



Implementasi Algoritma Naïve Bayes Classifier Pada Sistem Deteksi Berita Hoaks

Rafi Bagus Musyaffa¹, Betha Nurina Sari², Iqbal Maulana³

^{1,2,3}Universitas Singaperbangsa Karawang

Abstract

Received: 06 Februari 2026
Revised: 16 Februari 2026
Accepted: 28 Februari 2026

The rapid advancement of information technology has enabled the widespread dissemination of information through digital platforms, but it has also led to the spread of false or hoax news. This study aims to develop a text-based news classification system using the Naïve Bayes Classifier algorithm to detect and distinguish between hoax and non-hoax news. The research uses the Knowledge Discovery in Databases (KDD) method, which includes data selection, preprocessing (case folding, tokenization, stopword removal, and stemming), feature extraction with TF-IDF, and classification using Naïve Bayes. Evaluation is performed using a confusion matrix to measure accuracy, precision, recall, and F1-score. The resulting model achieved an accuracy of 83.15%. It showed a high recall of 94% for the hoax class, indicating strong performance in identifying hoax news. However, the recall for non-hoax news was only 60%, showing limitations in detecting legitimate news. Precision was relatively balanced at 83% for hoax and 82% for non-hoax. Overall, the Naïve Bayes algorithm proved to be fairly effective in building a text-based hoax news detection system and is expected to serve as an initial solution to help the public verify the truth of information automatically and efficiently.

Keywords: Hoax news, naive bayes, text classification, KDD

(*) Corresponding Author:

How to Cite: Musyaffa, R., Sari, B., & Maulana, I. (2026). Implementasi Algoritma Naive Bayes Classifier pada Sistem Deteksi Berita Hoaks. *Jurnal Ilmiah Wahana Pendidikan*, 12(3.C), 80-91. Retrieved from <https://jurnal.peneliti.net/index.php/JIWP/article/view/12752>.

PENDAHULUAN

Perkembangan teknologi informasi yang pesat telah mempermudah penyebaran informasi melalui platform digital, seperti situs berita online maupun media social. Perkembangan ini memberikan dampak positif, seperti peningkatan akses terhadap berita terkini. Walaupun tujuan utama diciptakannya teknologi informasi untuk menghasilkan dampak yang positif, namun di sisi lain teknologi tersebut juga memberikan dampak yang negatif jika tidak digunakan dengan bijak (Mutaqin, Dharmopadni, & Mutaqin, 2024). Salah satu dampak negatif dari penyebaran teknologi informasi adalah maraknya penyebaran berita palsu atau hoaks.

Hoaks adalah informasi yang direayasa untuk menutupi informasi yang sebenarnya (Nugroho, 2022). Hoaks merupakan informasi atau berita yang tidak benar atau menyesatkan, yang sengaja disebarluaskan untuk tujuan tertentu, seperti menipu, mempengaruhi opini publik, atau sekedar menarik perhatian. Hoaks seringkali disebar dengan sangat luas dan cepat melalui media sosial dan platform digital lainnya. Media sosial seperti Whatsapp, Instagram, Twitter,

Facebook, dan Tiktok merupakan media yang paling sering digunakan untuk penyebaran berita hoaks (Oktaviana, Dewi, & Purnamasari, 2021).

Berdasarkan data dari Kementerian Komunikasi dan Digital (Komdigi), sepanjang tahun 2024, teridentifikasi 1923 konten hoaks. Kategori hoaks yang paling banyak ditemukan adalah penipuan, dengan jumlah 890 konten. Disusul oleh pemerintahan sebanyak 214 konten, dan kesehatan sebanyak 145 konten. Hal ini menunjukkan bahwa deteksi dini terhadap berita hoaks menjadi sangat penting untuk menjaga integritas informasi di masyarakat.

Kendala utama dalam menangani penyebaran berita hoaks terletak pada jumlah informasi yang sangat besar dan terus bertambah setiap harinya, sehingga tidak memungkinkan untuk dilakukan verifikasi secara manual oleh manusia. Tingginya jumlah konten hoaks tersebut menunjukkan perlunya Upaya sistematis dan efisien dalam mendeteksi serta mengklasifikasikan informasi yang tersebar di internet. Salah satu pendekatan yang dapat digunakan adalah penerapan metode klasifikasi teks. Dalam konteks ini, algoritma klasifikasi berperan dalam mempelajari pola – pola dari data teks dan mengkategorikan berita berdasarkan ciri – ciri tertentu. Salah satu algoritma sederhana yang cukup efektif dalam menangani teks adalah Naïve Bayes Classifier.

Algoritma Naïve Baiyes Classifier merupakan algoritma berbasis probabilistik yang menggunakan Teorema Bayes dengan asumsi independensi antar fitur. Algoritma Naïve Bayes tetap efektif dalam berbagai situasi kompleks meski asumsi independensinya tidak realistis, karena algoritma ini sederhana, mudah diskalakan dan kinerjanya cukup baik (Pei, 2020). Salah satu kekurangan utama yang sering disebutkan dalam literatur adalah asumsi independensi antara fitur yang digunakan dalam Naive Bayes. Asumsi ini sering kali tidak realistis dalam banyak aplikasi dunia nyata, seperti dalam pengklasifikasian teks atau data sekuensial, yang dapat menyebabkan penurunan akurasi prediksi (Hussain et al, 2022).

