



## KLASIFIKASI BERITA PALSU BERBAHASA INDONESIA MENGUNAKAN ALGORITMA NAIVE BAYES BERBASIS WEB

Taufik Fajar Mustafa<sup>1)</sup>, Henny Alfianti<sup>2)</sup>

<sup>1</sup>Program Studi Teknik Informatika, Politeknik TEDC Bandung

Corresponding Author: <sup>1</sup>tfajar321321@gmail.com

### Article Info

#### Article history:

Received: May 24, 2025

Revised: Agust 10, 2025

Accepted: Okt 20, 2025

Published: Nov 10, 2025

#### Keywords:

Deteksi Berita Palsu

Naive Bayes

Klasifikasi Teks

Machine Learning

Aplikasi Web

### ABSTRACT

Penyebaran berita palsu (hoaks) menjadi tantangan besar di era digital, terutama melalui media sosial yang menyebarkan informasi secara masif dan cepat. Untuk mengatasi masalah ini, penelitian ini mengembangkan aplikasi web berbasis algoritma Naive Bayes untuk mendeteksi berita palsu berbahasa Indonesia. Dataset yang digunakan terdiri dari 19697 berita yang dikumpulkan dari berbagai sumber, baik fakta maupun hoaks. Proses pelatihan melibatkan tahapan preprocessing teks, vectorization dengan TF-IDF, serta teknik validasi menggunakan K-Fold Cross Validation. Hasil pengujian menunjukkan bahwa model mampu mencapai akurasi sebesar 92,5% dengan nilai precision, recall, dan f1-score yang seimbang pada kedua kelas. Aplikasi ini diimplementasikan menggunakan framework Streamlit dan dilengkapi dengan integrasi API Google News untuk mendukung verifikasi informasi secara real-time. Penelitian ini menunjukkan bahwa kombinasi antara machine learning dan teknologi web dapat menjadi solusi efektif dalam deteksi hoaks secara otomatis.



This is an open-access article distributed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International (CC BY SA 4.0)

## 1. INTRODUCTION

Penyebaran berita palsu (hoaks) merupakan tantangan besar di era digital, terutama dengan pesatnya arus informasi melalui media sosial. Berita palsu dapat menimbulkan keresahan sosial, sebagaimana terlihat pada masa pandemi COVID-19 yang mencatat ratusan sebaran hoaks [1]. Kementerian Komunikasi dan Informatika (Kominfo) mencatat telah menangani lebih dari 12.000 isu hoaks sejak 2018 [2]. Hal ini menunjukkan perlunya solusi berbasis teknologi untuk mendeteksi dan menangkal hoaks secara efisien.

Salah satu pendekatan potensial adalah penggunaan *machine learning*, yang memungkinkan sistem belajar dari data untuk mengidentifikasi pola tertentu tanpa aturan eksplisit [3]. *Machine learning* memiliki beberapa algoritma yang populer dan banyak digunakan dalam pelatihan data seperti *Random Forest*, *K-Means*, *KNN*, dll. Di antara berbagai algoritma yang tersedia, penelitian ini memilih Naive Bayes karena efisiensinya dalam klasifikasi teks serta menggunakan teorema Bayes yang terkenal efisien dari segi waktu pemrosesan [4].

Penelitian mengenai deteksi berita palsu telah banyak dilakukan dengan pendekatan algoritma machine learning. Agustina (2022) menerapkan algoritma Naive Bayes Classifier dengan pendekatan

CRISP-DM pada data dari situs Kumbaran dan memperoleh akurasi sebesar 81%, dengan deployment aplikasi ke platform Heroku menggunakan Flask [5]. Sementara itu, Nurhikam (2023) menggunakan algoritma Random Forest untuk mendeteksi berita palsu seputar Pemilu 2024 dengan dataset dari berbagai portal berita nasional, dan mencapai akurasi 84,88%, serta menggabungkan analisis kata umum dan visualisasi word cloud untuk memperkuat temuan [6]. Penelitian lain oleh Setiawan (2021) juga menggunakan Naive Bayes untuk klasifikasi berita pada media sosial Twitter dan berhasil mencapai akurasi 92% setelah preprocessing, serta 93,33% pada uji 10-fold cross-validation. Model tersebut juga telah diterapkan dalam aplikasi berbasis web [7]. Secara umum, penelitian terdahulu menunjukkan bahwa algoritma seperti Naive Bayes dan Random Forest dapat digunakan secara efektif dalam deteksi berita palsu, namun tantangan terkait ukuran dataset dan peningkatan akurasi masih menjadi ruang eksplorasi lebih lanjut.

