

SISTEM DETEKSI BERITA *HOAX* MENGGUNAKAN *PLATFROM MOBILE PHONE*

Fakhruz Zumara¹, Depandi Enda²

^{1,2}Politeknik Negeri Bengkalis

Email: fakhruzzumara41@gmail.com¹, depandienda@polbeng.ac.id²

Abstrak: Penyebaran berita hoaks di era digital menjadi tantangan serius yang berdampak pada kepercayaan publik terhadap informasi daring. Penelitian ini bertujuan mengembangkan sistem deteksi berita hoaks berbasis aplikasi *mobile* menggunakan *Flutter* yang terintegrasi dengan API *Flask*. Sistem ini mengadopsi metode *machine learning* dengan algoritma *Support Vector Machine* (SVM) untuk mengklasifikasikan berita ke dalam kategori hoaks atau fakta. Data dikumpulkan melalui teknik web scraping dari berbagai portal berita nasional dan situs verifikasi fakta, kemudian dibagi menjadi data latih dan data uji. Proses pelatihan model menghasilkan tingkat akurasi 0.9648–0.9805, presisi 0.9640–0.9800, recall mencapai 1.000, *F1-Score* antara 0.9817–0.9899, dan ROC-AUC antara 0.9961–0.9994. Hasil menunjukkan kinerja klasifikasi yang andal dan konsisten. Aplikasi ini menyediakan fitur pencarian berita berdasarkan kata kunci, penampilan detail berita, serta hasil deteksi secara *real-time* melalui antarmuka yang responsif. Hasil integrasi antara *frontend* dan *backend* menunjukkan bahwa sistem mampu bekerja dengan baik dan memberikan respon cepat. Dengan demikian, aplikasi ini dapat menjadi solusi efektif untuk membangun pengguna dalam memverifikasi informasi secara praktis, cepat, dan akurat melalui satu *platfor* terpadu.

Kata Kunci: Deteksi Berita Hoaks, Aplikasi Mobile Flutter, Machine Learning, Support Vector Machine (SVM), Web Scraping.

Abstract: The spread of hoax news in the digital era has become a serious challenge that affects public trust in online information. This study aims to develop a mobile-based hoax news detection system using Flutter integrated with a Flask API. The system adopts a machine learning approach with the Support Vector Machine (SVM) algorithm to classify news articles into hoax or factual categories. Data were collected through web scraping techniques from various national news portals and fact-checking sites, then divided into training and testing datasets. The model training process produced high accuracy ranging from 0.9648 to 0.9805, precision between 0.9640 and 0.9800, perfect recall of 1.0000, F1-Score between 0.9817 and 0.9899, and ROC-AUC from 0.9961 to 0.9994. Indicating reliable and consistent classification performance. The application provides features such as keyword-based news search, detailed news display, and real-time detection results through a responsive interface. Integration between the frontend and backend demonstrated that the system functions properly and responds quickly. Therefore, this application offers an effective solution to assist users in verifying information in a practical, fast, and accurate manner through a unified platform.

Keywords: Hoax News Detection, Flutter Mobile Application, Machine Learning, Support Vector Machine, Web Scraping.

PENDAHULUAN

Hoax adalah informasi atau berita yang belum terverifikasi atau bukan fakta yang sebenarnya. Penyebaran berita *hoax* menjadi tantangan terbesar dalam ekosistem informasi. Berita dapat dengan mudah diakses. Tetapi dengan banyaknya berita yang tersedia, sulit untuk memvalidasi berita fakta dan berita *hoax*. Berita *hoax* menjadi masalah besar atau masalah serius karena berita *hoax* dapat menyebabkan kebingungan dan kerugian. Berita Hoax merupakan kumpulan kata atau kalimat yang mengandung informasi yang tidak benar yang berupaya untuk membohongi atau mengarahkan pembaca agar mendukung atau percaya dengan isi beritanya. Penyebaran berita *hoax* umumnya mengetahui bahwa berita yang disebarkan tidak benar. Berita *hoax* masih tersebar luas dan memiliki resiko yang besar untuk memanipulasi pola pikir. Terutama disitus publik atau pemerintah sangat penting. Namun, dalam penerapannya penting juga untuk menemukan metode terbaik dalam menentukan apakah berita itu *hoax* atau fakta. (enam)

Banyaknya informasi yang tersebar, sulit untuk membedakan antara berita *hoax* dan fakta. Berita *hoax* seringkali memanfaatkan judul sensasional untuk menarik perhatian, meskipun informasi yang disampaikan tidak benar. Oleh karena itu, permasalahan yang intin dipecahkan adalah bagaimana membangun aplikasi yang dapat mengidentifikasi berita *hoax* dan fakta dengan menggunakan algoritma *Support Vectir Machine* (SVM). Sistem yang dikembangkan akan memproses teks berita untuk melakukan klasifikasi apakah berita tersebut termasuk berita *hoax* atau fakta.