Penelitian untuk klasifikasi menggunakan Algoritma Naïve Bayes Classifier sudah pernah dilakukan sebelumnya oleh Triana et al (2024), penelitian tersebut untuk mengklasifikasi sentiment ulasan pengguna aplikasi KAI *Access* yang mana penelitian tersebut mendapatkan hasil 97,26% untuk *precision*, 95,65% untuk skor *recal*, dan 96,27% untuk skor *f1*. Hasil ini menunjukkan algoritma Naïve Bayes Classifier efektif dan dapat diandalkan untuk proses klasifikasi. Penelitian lain yang terkait dengan topik ini dilakukan oleh Imran, Karim, & Ningsih (2024), penelitian tersebut mengklasifikasi berita hoaks terkait Pemilu Presiden 2024 yang mana penelitian tersebut mendapatkan hasil 97% untuk *precision*, 100% untuk *recall*, dan 94% untuk skor *f1*. Hasil ini menunjukkan algoritma Naïve Bayes Classifier efektif dan dapat diandalkan untuk proses klasifikasi.

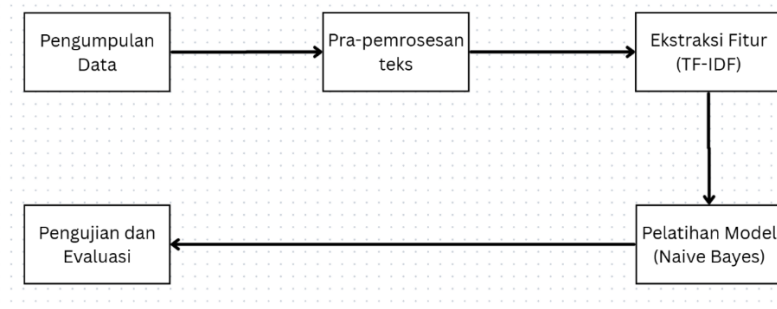
Berdasarkan latar belakang diatas maka penulis akan melakukan penelitian dengan judul “Implementasi Algoritma Naïve Bayes Classifier pada Sistem Deteksi Berita Hoaks”. Dalam konteks penelitian ini, algoritma Naïve Bayes Classifier akan diterapkan untuk membangun sistem klasifikasi berita berbasis teks yang dapat membedakan antara berita hoaks dan berita asli. Dengan sistem ini, diharapkan pengguna dapat memperoleh informasi yang lebih valid.

METODOLOGI PENELITIAN

Metodologi penelitian yang digunakan dalam penelitian ini adalah Knowledge Discovery in Database (KDD), KDD adalah suatu rangkaian proses yang mencakup kegiatan penemuan pola atau kecenderungan dari data melalui *data mining*, kemudian menginterpretasikan hasil tersebut menjadi informasi yang akurat dan mudah dipahami oleh pengguna python (Utomo, Kurniasari, & Yuniartati, 2023).

Istilah *data mining* dan *knowledge discovery in databases* (KDD) kerap digunakan secara bergantian untuk menggambarkan proses pencarian informasi tersembunyi dari basis data yang berskala besar. Namun, secara konsep keduanya memiliki perbedaan meskipun saling berkaitan. *Data mining* sendiri merupakan salah satu tahapan penting dalam keseluruhan proses KDD (Pangabean, Bulolo, & Silalahi, 2020).

Rancangan penelitian ini disusun untuk menggambarkan alur proses dalam membangun sistem deteksi berita hoaks berbasis algoritma Naive Bayes dengan pendekatan KDD. Proses ini dimulai dari pengumpulan data hingga evaluasi model, sebagaimana ditunjukkan pada Gambar 1



Gambar 1. Rancangan Penelitian

Penjelasan tiap tahap:

1. Pengumpulan Data

Tahap pemahaman data dimulai dengan pengumpulan data awal dan dilanjutkan dengan berbagai aktivitas guna mengenali data, mengidentifikasi masalah kualitas data, menemukan wawasan pertama ke dalam data atau mendeteksi subset menarik untuk membentuk hipotesis bagi informasi tersembunyi (Brzozowska et al, 2023). Mengumpulkan dataset berita dari berbagai sumber daring, seperti Kaggle dan situs berita nasional. Data ini dilabeli sebagai hoaks dan non-hoaks.

2. Pra-pemrosesan.Teks

Dilanjutkan dengan fase *Data Preparation* atau persiapan data, yang mencakup semua kegiatan untuk membangun dataset akhir dari data mentah awal. Melakukan proses seperti case folding, tahapan ini bertujuan untuk menyeragamkan bentuk kata. Pada proses ini, setiap kata diubah menjadi huruf kecil dengan memanfaatkan fungsi *lower case* dalam bahasa pemrograman Python. Selanjutnya tokenisasi, tahapan ini berfungsi untuk memisahkan kalimat menjadi unit-unit kecil yang disebut token. Token ini bisa berupa kata, frasa, atau satuan makna lainnya. Proses ini dapat dilakukan dengan memanfaatkan pustaka *nlk.tokenize* dalam Python untuk memecah kalimat menjadi bagian-bagian huruf atau kata. Selanjutnya *stopword removal*, tahapan ini digunakan untuk menyaring kata-kata yang tidak memberikan kontribusi

signifikan terhadap pemahaman data dalam penelitian. Selanjutnya stemming untuk membersihkan dan menstandarkan data teks.