Penelitian ini mengembangkan aplikasi web untuk mendeteksi berita palsu berbahasa Indonesia menggunakan algoritma Naive Bayes. Aplikasi ini juga diintegrasikan dengan API berita dari Google News agar pengguna dapat membandingkan informasi yang terdeteksi sebagai hoaks dengan sumber

terpercaya secara *real-time*. Diharapkan, solusi ini dapat menjadi dasar sistem deteksi hoaks yang adaptif dan berkelanjutan.

## 2. MATERIALS AND METHODS

### 2.1. Pengumpulan Data

*Dataset* berita dikumpulkan dengan metode *News Scrapper* dari situs berita seperti CNN Indonesia, Kompas, TurnBackHoax, dll. *Dataset* Kaggle yang diambil adalah *dataset* yang dikumpulkan oleh M Razif Rizqullah dan Radhinansyah Hemsah Ghaida yang terakhir diperbarui pada 2023. *Dataset* kemudian ditambah dengan data baru yang dikumpulkan dengan metode *Web Scraping* agar *dataset* mendapat pembaruan. Pada hasil akhir, *dataset* yang dikumpulkan berjumlah 19697 data dengan rincian 9999 data berlabel 0 yang menunjukkan fakta (50,7%) dan 9698 data berlabel 1 yang menunjukkan palsu (49,3%). Meski ada perbedaan rasio kelas dalam *dataset* ini, tapi perbedaan data masih dalam proporsi yang hampir sama (mendekati 50:50). Oleh karena itu, hal ini tidak akan terlalu berpengaruh terhadap model yang dihasilkan sehingga memungkinkan model belajar secara adil terhadap kedua kelas.

Berikut adalah sampel *dataset* yang telah dibersihkan :

Tabel 1 Sampel Dataset

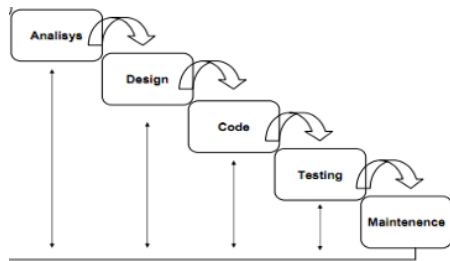
No	Narasi	Label
1.	Kiper Timnas Indonesia Ernando Ari Sutaryadi tertangkap kamera duduk santai di depan gawang saat Tim Merah Putih tengah menyerang jelang gol Ramadhan Sananta. Timnas Indonesia sukses mengalahkan Vietnam 3-0 dalam pertandingan lanjutan Grup F putaran kedua Kualifikasi Piala Dunia 2026 di Stadion My Dinh, Hanoi, Selasa (26/3) malam. Dalam pertandingan tersebut terdapat momen kiper Timnas Indonesia Ernando Ari kedatangan duduk santai ketika Tim Merah Putih sedang menyerang lewat sepak pojok.	0
2.	Lihat gambar untuk lebih jelasnya. Intinya ada foto cewek dengan puluhan hp di depannya, diklaim sebagai tukang fitnah yang menggunakan banyak akun.	1
3.	Ke Lapas! Sandra Dewi Auto Kaget, Jokowi Tindak Tegas Suami Sandra Begini & Lita Gading Beri Reaksi   TEMU HARVEY – SANDRA DEWI KAGET SANG SUAMI MAKIN KURUS DI DALAM BUI	1
4.	PP Presisi Tbk mencatatkan laba bersih konsolidasian di 2022	0

