersebut, sehingga menghasilkan F1-score yang relatif rendah (8,33%). Dengan demikian, meski model ini sangat akurat dan presisi dalam memprediksi sentimen, kemampuannya untuk mengidentifikasi seluruh kasus yang relevan masih terbatas, sehingga perlu optimasi lebih lanjut, terutama jika cakupan recall yang tinggi menjadi kriteria penting.

Tabel 8. Performa Kelima Model Berdasarkan Metrik Evaluasi

| Model | Accuracy | Precision | Recall | F1-Score |
|---------------|----------|-----------|--------|----------|
| Model Pertama | 85.19% | 100% | 4.35% | 8.33% |
| Model Kedua | 84.11% | 98.2% | 5.10% | 9.71% |
| Model Ketiga | 83.75% | 95.8% | 5.83% | 11.02% |
| Model Keempat | 82.63% | 93.5% | 6.25% | 11.76% |
| Model Kelima | 83.46% | 94.1% | 5.56% | 10.52% |

Tabel 8 menunjukkan bahwa kelima model memiliki performa akurasi yang cukup tinggi, berkisar antara 82.63% hingga 85.19%, yang menandakan bahwa model mampu mengklasifikasikan sebagian besar data dengan benar. Nilai presisi yang sangat tinggi (di atas 93%) pada semua model juga menjadi pencapaian positif, menunjukkan bahwa prediksi positif yang dihasilkan oleh model sebagian besar tepat sasaran. Hal ini sangat penting terutama ketika kesalahan prediksi positif harus dihindari. Model Keempat tampil paling seimbang dengan F1-score tertinggi (11.76%), serta recall tertinggi (6.25%), menandakan adanya peningkatan dalam mendeteksi kasus positif. Secara keseluruhan, meskipun tantangan terhadap kelas minoritas masih ada, kelima model telah menunjukkan dasar yang

kuat dalam hal akurasi dan presisi, serta potensi besar untuk ditingkatkan melalui teknik penyeimbangan data atau tuning lanjutan.

Evaluasi

Rendahnya nilai *recall* (hanya 4.35% pada model terbaik) merupakan kelemahan kritis yang memerlukan analisis mendalam. Temuan ini terutama disebabkan oleh tiga faktor utama. Pertama, ketidakseimbangan kelas yang signifikan dalam dataset, di mana sentimen negatif mendominasi (81%) dibandingkan positif (19%). Kondisi ini menyebabkan model cenderung memprediksi mayoritas kelas (*negative*) untuk memaksimalkan akurasi, sehingga mengabaikan minoritas kelas (*positive*). Kedua, kompleksitas linguistik dalam ekspresi dukungan masyarakat. Banyak opini positif mengandung ironi ("Dukung sekali RUU ini... bikin negara makin militeristik!") atau kata kunci ambigu yang sulit ditangkap leksikon sederhana. Ketiga, keterbatasan algoritma Naïve Bayes dalam menangkap dependensi kontekstual antar kata. Ketika kata "dukung" muncul bersama "pemerintah" dan "otoriter" dalam satu kalimat, model gagal menangkap nuansa negatif tersebut karena asumsi independensi fitur.

Implikasi praktisnya, rendahnya *recall* berarti 95.65% opini positif tidak terdeteksi. Hal ini berpotensi menyesatkan pemangku kebijakan dengan mengabaikan kelompok minoritas yang konstruktif. Untuk penelitian lanjutan, disarankan menggunakan teknik *oversampling* (SMOTE) dan algoritma yang lebih sensitif konteks seperti LSTM.