3. Ekstraksi Fitur (TF-IDF)

TF-IDF (Term Frequency-Inverse Document Frequency) merupakan metode yang digunakan untuk mengubah teks menjadi representasi numerik. Tujuannya adalah untuk memberikan bobot pada setiap kata atau fitur berdasarkan tingkat kepentingannya dalam suatu dokumen (Rahmadani, Rahim, & Rudiman, 2024). Nilai yang dihasilkan dari pembobotan menggunakan TF-IDF cenderung menurun jika suatu kata sering muncul dalam banyak dokumen, karena metode ini didasarkan pada frekuensi dokumen terbalik. Artinya, semakin sering sebuah kata muncul, semakin rendah bobotnya, sedangkan kata yang jarang muncul akan memiliki bobot yang lebih tinggi.(Grandis, & Arumsari, 2021). Pada kerangka kerja KDD proses TF-IDF dilakukan untuk memahami data dan menyipakan data (tahap data understanding dan data preparation), proses seleksi diperlukan untuk membersihkan dan merapikan data ulasan yang telah dikumpulkan (Singgalen, 2023).

4. Pelatihan Model (Naïve_Bayes)

Setelah menyelesaikan fase ketiga, proses dilanjutkan ke tahap selanjutnya, yaitu fase *modeling*. Pada tahap ini, berbagai teknik pemodelan dipilih dan digunakan sesuai kebutuhan. Beberapa metode memiliki persyaratan format data tertentu. Dalam penelitian ini, proses Modeling dilakukan dengan menerapkan algoritma Naïve Bayes untuk memperoleh hasil yang optimal. Pemodelan meliputi pemilihan teknik, pembuatan, dan penilaian model.

5. Evaluasi dan Analisis

Menilai kinerja model berdasarkan metrik evaluasi seperti akurasi, presisi, recall, dan F1-score untuk mengetahui efektivitas sistem dalam mendeteksi berita hoaks.. Tahap evaluasi dalam KDD merupakan proses penilaian terhadap model yang telah dibangun, guna memastikan bahwa model tersebut akurat, relevan, dan selaras dengan tujuan yang ingin dicapai.

HASIL DAN PEMBAHASAN

Tahap 1 – Data Selection

Data yang digunakan dalam penelitian ini diperoleh dari situs Kaggle (<https://www.kaggle.com>), yaitu platform berbagi dataset yang umum digunakan dalam bidang data science dan machine learning. Dataset yang digunakan berjudul " Deteksi Berita Hoaks Indo - Dataset", yang diunggah oleh salah satu pengguna Kaggle dan berasal dari hasil web scraping terhadap situs turnbackhoax.id.

Pada tahapan ini data yang telah diperoleh dari situs Kaggle akan dilakukan *data selection*, yaitu akan dipilih atribut-atribut yang akan digunakan dalam proses pembersihan data dan klasifikasi nantinya. Beberapa atribut yang tidak digunakan akan dihapus dari data yang telah dimiliki. Sebelumnya data yang telah diperoleh memiliki atribut diantaranya url, judul, tanggal, isi_berita, Narasi, Clean Narasi.

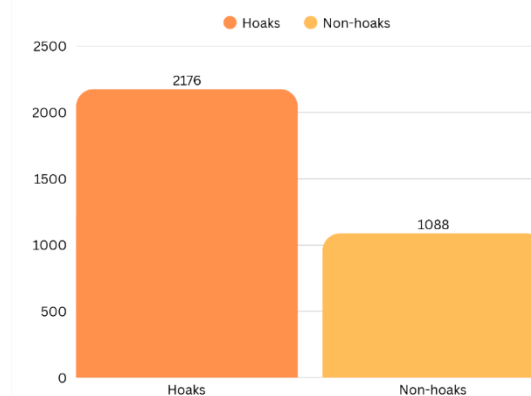
Dataset yang digunakan dalam penelitian ini merupakan kumpulan berita yang telah dilabeli ke dalam dua kategori utama, yaitu benar dan salah. Namun, format awal dari dataset tersebut tidak memisahkan label secara eksplisit dalam kolom terpisah, melainkan label tergabung di dalam teks judul berita. Oleh karena itu, dilakukan proses ekstraksi label dengan memanfaatkan pola kata tertentu pada

judul yang merepresentasikan kategori berita tersebut. Setelah label berhasil diekstrak, judul kemudian dibersihkan dari kata-kata label tersebut agar tidak memengaruhi proses pelatihan model klasifikasi. Gambar 2 adalah data setelah proses ekstraksi label.

	judul	judul_bersih	label
0	[BENAR] Tautan âNew Gebyar Program Bank BCA ...	Tautan âNew Gebyar Program Bank BCA Tahun 20...	non-hoaks
1	[SALAH] Video âAda Bangkai Kereta di Tebing,...	Video âAda Bangkai Kereta di Tebing, Akibat ...	hoaks
2	[SALAH] Jokowi Pakai Rp38,5 Triliun Dana Haji ...	Jokowi Pakai Rp38,5 Triliun Dana Haji Rakyat T...	hoaks
3	[BENAR] Pencairan Dana Bansos Kemenkeu	Pencairan Dana Bansos Kemenkeu	non-hoaks
4	[BENAR] Tautan âBantuan Subsidi Upahâ	Tautan âBantuan Subsidi Upahâ	non-hoaks