No	Narasi	Label
	sebesar Rp 182 miliar atau naik 23,7 persen dibanding tahun sebelumnya dengan pendapatan naik 29,5 persen dari Rp 2,8 triliun menjadi Rp 3,6 triliun. Perusahaan ini berhasil mendapatkan kontrak baru senilai Rp 5,2 triliun mayoritas dari jasa pertambangan 55 persen dan civil work 41 persen. PP Presisi menargetkan perolehan kontrak baru pada tahun 2023 meningkat 20 persen-30 persen. Ratio profitabilitas mencapai 2,39 persen dan ROE meningkat dari 4,9 persen menjadi 5,8 persen.	
5.	BREAKING NEWS.. Diposting oleh ; Nursyahbani Kacasungka Tgl 10 Febuari 2017 jam 22.30 Wib. YA Tuhan LINDUNGI Bangsa Ini. Beredar nama-nama anggota DPR-RI Penerima Dana Milyaran Fee E KTP. Ada nama Ahok (Basuki Tjahja Purnama) Menkumham (Yasona Laoly) Gub Jateng dan Gubernur Sulut. Berikut nama-nama yang berasal dari BAP Tersangka Irman (mantan dirjen Disdukcapil Depdagri) Fraksi Partai Demokrat	1
...	...	...
19697	Perempuan di Finlandia terlihat melakukan protes terhadap laki-laki yang hanya menikahi satu istri. Mereka ingin laki-laki menikahi lebih dari satu istri karena populasi perempuan tanpa suami terus meningkat di negara mereka	1

### 2.2. Metode Pengembangan Perangkat Lunak

Penelitian ini menggunakan pendekatan *Software Development Life Cycle* (SDLC) dengan model *Waterfall* untuk mengembangkan aplikasi web pendeteksi berita palsu berbasis web. *Waterfall* adalah pengerjaan dari suatu sistem dilakukan secara berurutan atau secara linear. Jadi jika langkah satu belum dikerjakan maka tidak akan bisa melakukan pengerjaan langkah 2, 3 dan seterusnya [8].

Langkah-langkah dalam metode *waterfall* adalah tahap analisis, desain, pengkodean, pengujian dan pemeliharaan, seperti yang diilustrasikan pada Gambar 1.



Gambar 1 Waterfall Model [8]

Berikut adalah penjelasan setiap tahapan dari metode yang digunakan:

### 2.2.1. Analisis

Tahap ini mencakup identifikasi permasalahan dalam deteksi berita palsu, pengumpulan *dataset* berita asli dan hoaks dari sumber terpercaya, serta perumusan kebutuhan fungsional dan non-fungsional sistem. Hasil analisis digunakan sebagai dasar perancangan sistem.

### 2.2.2. Desain

Pada tahap ini dirancang arsitektur sistem berbasis web, termasuk perancangan *database* dan antarmuka pengguna.

### 2.2.3 Pengkodean

Implementasi sistem dilakukan dengan bahasa pemrograman Python menggunakan *framework* Streamlit. Algoritma Naive Bayes diterapkan untuk proses klasifikasi teks berita seperti melakukan preprocessing teks dan melakukan pelatihan model. Selain itu, dilakukan integrasi dengan database agar sistem dapat menyimpan dan mengakses data secara efisien.

### 2.2.4 Pengujian

Pengujian sistem dilakukan dengan metode *blackbox testing*, yaitu pengujian berdasarkan fungsi-fungsi aplikasi tanpa melihat struktur internal kode [9]. Tujuannya adalah memastikan setiap fitur berjalan sesuai dengan spesifikasi. Evaluasi terhadap tingkat akurasi algoritma Naive Bayes juga dilakukan dengan menguji akurasi, *precision*, *recall* dan *f1-score* kemudian *confusion matrix* untuk mengukur kinerja klasifikasi.

### 2.2.5 Pemeliharaan

Tahap ini mencakup perbaikan bug, peningkatan performa sistem, serta pengembangan fitur tambahan berdasarkan umpan balik dari pengguna menggunakan metode survey pengguna.

## 3. RESULTS AND DISCUSSION

Implementasi dilakukan dengan menggunakan bahasa pemrograman Python dan *library* pendukung seperti scikit-learn, NLTK, Sastrawi, dan Streamlit. Aplikasi web ini memanfaatkan algoritma Naive Bayes dengan proses pelatihan menggunakan *dataset* lebih dari 8718 data.