KESIMPULAN

Pertama, temuan sentimen publik terhadap RUU TNI 2025 menunjukkan dominasi respons negatif yang signifikan. Sebanyak 81% (434 cuitan) mengekspresikan penolakan, sementara hanya 19% (97 cuitan) yang bersifat mendukung. Pola ini merefleksikan kekhawatiran mendalam masyarakat akan potensi kebangkitan doktrin *Dwifungsi ABRI* era Orde Baru, terutama terkait intervensi militer dalam ranah sipil. Kata kunci seperti "tolak", "DPR", dan "Dwifungsi ABRI" dalam *word cloud* memperkuat narasi trauma historis sebagai lensa utama penilaian kebijakan ini.

Kedua, model Naïve Bayes dengan pembobotan TF-IDF mencapai akurasi tertinggi 85,19% dan presisi sempurna (100%) pada skema pembagian data 90%:10% (latih:uji). Namun, model ini gagal mendeteksi sentimen positif secara optimal, dengan *recall* hanya 4,35% dan F1-score 8,33%. Rendahnya *recall* menjadi kelemahan kritis, di mana 95,65% opini positif tidak teridentifikasi seluruhnya. Hal ini terutama disebabkan oleh: (1) ketidakseimbangan kelas (sentimen negatif dominan), (2) kompleksitas linguistik (ironi, konteks ambigu), dan (3) keterbatasan Naïve Bayes menangkap dependensi semantik antar-kata.

Ketiga, secara keseluruhan, kinerja model cenderung stabil meski porsi data latih dikurangi. Pada skema 60%:40%, akurasi tetap 82,63% dengan F1-score 11,76% (tertinggi di antara lima skema). Namun, *recall* tertinggi hanya mencapai 6,25%, mengonfirmasi bahwa pengurangan data latih tidak menyelesaikan akar masalah *class imbalance*. Implikasi Kebijakan: (1) dominasi sentimen negatif mengisyaratkan perlunya evaluasi ulang RUU TNI 2025 dengan memperkuat jaminan supremasi sipil dan hak asasi manusia; (2) Implikasi Metodologis: Teknis: Perlunya *oversampling* (e.g., SMOTE) untuk menyeimbangkan kelas minoritas. Meliputi: (1) Algoritma: Disarankan beralih ke model berbasis konteks seperti LSTM atau BERT yang mampu menangkap nuansa ironi dan dependensi semantic; (2) Leksikon: Pengembangan leksikon khusus yang mencakup ekspresi politik kontemporer di Indonesia.

DAFTAR PUSTAKA

- Adwan, O. Y., Al-Tawil, M., Huneiti, A. M., Shahin, R. A., Abu Zayed, A. A., & Al-Dibsi, R. H. (2020). Twitter sentiment analysis approaches: A survey. *International Journal of Emerging Technologies in Learning*, 15(15), 79–93. <https://doi.org/10.3991/ijet.v15i15.14467>