Gambar 2. Data Setelah Ekstraksi Label

Dataset awal yang digunakan dalam penelitian ini terdiri dari 10.033 data berita berlabel hoaks dan 1.088 data berita berlabel non-hoaks. Terlihat adanya ketidakseimbangan jumlah data antar kelas (class imbalance) yang cukup signifikan, di mana jumlah data hoaks jauh lebih besar dibanding non-hoaks. Ketidakseimbangan ini dapat menyebabkan bias pada model klasifikasi, di mana model cenderung mengklasifikasikan data ke dalam kelas mayoritas (hoaks). Untuk mengatasi permasalahan tersebut, dilakukan proses undersampling terhadap data kelas mayoritas (hoaks), yaitu dengan mengurangi jumlah data hoaks dari 10083 menjadi 2176. Rasio 2:1 antara kelas hoaks dan non-hoaks digunakan dalam proses resampling guna mengurangi ketidakseimbangan data. Pendekatan ini dipilih sebagai solusi agar model tidak terlalu bias terhadap kelas mayoritas (hoaks), namun tetap memiliki cukup representasi data dari kelas minoritas (non-hoaks). Dengan demikian, model dapat melakukan klasifikasi secara lebih seimbang dan meningkatkan performa, khususnya pada metrik *recall* dan *F1-score*.. Gambar 3 adalah Data setelah proses *undersampling*.



Gambar 3. Data Setelah Proses Undersampling

Tahap 2 – Preprocessing Data

Sebelum data digunakan untuk pelatihan model klasifikasi, dilakukan serangkaian tahap preprocessing untuk membersihkan dan menyiapkan data teks agar lebih siap diolah oleh algoritma. Proses ini merupakan bagian penting dalam Natural Language Processing (NLP), karena data teks mentah umumnya masih mengandung banyak kata yang tidak relevan dan dapat mengganggu proses analisis. Tahapan preprocessing yang diterapkan dalam penelitian ini meliputi:

1. Case Folding

Pada tahap ini, seluruh huruf pada teks diubah menjadi huruf kecil. Tujuannya adalah untuk menyeragamkan bentuk kata, sehingga kata seperti “Indonesia” dan “indonesia” dianggap sama dan tidak diperlakukan sebagai entitas berbeda. Gambar 4 menunjukkan hasil setelah proses *case folding*

index	judul_bersih	case_folding
0	Tautan New Gebyar Program Bank BCA Tahun 2025	tautan new gebyar program bank bca tahun 2025
1	Video Ada Bangkai Kereta di Tebing, Akibat Jembatan Runtuh	video ada bangkai kereta di tebing, akibat jembatan runtuh
2	Jokowi Pakai Rp38,5 Triliun Dana Haji Rakyat Tak Diberi Tahu	jokowi pakai rp38,5 triliun dana haji rakyat tak diberi tahu
3	Pencairan Dana Bansos Kemenkeu	pencairan dana bansos kemenkeu
4	Tautan Bantuan Subsidi Upah	tautan bantuan subsidi upah

Gambar 4. Data Setelah Case Folding

2. Cleansing

Karakter seperti titik, koma, tanda seru, tanda tanya, dan simbol lainnya dihapus karena tidak memiliki nilai informasi yang signifikan dalam proses klasifikasi. Gambar 5 merupakan data setelah proses *cleansing*.

index	case_folding	cleansing
0	tautan new gebyar program bank bca tahun 2025	tautan new gebyar program bank bca tahun
1	video ada bangkai kereta di tebing, akibat jembatan runtuh	video ada bangkai kereta di tebing akibat jembatan runtuh
2	jokowi pakai rp38,5 triliun dana haji rakyat tak diberi tahu	jokowi pakai rp triliun dana haji rakyat tak diberi tahu
3	pencairan dana bansos kemenkeu	pencairan dana bansos kemenkeu
4	tautan bantuan subsidi upah	tautan bantuan subsidi upah

Gambar 5. Data Setelah Cleansing

3. Tokenisasi

Kalimat dipecah menjadi unit kata yang disebut token. Contohnya, kalimat “Presiden Jokowi mengunjungi Papua” akan diubah menjadi deretan kata [“presiden”, “jokowi”, “mengunjungi”, “papua”]. Gambar 6 adalah data setelah proses tokenisasi.

index	cleansing	token
0	tautan new gebyar program bank bca tahun	tautan,new,gebyar,program,bank,bca,tahun
1	video ada bangkai kereta di tebing akibat jembatan runtuh	video,ada,bangkai,kereta,di,tebing,akibat,jembatan,runtuh
2	jokowi pakai rp triliun dana haji rakyat tak diberi tahu	jokowi,pakai,rp,triliun,dana,haji,rakyat,tak,diberi,tahu
3	pencairan dana bansos kemenkeu	pencairan,dana,bansos,kemenkeu
4	tautan bantuan subsidi upah	tautan,bantuan,subsidi,upah