### 3.1. Analisis Sistem yang Sedang Berjalan

Berdasarkan penelitian-penelitian sebelumnya, penulis memperoleh beberapa informasi yaitu proses pendeteksian berita palsu sudah memiliki aplikasi web

yang dapat mendeteksi berita palsu secara otomatis, namun masih belum efektif karena dataset yang digunakan relatif sedikit sehingga besar kemungkinan model yang dihasilkan memiliki bias ke salah satu sisi (hoaks atau fakta) dan tidak efektif dalam pendeteksian berita dengan berbagai macam gaya penulisan.

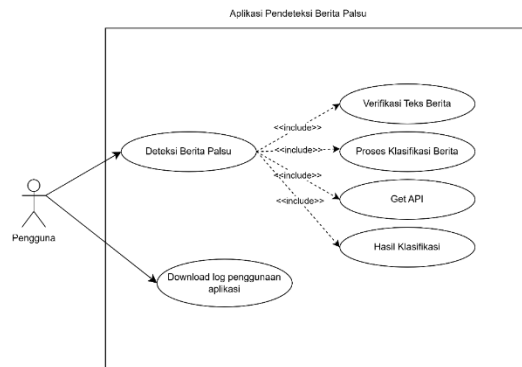
### 3.2. Analisis Sistem yang Dibangun

Sistem yang akan dibangun berupa pengembangan lebih lanjut dari aplikasi web klasifikasi berita palsu berbahasa Indonesia berbasis web dengan dataset yang lebih luas dan disertai integrasi API berita dari Google News. Selain itu, hasil penggunaan aplikasi dapat diunduh secara publik yang dapat digunakan untuk pembaruan dataset.

### 3.3. Perancangan

#### 3.3.1. Use Case Diagram

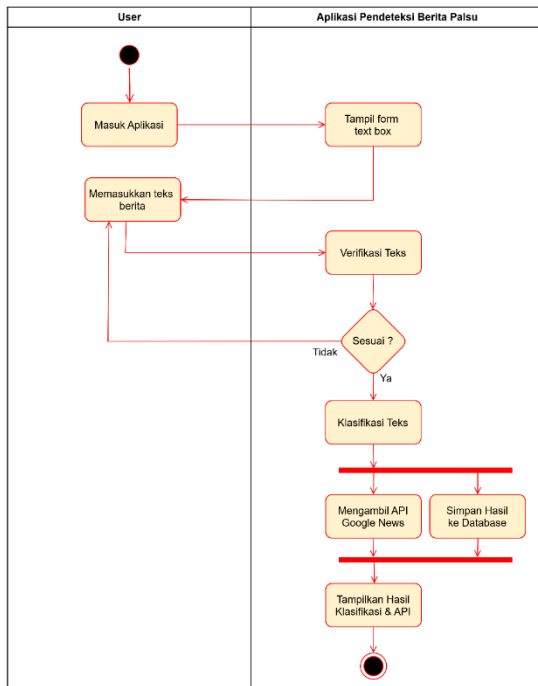
*Use case diagram* memungkinkan kita untuk menggambarkan kemungkinan skenario penggunaan (*use case*) saat suatu sistem dikembangkan [10]. Berikut adalah *use case* dari aplikasi web Klasifikasi Berita Palsu Berbahasa Indonesia :



Gambar 2 Use Case Diagram

#### 3.3.2. Activity Diagram

*Activity diagram* menggambarkan bagaimana proses aplikasi berjalan dari awal hingga akhir. *Activity Diagram* berfungsi untuk menggambarkan bagaimana suatu proses dilakukan dalam sistem, termasuk alur langkah-langkahnya (*control flow*) dan bagaimana data berpindah di antara langkah-langkah tersebut (*data flow*) [10].