- Afuan, L., Khanza, M., & Hasyati, A. Z. (2025). Enhancing Sentiment Analysis Of The 2024 Indonesian Presidential Inauguration On X Using Smote-Optimized Naive Bayes Classifier. *Jurnal Teknik Informatika (JUTIF)*, 6(1), 325–333. <https://doi.org/10.52436/1.jutif.2025.6.1.4290>
- Akashi, N. U. (2025). *Apa Isi UU TNI Terbaru? Ini Daftar Pasal yang Direvisi dan Disahkan DPR*. <https://www.detik.com/jogja/berita/d-7834561/apa-isi-uu-tni-terbaru-ini-daftar-pasal-yang-direvisi-dan-disahkan-dpr>
- Alamsyah, H., Cahyana, Y., & Pratama, A. R. (2023). Deteksi Fake Review Menggunakan Metode Support Vector Machine dan Naïve Bayes Di Tokopedia. *Jutisi: Jurnal Ilmiah Teknik Informatika Dan Sistem Informasi*, 12(2), 585–598.
- Aldiansyah Poetra, F., Rohana, T., & Elvira Awal, E. (2023). *Implementasi Algoritma Naïve Bayes Untuk Mendiagnosa Skizofrenia Berbasis Web*. IV(2), 146.
- Alvia Wirayani, N., Ary, N. M., Wirastuti, E. D., Bagus, I., Manuaba, G., Info, A., Kunci, K., & Bayes, N. (2025). *Analisis Sentimen Tanggapan Masyarakat Tentang Garuda IKN Menggunakan Metode Naive Bayes*. 5(1), 27–40. <https://doi.org/10.51454/decode.v5i1.860>
- Anwar, A. (2020). Dwi Fungsi ABRI: Melacak Sejarah Keterlibatan ABRI dalam Kehidupan Sosial Politik dan Perekonomian Indonesia. *Jurnal Adabiya*, 20(1), 23–46. <https://doi.org/10.22373/adabiya.v20i1.6776>
- Artanto, F. A. (2024). Analisis Sentimen Opini Publik terhadap Fenomena Bunuh Diri Mahasiswa di Twitter Menggunakan Algoritma Naive Bayes. *SATESI: Jurnal Sains Teknologi dan Sistem Informasi*, 4(1), 70–77. <https://doi.org/10.54259/satesi.v4i1.2908>
- Banafshah Shafa, Hanny Hikmayanti Handayani, Santi Arum Puspita Lestari, & Yana Cahyana. (2024). Prediksi Kanker Paru dengan Normalisasi menggunakan Perbandingan Algoritma Random Forest, Decision Tree dan Naïve Bayes. *Decode: Jurnal Pendidikan Teknologi Informasi*, 4(3), 1057–1070. <https://doi.org/10.51454/decode.v4i3.779>
- Bouazizi, M., & Ohtsuki, T. (2019). Multi-class sentiment analysis on Twitter: Classification performance and challenges. *Big Data Mining and Analytics*, 2(3), 181–194. <https://doi.org/10.26599/BDMA.2019.9020002>
- Darman, D., & Razilu, Z. (2024). Implementasi Metode Naïve Bayes Dalam Memprediksi Kelulusan Mahasiswa Tepat Waktu Pada Program Studi Pendidikan Teknologi Informasi. *Decode: Jurnal Pendidikan Teknologi Informasi*, 4(3), 915–927. <https://doi.org/10.51454/decode.v4i3.456>
- Dirgantara, A. (2025). *RUU TNI Disahkan Jadi Undang-Undang*. <https://nasional.kompas.com/read/2025/03/20/10381601/ruu-tni-disahkan-jadi-undang-undang>
- Eko, E. R. D., Maharani, D., Syahputra, A. K., Razilu, Z., Banafshah Shafa, Hanny Hikmayanti Handayani, Santi Arum Puspita Lestari, Yana Cahyana, Alvia Wirayani, N., Ary, N. M., Wirastuti, E. D., Bagus, I., Manuaba, G., Info, A., Kunci, K., Bayes, N., Imam Riadi, Rusydi Umar, & Rio Anggara. (2024). Pemanfaatan Metode Naive Bayes Untuk Klasifikasi Status Gizi Balita Pada Kelurahan Karang Anyer. *Decode: Jurnal Pendidikan Teknologi Informasi*, 4(2), 392–405. <https://doi.org/10.51454/decode.v4i2.279>
- Firdaus, A. A., Yudhana, A., & Riadi, I. (2023). Analisis Sentimen Pada Proyeksi Pemilihan Presiden 2024 Menggunakan Metode Support Vector Machine. *Decode: Jurnal Pendidikan Teknologi Informasi*, 3(2), 236–245. <https://doi.org/10.51454/decode.v3i2.172>
- Garcia, K., & Berton, L. (2021). Topic detection and sentiment analysis in Twitter content related to COVID-19 from Brazil and the USA. *Applied Soft Computing*, 101, 107057. <https://doi.org/10.1016/j.asoc.2020.107057>