Gambar 6. Data Setelah Tokenisasi

4. Stopword Removal

Kata-kata umum yang sering muncul dalam bahasa Indonesia namun memiliki kontribusi kecil terhadap makna, seperti “dan”, “di”, “ke”, atau “dari”, dihapus menggunakan daftar stopwords dari pustaka nltk. Gambar 4.8 adalah data setelah proses *stopword removal*.

index	token	stopword
0	tautan,new,gebyar,program,bank,bca,tahun	tautan,new,gebyar,program,bank,bca
1	video,ada,bangkai,kereta,di,tebing,akibat,jembatan,runtuh	video,bangkai,kereta,tebing,akibat,jembatan,runtuh
2	jokowi,pakai,rp,triliun,dana,haji,rakyat,tak,diberi,tahu	jokowi,pakai,rp,triliun,dana,haji,rakyat
3	pencairan,dana,bansos,kemenkeu	pencairan,dana,bansos,kemenkeu
4	tautan,bantuan,subsidi,upah	tautan,bantuan,subsidi,upah

Gambar 7. Data Setelah Stopword Removal

5. Stemming

Stemming dilakukan untuk mengubah kata ke bentuk dasarnya atau akar katanya. Sebagai contoh, kata “bermain”, “memainkan”, dan “permainan” akan dikembalikan ke bentuk “main”. Proses ini dilakukan menggunakan library Sastrawi yang dirancang khusus untuk bahasa Indonesia. Gambar 4.9 adalah data setelah proses *stemming*

index	stopword	stemming
0	tautan,new,gebyar,program,bank,bca	taut,new,gebyar,program,bank,bca
1	video,bangkai,kereta,tebing,akibat,jembatan,runtuh	video,bangkai,kereta,tebing,akibat,jembatan,runtuh
2	jokowi,pakai,rp,triliun,dana,haji,rakyat	jokowi,pakai,rp,triliun,dana,haji,rakyat
3	pencairan,dana,bansos,kemenkeu	cair,dana,bansos,kemenkeu
4	tautan,bantuan,subsidi,upah	taut,bantu,subsidi,upah

Gambar 8. Data Setelah Proses Stemming

Tahap 3 – Ekstraksi Fitur TF-IDF

Setelah teks berita melalui tahap *preprocessing*, langkah berikutnya adalah melakukan ekstraksi fitur menggunakan metode TF-IDF (*Term Frequency–Inverse Document Frequency*). TF-IDF adalah metode pembobotan kata yang digunakan untuk merepresentasikan teks dalam bentuk numerik sehingga dapat digunakan oleh algoritma pembelajaran mesin seperti Naive Bayes.

Setelah melakukan proses *preprocessing* dan pembobotan menggunakan metode TF-IDF (*Term Frequency–Inverse Document Frequency*), diperoleh representasi fitur dalam bentuk matriks dengan dimensi (3264, 3000) seperti pada gambar 4.11. Nilai ini menunjukkan bahwa terdapat 3264 dokumen teks (berita yang sudah dibersihkan dan relevan), serta 3000 kata unik yang dipilih sebagai fitur utama berdasarkan bobot tertinggi.

Berdasarkan hasil pembobotan TF-IDF yang ditampilkan pada gambar 4.12 kata “video” memiliki skor tertinggi yaitu 82.18, menunjukkan bahwa kata tersebut paling dominan dan signifikan dalam korpus berita yang dianalisis. Kata-kata lain seperti “akun”, “foto”, dan “jokowi” juga memiliki skor tinggi, yang mengindikasikan seringnya kemunculan istilah-istilah tersebut dalam berita-berita, khususnya yang tergolong hoaks. Kehadiran kata “whatsapp”, “bantu”, “kerja”, dan “uang” memperkuat dugaan bahwa banyak berita hoaks menyangkut tema sosial dan ekonomi. Dengan demikian, hasil ini memberikan gambaran umum mengenai fokus narasi dalam dataset, sekaligus membantu dalam membangun fitur yang relevan untuk klasifikasi otomatis berita hoaks. Gambar 9 adalah hasil skor TF-IDF.

	Kata	Skor TF-IDF
2903	video	82.185824
65	akun	68.161776
811	foto	46.274676
1137	jokowi	45.406345
1155	juta	45.140201
2938	whatsapp	44.757992
1325	kerja	42.750627
235	bantu	42.212236
1022	indonesia	41.737363
2677	taut	41.397743
475	covid	40.153060
2831	uang	39.141483
1729	lowong	38.488787
913	hadiah	37.747306
772	facebook	37.233164
192	bagi	33.361267
491	daftar	32.716469
113	anies	30.465621
2375	rp	30.462681
227	bank	29.921187

Gambar 9. Hasil Skor TF-IDF

Tahap 4 – Analysis & Verification

Setelah teks melalui proses *preprocessing* dan direpresentasikan menggunakan metode TF-IDF, tahap berikutnya adalah melakukan klasifikasi dengan membangun model menggunakan algoritma Naive Bayes. Penelitian ini menggunakan varian Multinomial Naive Bayes, karena algoritma ini cocok digunakan dalam pemrosesan data teks yang telah dikonversi menjadi bentuk frekuensi atau bobot kata seperti hasil TF-IDF.