Gambar 3 Activity Diagram

### 3.3.3. Perancangan Database

Aplikasi web ini hanya menggunakan 1 tabel *database*, sehingga database seperti SQLite lebih cocok untuk digunakan. SQLite adalah mesin basis data relasional yang berdiri sendiri dan tidak memerlukan konfigurasi, dirancang untuk disematkan langsung ke dalam aplikasi [11]. Berikut adalah struktur tabel *database* :

```
sqlite> .schema log
CREATE TABLE log (
  id INTEGER PRIMARY KEY AUTOINCREMENT,
  original_text TEXT,
  prediction INTEGER,
  probability REAL,
  model_used TEXT,
  timestamp DATETIME DEFAULT CURRENT_TIMESTAMP
);
```

Gambar 4 Perancangan Database

### 3.3.4. Arsitektur Sistem dan Keamanan Data

Sistem aplikasi web ini terdiri dari beberapa komponen utama yang digambarkan dalam arsitektur berikut :

- Antarmuka pengguna (UI) dibangun menggunakan Streamlit, memungkinkan pengguna menginput teks berita untuk melihat hasil klasifikasi dan berita terkait.
- Model klasifikasi menggunakan Naive Bayes yang telah dilatih dan disimpan dalam bentuk file .pkl.
- Database SQLite digunakan untuk menyimpan riwayat input dan hasil klasifikasi pengguna.
- Google News API digunakan untuk mengambil berita relevan berdasarkan input pengguna.

Aplikasi hanya menyimpan data teks berita dan hasil klasifikasinya secara lokal dan tidak mengirim

data ke server eksternal. Seluruh proses dilakukan secara lokal di sisi pengguna atau server aplikasi, sehingga menjaga privasi dan keamanan data. Selain itu, dari sisi sistem hanya menyediakan fitur untuk mengunduh riwayat penggunaan dalam bentuk csv yang dapat digunakan untuk pembaruan *dataset*.

## 3.4. Pelatihan Model

### 3.4.1. Preprocessing Data

Tahap *preprocessing data* mencakup proses tokenisasi, *stemming* dan *stopwords removal*.

```
# Cek hasilnya
print(df[['Narasi', 'text']].head())
```

Narasi	text
0 Kiper Timnas Indonesia Ernando Ari Sutaryadi t...	0 kiper timnas indonesia ernando ari sutaryadi t...
1 Lihat gambar untuk lebih jelas nya. Intinya ad...	1 lihat gambar nya inti foto cewek puluh hp depa...
2 Ke Lapas! Sandra Dewi Auto Kaget, Jokowi Tinda...	2 lapas sandra dewi auto kaget jokowi tindak sua...
3 Pelatih Portugal Roberto Martinez menjawab per...	3 latih portugal roberto martinez cristiano rona...
4 Perusahaan petrokimia asal Amerika Serikat, Ai...	4 usaha petrokimia amerika serikat air products ...

Gambar 5 Hasil Preprocessing

Gambar 5 menunjukkan teks sebelum dan sesudah proses *preprocessing*. Proses ini menghasilkan teks yang telah diubah kata-katanya ke bentuk dasarnya karena adanya proses *stemming*. Selain itu, kata-kata yang dianggap tidak penting juga dihapus melalui proses *stopwords removal*. Hasil ini membuat *dataset* menjadi lebih baik untuk proses *vectorization* karena variasi kata telah diubah ke bentuk dasar sehingga dapat mengurangi dimensi fitur dan meningkatkan akurasi model.

### 3.4.2. Vectorization TF-IDF

Tahap *vectorization* mengubah teks menjadi representasi numerik TF-IDF. Berikut adalah proses implementasi *vectorization* yang telah dilakukan dalam penelitian ini :

```
vectorizer = TfidfVectorizer(max_df=0.9, min_df=10)
X_train_vec = vectorizer.fit_transform(X_train)
X_test_vec = vectorizer.transform(X_test)
```