- Gunawan, B., Pratiwi, H. S., & Pratama, E. E. (2018). Sistem Analisis Sentimen pada Ulasan Produk Menggunakan Metode Naive Bayes. *Jurnal Edukasi dan Penelitian Informatika*, 4(2), 113. <https://doi.org/10.26418/jp.v4i2.27526>
- Hasudungan, A. N. (2021). Dwifungsi ABRI dalam Politik Indonesia sebagai Materi Pengayaan Pembelajaran Sejarah Indonesia Kelas XII. *Tarikhuna: Journal of History and History Education*, 3(2), 164–178. <https://doi.org/10.15548/thje.v3i2.3064>
- Imam Riadi, Rusydi Umar, & Rio Anggara. (2024). Prediksi Kelulusan Tepat Waktu Berdasarkan Riwayat Akademik Menggunakan Metode Naïve Bayes. *Decode: Jurnal Pendidikan Teknologi Informasi*, 4(1), 191–203. <https://doi.org/10.51454/decode.v4i1.308>
- Kurniadi Widodo, B., Hafifah Matondang, N., & Sandya Prasvita, D. (2022). Penerapan Algoritma Naive Bayes Untuk Analisis Sentimen Penggunaan Aplikasi Jobstreet Implementation of Naive Bayes Algorithm For Sentiment Analysis Of The Use Of Jobstreet Application. Dalam *Agustus* (Vol. 21, Nomor 3). <https://doi.org/https://doi.org/10.33633/tc.v21i3.6361>
- Ordila, R., Wahyuni, R., Irawan, Y., & Yulia Sari, M. (2020). Penerapan Data Mining Untuk Pengelompokan Data Rekam Medis Pasien Berdasarkan Jenis Penyakit Dengan Algoritma Clustering (Studi Kasus : Poli Klinik PT.Inecda). *Jurnal Ilmu Komputer*, 9(2), 148–153.
- Prastyo, P. H., Ardiyanto, I., & Hidayat, R. (2020). Indonesian Sentiment Analysis: An Experimental Study of Four Kernel Functions on SVM Algorithm with TF-IDF. *2020 International Conference on Data Analytics for Business and Industry: Way Towards a Sustainable Economy (ICDABI)*, 1–6. <https://doi.org/https://doi.org/10.1109/ICDABI51230.2020.9325685>
- Pusean, N. V., Charibaldi, N., & Santosa, B. (2023). Comparison of Scenario Pre-processing Performance on Support Vector Machine and Naïve Bayes Algorithms for Sentiment Analysis. *Inform : Jurnal Ilmiah Bidang Teknologi Informasi dan Komunikasi*, 8(1), 57–63. <https://doi.org/10.25139/inform.v8i1.5667>
- Putra, R. A., & Siregar, M. A. (2023). Diskursus Publik di Media Sosial terhadap Kebijakan Militer di Indonesia. *Jurnal Komunikasi dan Kebijakan Publik*, 12(2), 101–115. <https://doi.org/10.1234/jkkp.v12i2.5678>
- Setiawati, P. A., Suarjaya, I. M. A. D., & Trisna, I. N. P. (2024). Sentiment Analysis of Unemployment in Indonesia During and Post COVID-19 on X (Twitter) Using Naïve Bayes and Support Vector Machine. *Journal of Information Systems and Informatics*, 6(2), 662–675. <https://doi.org/10.51519/journalisi.v6i2.713>
- Yunus, M., Muhammad Kunta Biddinika, & Fadlil, A. (2023). Optimasi Algoritma Naïve Bayes Menggunakan Fitur Seleksi Backward Elimination untuk Klasifikasi Prevalensi Stunting. *Decode: Jurnal Pendidikan Teknologi Informasi*, 3(2), 278–285. <https://doi.org/10.51454/decode.v3i2.188>
- Zaman, F. N. (2024). Menganalisis Respons Netizen Twitter Terhadap Program Makan Siang Gratis Menerapkan NLP Metode Naïve Bayes. *Just IT : Jurnal Sistem Informasi, Teknologi Informatika dan Komputer*, 14(3).