Multinomial Naive Bayes menerapkan pendekatan berbasis probabilitas, di mana model akan memperkirakan kemungkinan sebuah teks tergolong dalam kategori tertentu (hoaks atau non-hoaks), berdasarkan pola distribusi kata-kata yang terdapat di dalamnya.

Langkah-langkah Modeling yang Dilakukan:

1. Pembagian Data

Dataset yang sudah diproses dibagi menjadi dua bagian. Pembagian data dilakukan dengan menggunakan tiga skenario. Skenario pertama dilakukan dengan proporsi 80% data latih dan 20% data uji. Skenario ini adalah skenario yang paling sering dilakukan pada penelitian terkait topik ini. Sebagai perbandingan, penulis juga membuat skenario 70:30 (70% data uji, 30% data latih) dan skenario 90:10 (90% data uji dan 10% data latih). Proses ini menggunakan fungsi *train_test_split* dari Pustaka *scikit-learn*.

Tabel 1. Skenario Pembagian Data

Skenario	Data Latih	Data Uji
Skenario 70:30	2285	979
Skenario 80:20	2611	653
Skenario 90:10	2938	326

2. Pelatihan Model

Model ini menggunakan algoritma Multinomial Naïve Bayes, algoritma ini cocok digunakan untuk data teks yang sudah diubah menjadi bentuk angka yang sebelumnya dilakukan pada tahap ekstraksi fitur TF-IDF. Pada tahap ini model diberi data latih yang berisi kumpulan fitur (kata-kata yang sudah diubah menjadi angka) beserta label dari masing-masing data, apakah termasuk berita hoaks atau bukan. Melalui proses ini, model akan belajar pola dari data tersebut untuk bisa digunakan dalam memprediksi berita baru.

3. Prediksi dan Evaluasi

Setelah model selesai dilatih, dilakukan prediksi terhadap data uji. Hasil prediksi ini kemudian dibandingkan dengan label sebenarnya untuk menilai performa model.

4. Evaluasi Kinerja

Untuk mengevaluasi model, digunakan metrik-metrik berikut:

- Accuracy: persentase prediksi yang benar dari seluruh data.
- Precision: akurasi prediksi positif model.
- Recall: kemampuan model dalam menemukan semua data yang termasuk dalam kelas tertentu.
- F1-score: rata-rata harmonis dari precision dan recall.

Confusion Matrix: untuk melihat jumlah prediksi benar dan salah dari masing-masing kelas.

Tahap 5 - Evaluasi

Model klasifikasi teks yang dikembangkan dalam penelitian ini menggunakan algoritma Multinomial Naive Bayes dan dievaluasi menggunakan metrik-metrik performa seperti *accuracy*, *precision*, *recall*, dan *f1-score*. Untuk skenario 80:20 didapatkan hasil *precision*, *recall*, dan *f1-score* untuk kelas hoaks

masing-masing sebesar 0.90, 0.91, dan 0.90. Sementara itu, untuk kelas non-hoaks, *precision* sebesar 0.80, *recall* 0.79, dan *f1-score* 0.80. Sedangkan untuk akurasi dari skenario ini adalah 87% Gambar 4.14 adalah hasil klasifikasi skenario 80:20.

Untuk skenario 70:30 didapatkan hasil akurasi sebesar 87%, sedangkan *precision*, *recall*, dan *f1-score* untuk kelas hoaks masing-masing sebesar 0.88, 0.92, dan 0.90. Sedangkan untuk kelas non-hoaks, *precision* sebesar 0.82, *recall* 0.75, dan *f1-score* 0.79. Gambar 4.14 adalah hasil klasifikasi skenario 70:30.

Sedangkan untuk skenario 90:10 didapatkan hasil *precision*, *recall*, dan *f1-score* untuk kelas hoaks masing-masing adalah 0.91, 0.92, dan 0.92, sedangkan untuk kelas non-hoaks masing-masing adalah 0.84, 0.82, dan 0.83. Dan untuk akurasi pada skenario ini adalah 89%. Gambar 4.18 adalah hasil klasifikasi skenario 90:10.

Berdasarkan hasil diatas perbandingan antara skenario 70:30, 80:20, dan 90:10 dapat dilihat di table berikut. Hasil tertinggi diperoleh pada skenario 90:10, dengan akurasi mencapai 89% dan F1-score 92%. Hal ini menunjukkan bahwa semakin besar proporsi data latih, model Naïve Bayes semakin mampu mengenali pola-pola dalam data.