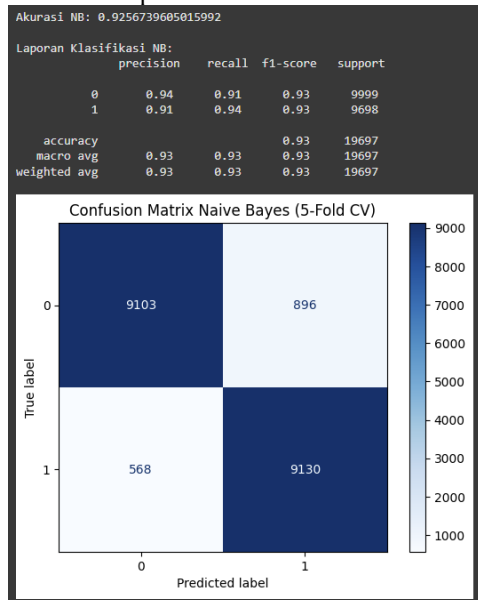
Gambar 6 Vectorization

Proses ini yang diperlukan sebelum data diproses oleh algoritma *machine learning*. *max\_df=0.9* untuk menghapus kata-kata yang terlalu sering muncul (lebih dari 90% dokumen) dan *min\_df=10* untuk mengabaikan kata-kata yang terlalu jarang (muncul di kurang dari 10 dokumen). Kemudian, *data training* ( $X_{train}$ ) diubah menjadi matriks TF-IDF menggunakan *fit\_transform*, sedangkan data uji ( $X_{test}$ ) diubah menggunakan *transform* agar tetap menggunakan fitur yang sama dengan *data training*. Hasil akhirnya adalah dua matriks vektor ( $X_{train\_vec}$  dan  $X_{test\_vec}$ ) yang siap digunakan dalam proses klasifikasi.

### 3.4.3. Performa Model

Untuk mengevaluasi performa model secara lebih menyeluruh dan menghindari kemungkinan

overfitting, maka dilakukan validasi menggunakan teknik *K-Fold Cross-Validation*. *K-Fold Cross-Validation* adalah salah satu metode statistic yang diimplementasikan untuk mengevaluasi performansi dari model atau algoritma yang telah dirancang [12]. Dalam penelitian ini digunakan *5-fold cross-validation*, yang membagi dataset menjadi 5 bagian secara acak dan bergantian menggunakan 4 bagian sebagai data pelatihan dan 1 bagian sebagai data uji. Hasil evaluasi dapat dilihat dalam rincian berikut :



Gambar 7 Akurasi Model

Berdasarkan Gambar 7, akurasi model yang dihasilkan mencapai 92,5%. Kemudian, berdasarkan *confusion matrix*, model berhasil mengklasifikasikan 9103 berita asli dan 9130 berita palsu dari dataset secara benar, dengan kesalahan yaitu 896 berita asli yang dikira palsu (*false positive*) dan 568 berita palsu yang dikira asli (*false negative*). Laporan klasifikasi juga menunjukkan bahwa nilai precision, recall, dan f1-score untuk kedua kelas (asli dan palsu) memiliki rata-rata 0,93, yang menandakan performa model sangat baik dan seimbang dalam mengenali kedua jenis berita.

Selain itu, dilakukan perbandingan akurasi model dengan algoritma lain yaitu Random Forest. Tabel berikut menunjukkan hasil evaluasi kedua algoritma berdasarkan data uji dari *dataset* dan metode yang sama :

Tabel 2 Perbandingan Algoritma

Algoritma	Akurasi	Precision	Recall	F1-Score
Naive Bayes	92,5%	0.93	0.93	0.93
Random Forest	95,6%	0.96	0.96	0.96

Hasil ini menunjukkan bahwa akurasi algoritma Random Forest sedikit lebih baik dari Naive Bayes dalam semua aspek evaluasi. Namun demikian, Naive Bayes tetap memiliki keunggulan dari sisi efisiensi

karena model ini berbasis perhitungan probabilitas sederhana yang artinya tidak ada pencarian pohon, iterasi atau *ensemble model* yang kompleks, kemudian proses ini bersifat linier terhadap jumlah fitur dan data sehingga jauh lebih ringan dibandingkan Random Forest.