Tabel 2. Hasil Evaluasi Model

Skenario	Akurasi	Presisi	Recall	F1
70:30	87%	88%	92%	90%
80:20	87%	90%	91%	90%
90:10	89%	91%	92%	92%

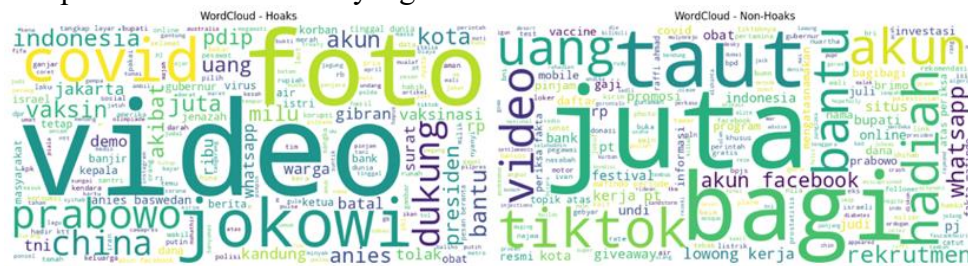
Berdasarkan *confusion matrix*, pada skenario 80:20, model menghasilkan 403 *true positive* (TP), yaitu data hoaks yang berhasil diklasifikasikan dengan benar, 40 *false negative* (FN) di mana data hoaks salah dikenali sebagai non-hoaks, 45 *false positive* (FP) berupa data non-hoaks yang salah diklasifikasikan sebagai hoaks, dan 165 *true negative* (TN) atau data non-hoaks yang berhasil dikenali dengan benar. Pada skenario 90:10, model mencatat 204 TP, 17 FN, 19 FP, dan 87 TN. Sementara itu, pada skenario 70:30, model menghasilkan 607 TP, 53 FN, 79 FP, dan 241 TN, yang menunjukkan kinerja stabil, meskipun jumlah kesalahan klasifikasi pada non-hoaks sedikit lebih tinggi dibanding skenario lainnya. Tabel 3 adalah hasil *confusion matrix* dari ketiga skenario.

Tabel 3. Hasil Confusion Matrix

Skenario	True Positive	False Negative	False Positive	True Negative
70:30	607	53	79	241
80:20	403	40	45	165
90:10	204	17	19	87

Berdasarkan hasil wordcloud konten hoaks, ditemukan bahwa sebagian besar kata yang muncul cenderung bertujuan untuk membangkitkan reaksi emosional, terutama ketakutan dan kemarahan. Kata-kata seperti “jenazah”, “bakar”, “demo”, “tangkap”, “ancam”, dan “gantung” merupakan contoh dari istilah yang sering

digunakan untuk memicu kepanikan di kalangan masyarakat. Selain itu, hoaks umumnya mengangkat topik-topik yang sensitif atau sedang viral, seperti isu politik “Jokowi”, “Prabowo”, “Anies”, “pandemi” “COVID”, “vaksin”, “corona” yang memiliki daya sebar tinggi. Selain itu, karakteristik lain yang menonjol dalam konten hoaks adalah penggunaan judul yang bersifat sensasional, dengan menyertakan kata-kata seperti “meninggal”, “bakar”, “tangkap” menarik perhatian dan menciptakan kesan dramatis yang berlebihan.



Gambar 10. Wordcloud Hasil Klasifikasi

CONCLUSION

1. Penelitian ini telah berhasil mengimplementasikan algoritma Naive Bayes dalam membangun sistem klasifikasi berita hoaks berbasis teks. Proses pembangunan sistem ini mencakup tahapan-tahapan penting, yaitu pengumpulan data dari sumber terpercaya (turnbackhoax.id melalui dataset Kaggle), pembersihan dan pra-pemrosesan data teks (melalui *case folding*, *tokenizing*, *stopword removal*, dan *stemming*), serta ekstraksi fitur menggunakan metode TF-IDF. Setelah fitur diekstraksi, dilakukan pelatihan model klasifikasi menggunakan algoritma multinomial naive bayes dan dievaluasi menggunakan metrik evaluasi seperti *accuracy*, *precision*, *recall*, dan *f1-score*.
2. Performa model Naive Bayes dalam mendeteksi berita hoaks, hasil evaluasi menunjukkan bahwa model memiliki performa yang baik dan stabil di berbagai skenario pembagian data (80:20, 70:30, dan 90:10). Nilai akurasi berkisar antara 87% hingga 89%, dengan nilai *f1-score* untuk kelas hoaks berkisar antara 0,90 hingga 0,92, yang menunjukkan bahwa model mampu mengenali berita hoaks secara konsisten dan akurat. Untuk kelas non-hoaks, nilai *f1-score* sedikit lebih rendah (sekitar 0,79 hingga 0,83), namun tetap menunjukkan kinerja yang layak. Model juga menunjukkan tingkat kesalahan klasifikasi yang rendah, berdasarkan hasil *confusion matrix*, yang berarti bahwa model cukup baik dalam membedakan antara berita hoaks dan non-hoaks.
3. Pengaruh kualitas dan jumlah data terhadap kinerja model terbukti signifikan. Skenario dengan jumlah data pelatihan yang lebih besar (seperti pada pembagian 90:10) menghasilkan performa yang lebih baik, terutama dalam mendeteksi berita non-hoaks, yang sebelumnya lebih sulit dikenali. Teknik pra-pemrosesan seperti pembersihan teks dan penghapusan kata-kata tidak penting (stopwords) juga memberikan dampak positif terhadap kinerja model, karena mampu mengurangi noise dalam data dan meningkatkan kualitas fitur yang dihasilkan. Dengan demikian, semakin baik kualitas data yang digunakan serta

semakin optimal tahap pra-pemrosesannya, maka performa model dalam mendeteksi berita hoaks juga akan semakin tinggi