### 3.5. Implementasi Aplikasi

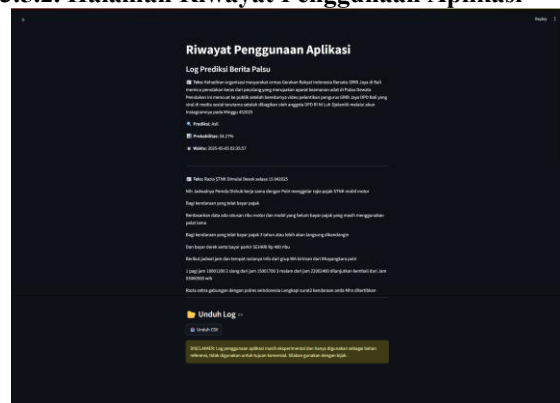
Model dan vektor TF-IDF yang telah dilatih diintegrasikan dengan *framework* Streamlit agar dapat dijalankan dalam bentuk aplikasi web. Integrasi model dengan Streamlit hanya memerlukan untuk melakukan *import file* model ke dalam aplikasi web. Kemudian aplikasi dirancang untuk dapat mengambil API dari Google News agar dapat menampilkan berita – berita terkait berdasarkan masukan dari pengguna. Berikut adalah tampilan halaman aplikasi yang telah diimplementasikan :

#### 3.5.1. Halaman Utama



Gambar 8 Halaman Utama

#### 3.5.2. Halaman Riwayat Penggunaan Aplikasi



Gambar 9 Halaman Riwayat Penggunaan Aplikasi

### 3.6. Pengujian

Metode pengujian yang digunakan dalam tahapan ini adalah *Black-Box Testing* yang menguji aplikasi web dari segi spesifikasi fungsionalitasnya, tanpa menguji kode program.

No	Kasus Uji	Masukan	Hasil yang diharapkan	Hasil yang didapat
1	Menguji form dengan gaya bahasa formal	Teks Berita Bahasa Formal	Berita berhasil dideteksi, dan menampilkan berita terkait dari API	Sesuai
2	Menguji form dengan gaya bahasa tidak formal	Teks Berita Bahasa Tidak Formal	Berita berhasil dideteksi, dan menampilkan berita terkait dari API	Sesuai
3	Menguji serangan XSS pada form pendeteksi berita	Teks sintaks HTML atau Javascript	Menampilkan peringatan elemen berbahaya	Sesuai
4	Menguji kata acak pada form pendeteksi berita	Teks acak seperti "asdasdasd"	Teks tidak diproses karena tidak bermakna	Sesuai
5	Menguji panjang kata pada form pendeteksi berita	Teks kurang dari 5 kata	Teks tidak diproses karena terlalu pendek	Sesuai
6	Menguji tombol unduh CSV	Menekan tombol unduh CSV	Mengunduh file CSV berisi riwayat log penggunaan aplikasi	Sesuai

Gambar 10 Pengujian *Blackbox*

### 3.7. Evaluasi Pengguna

Untuk menguji performa aplikasi secara real-time, dilakukan uji coba sederhana terhadap 10 pengguna dari kalangan mahasiswa. Pengguna diminta mencoba aplikasi dan mengisi form umpan balik. Hasil evaluasi menunjukkan bahwa :

- 80% pengguna menyatakan bahwa aplikasi mudah digunakan.
- 80% pengguna merasa terbantu dengan fitur berita terkait dari Google News untuk verifikasi berita.
- 70% pengguna merasa hasil klasifikasi sesuai harapan.

Terdapat beberapa saran dari pengguna meliputi penambahan pendeteksian gambar dan video hoaks/AI. Hal ini menunjukkan adanya potensi pengembangan lebih lanjut untuk meningkatkan pengalaman pengguna.

## 4. CONCLUSION

Dari hasil penelitian, aplikasi web untuk mendeteksi berita palsu berbahasa Indonesia menggunakan algoritma Naive Bayes telah berhasil dibuat dan dikembangkan. Aplikasi ini mampu mengklasifikasikan berita menjadi fakta atau palsu secara otomatis. Model yang digunakan menunjukkan performa yang baik dengan akurasi sebesar 92,5% berdasarkan pengujian dengan *K-Fold Cross Validation*, yang mencerminkan kemampuannya dalam menangani berbagai gaya bahasa dalam teks berita.

Aplikasi ini dirancang dengan antarmuka yang sederhana dan mudah dipahami oleh pengguna. Selain menampilkan hasil klasifikasi, sistem juga menyediakan fitur pencarian berita terkait melalui integrasi dengan API Google News, sehingga pengguna dapat langsung membandingkan informasi yang ditemukan. Hal ini mendukung tujuan utama sistem sebagai alat bantu verifikasi berita secara cepat dan efisien.

## REFERENCES

[1] L. Rizkinaswara, "Dua Upaya Kominfo Atasi Hoaks Vaksinasi Covid-19," Kominfo. Diakses: 4 Maret 2025. [Daring]. Tersedia pada: [https://aptika.kominfo.go.id/2021/02/dua-](https://aptika.kominfo.go.id/2021/02/dua-upaya-kominfo-atasi-hoaks-vaksinasi-covid-19/)

[upaya-kominfo-atasi-hoaks-vaksinasi-covid-19/](https://aptika.kominfo.go.id/2021/02/dua-upaya-kominfo-atasi-hoaks-vaksinasi-covid-19/)

[2] Siaran Pers, "Siaran Pers No. 02/HM/KOMINFO/01/2024 tentang Hingga Akhir Tahun 2023, Kominfo Tangani 12.547 Isu Hoaks," Komdigi. Diakses: 22 Desember 2024. [Daring]. Tersedia pada: <https://www.komdigi.go.id/berita/pengumuman/detail/siaran-pers-no-02-hm-kominfo-01-2024-tentang-hingga-akhir-tahun-2023-kominfo-tangani-12-547-isu-hoaks>

[3] A. Géron, *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow Concepts, Tools, and Techniques to Build Intelligent Systems*, 2 ed. Sebastopol: O'Reilly Media, Inc., 2019.

[4] L. Rokach dan O. Maimon, *Data Mining with Decision Tree*, 2 ed., vol. 81. Singapura: World Scientific Publishing Co. Pte. Ltd., 2015.

[5] N. Agustina, A. Adrian, dan M. Hermawati, "Implementasi Algoritma Naive Bayes Classifier untuk Mendeteksi Berita Palsu pada Sosial Media," *Faktor Exacta*, vol. 14, no. 4, hlm. 206, Jan 2022, doi: 10.30998/faktorexacta.v14i4.11259.

[6] A. S. Nurhikam dkk., "Deteksi Berita Palsu Pada Pemilu 2024 Dengan Menggunakan Algoritma Random Forest," *DoubleClick: Journal of Computer and Information Technology*, vol. 7, no. 1, hlm. 41–50, Agu 2023.

[7] E. I. Setiawan, S. Johanes, A. T. Hermawan, dan Y. Yamasari, "Deteksi Validitas Berita pada Media Sosial Twitter dengan Algoritma Naive Bayes," *Journal of Intelligent System and Computation*, vol. 3, no. 2, hlm. 55–60, Okt 2021, doi: 10.52985/insyst.v3i2.164.

[8] B. Hartono, *Cara Mudah dan Cepat Belajar Pengembangan Sistem Informasi*. Semarang: Yayasan Prima Agus Teknik, 2021.

[9] M. Syarif dan E. B. Pratama, "Analisis Metode Pengujian Perangkat Lunak Blackbox Testing dan Pemodelan Diagram UML pada Aplikasi Veterinary Services yang Dikembangkan dengan Model Waterfall," *Jurnal Teknik Informatika Kaputama (JTIK)*, vol. 5, no. 2, 2021.

[10] M. Seidl, M. Scholz, C. Huemer, dan G. Kappel, *UML @ Classroom*. dalam Undergraduate Topics in Computer Science. Cham: Springer International Publishing, 2015.

[11] J. A. Kreibich, *Using SQLite*, 1 ed. O'Reilly Media, Inc., 2010.

[12] Y. N. Fuadah, I. D. Ubaidullah, N. Ibrahim, F. F. Taliningsing, N. K. Sy, dan M. A. Pramuditho, "Optimasi Convolutional Neural Network dan K-Fold Cross Validation pada

Sistem Klasifikasi Glaukoma,” *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, vol. 10, no. 3, hlm. 728, Jul 2022, doi: 10.26760/elkomika.v10i3.728.