REFERENCES

- Adrian, R., Musaddam, M., Ikhsan, M., & Pahlevi, M. R. B. (2024). Detection of hoax news using TF-IDF vectorizer and Multinomial Naïve Bayes and Passive Aggressive. *Media Journal of General Computer Science*, 1(2), 54–61. <https://doi.org/10.62205/mjgcs.v1i2.24>
- Astuti, D. (2019). *Penentuan strategi promosi usaha mikro kecil dan menengah (UMKM) menggunakan metode CRISP-DM dengan algoritma K-Means clustering*. *Journal of Informatics, Information System, Software Engineering and Applications (INISTA)*, 1(2), 60–72. <https://doi.org/10.20895/inista.v1i2.71>
- Fadillah Grandis, G., & Arumsari, Y. (2021). *Seleksi Fitur Gain Ratio pada Analisis Sentimen Kebijakan Pemerintah Mengenai Pembelajaran Jarak Jauh dengan K-Nearest Neighbor* (Vol. 5, Issue 8).
- Gupta, V., & Lehal, G. S. (2009). *A survey of text mining techniques and applications*. *Journal of Emerging Technologies in Web Intelligence*, 1(1), 60–76. <https://doi.org/10.4304/jetwi.1.1.60-76>.
- Harita, T. C., Kridalukmana, R., & Eridani, D. (2022). Pengembangan aplikasi analisis sentimen terhadap brand berbasis web menggunakan kerangka kerja Flask. *Jurnal Teknik Komputer*, 1(2), 36–40. <https://doi.org/10.14710/jtk.v1i2.36307>
- Imran, B., Karim, M. N., & Ningsih, N. I. (2023). *Klasifikasi berita hoax terkait pemilihan umum Presiden Republik Indonesia tahun 2024 menggunakan Naïve Bayes dan SVM*. *Jurnal Ilmiah Dinarek*, Volume 14, Nomor khusus, hlm. 61–66
- Kurnia, Y., Kusuma, E. D., Kusuma, L. W., Suwitno, S., & Apridius, W. (2024). Perbandingan Naïve Bayes dan CNN yang dioptimasi PSO pada identifikasi berita hoax politik Indonesia. *Jurnal Bina Teknologi*, 6(3), 342. <http://jurnal.kdi.or.id/index.php/bt>
- Muhammad, R., & Yulianto, S. (2023). Penerapan pemrograman Python dalam menentukan waktu overhaul kondensor turbin uap. *Jurnal Konversi Energi dan Manufaktur*, 8(1), 49–57.
- Mutaqin, R., Mutaqin, G., & Dharmopadni, D. S. (2024). *Dampak perkembangan teknologi informasi dan komunikasi terhadap dinas militer*. *Jurnal Ilmiah Multidisiplin*, 2(3), 199–204.
- Nugroho, K. S., Istiadi, I., & Marisa, F. (2020). *Naive Bayes classifier optimization for text classification on e-government using particle swarm optimization*. *Jurnal Teknologi dan Sistem Komputer*, 8(1), 21–26. <https://doi.org/10.14710/jtsiskom.8.1.2020.21-26>.
- Oktaviani, R., & Permana, R. (2022). *Integrasi text mining dan algoritma klasifikasi Naive Bayes dalam deteksi berita hoaks*. *Jurnal Teknologi dan Sistem Informasi*, 8(3), 201–215..
- Panggabean, D. S. O., Buulolo, E., & Silalahi, N. (2020). Penerapan data mining untuk memprediksi pemesanan bibit pohon dengan regresi linear berganda. *JURIKOM (Jurnal Riset Komputer)*, 7(1), 56.

<https://doi.org/10.30865/jurikom.v7i1.1947>

- Prayoga, S. T. A., Hadiana, A. I., & Umbara, F. R. (2021). Deteksi hoax pada berita online bahasa Inggris menggunakan Bernoulli Naïve Bayes dengan ekstraksi fitur TF-IDF. *Jurnal Syntax Admiration*, 2(10).
- Putra, K. T., Hariyadi, M. A., & Crysdiyan, C. (2023). *Perbandingan feature extraction TF-IDF dan BOW untuk analisis sentimen berbasis SVM*. *Jurnal Cahaya Mandalika*, 3(2), 1449–1463. Retrieved from <https://ojs.cahayamandalika.com/index.php/jcm/article/view/2292>
- Sriyano, C. S., & Setiawan, E. B. (2021). Pendeteksian berita hoax menggunakan Naive Bayes Multinomial pada Twitter dengan fitur pembobotan TF-IDF. *e-Proceeding of Engineering*, 8(2), 3396.
- Triana, R. F., Sari, A. I. P., Bahtiar, A., & Wahyudin, E. (2025). *Implementasi algoritma Naïve Bayes untuk klasifikasi sentimen ulasan pengguna KAI Access*. *Jutisi : Jurnal Ilmiah Teknik Informatika dan Sistem Informasi*, 14(1). <https://doi.org/10.35889/jutisi.v14i1.2437>
- Utomo, Y. B., Kurniasari, I., & Yanuartanti, I. (2023). Penerapan Knowledge Discovery in Database untuk Analisa Tingkat Kecelakaan Lalu Lintas. *Jurnal Teknik Informatika Kaputama (JTIK)*, 7(1), 1–8.