Dalam penelitian ini akan membangun sistem deteksi berita *hoax*, penelitian ini berfokus mebangun aplikasi yang mampu memvverifikasi berita fakta dan *hoax* dengan menggunakan metode *Machine Learning* algoritma *Support Vector Machine*. Sumber data yang digunakan dalam penelitian ini berasal dari berbagai portal berita seperti Detik.com, Liputan.com CNNIndonesia.com, Riaupos.jawapos.com, Riauterkini.com, prokopim.bengkaliskab.go.id, Pikiran-rakyat.com, idntimes.com dan situs veritfikasi fakta seperti *Trunbackhoax.id*, dan *Cekfakta.com*. Data dikumpulkan melalui teknik *Web Scrapping*, kemudian diklasifikasikan menggunakan alogrita *Support Vector Machine* untuk membedakan antara berita fakta dan berita *hoax*. Hasil dari penelitian ini berupa sebuah aplikasi deteksi berita dan menyediakan berita dari berbagai sumber di dalam satu aplikasi.

Beberapa penelitian yang terkait yakin tentang data mining, khususnya penggunaan metode *Support Vector Machine* untuk mendeteksi berita *hoax* dari berita-berita Indonesia.

Tujuan dari penelitian ini adalah untuk mengimplementasikan algoritma mesin *virtual* pendukung pendeteksi berita palsu pada berita Indonesia dan mengetahui tingkat akurasi algoritma *Support Vectro Machine* dalam mendeteksi berita palsu pada berita Indonesia. [4] Penelitian ini berfokus pada pengembangan dan pengujian model klasifikasi yang dapat membedakan berita palsu dan asli. Tujuannya adalah untuk mengembangkan model klasifikasi yang dapat mendeteksi berita *hoax* dengan akurasi yang tinggi, dan pengguna dapat menggunakan model ini untuk secara langsung memprediksi kebenaran berita melalui antarmuka yang disediakan. Pengembangan dan implementasi sistem untuk mendeteksi penyebaran berita palsu di media sosial. Mendeteksi berita palsu yang beredar di media sosial dengan menggunakan Algoritma *Naïve Bayes* untuk klasifikasi berita palsu. Adapun implementasi metode klasifikasi berita *hoax* terkait pemilu Presiden Indonesia 2024 menggunakan dua Algoritma yaitu *Naïve Bayes* dan *Support Vector Machine* dan membandingkan kedua algoritma dalam mengimplementasikan metode klasifikasi berita *hoax* terkait Pilpres 2024. Pengembangan dan analisi model klasifikasi berita *hoax* terkait Covid-19 menggunakan algoritma SVM untuk mengklasifikasikan berita menjadi hoaks dan bukan hoaks berdasarkan sumbernya.

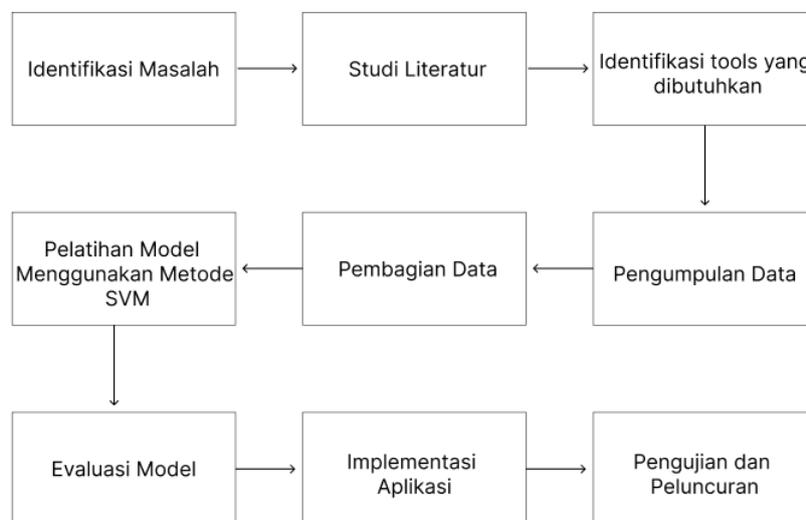
Dari beberapa penelitian terkait tersebut, dapat disimpulkan bahwa untuk melakukan deteksi dibutuhkan klasifikasi data latih yang berfungsi untuk *input* data sesuai kategori yang telah ditentukan, yaitu *hoax* dan fakta. Berbagai metode yang bisa digunakan untuk klasifikasi data latih, tetapi rata-rata penelitian menggunakan metode *Naïve Bayes* untuk melakukan klasifikasi data latih. Selain itu beberapa penelitian juga melakukan perbandingan dengan menggunakan beberapa metode untuk mengetahui hasil klasifikasi yang maksimal dalam mengklasifikasi sebuah data latih yang dimana data tersebut akan digunakan untuk melakukan deteksi. Sementara itu beberapa penelitian terdahulu hanya menggunakan satu atau dua sumber berita. Berbeda dengan penelitian-penelitian tersebut, penelitian ini tidak hanya mengandalkan dari satu atau dua sumber, tetapi menggunakan berbagai sumber berita terpercaya, hal ini akan membuat model bisa mempelajari perbedaan teks dari berita *hoax* dan fakta. Selain itu penelitian ini berfokus pada penggunaan algoritma *Support Vector Machine* dalam klasifikasi data latih untuk mengklasifikasi data sesuai kategori yang sudah ditentukan.

Berdasarkan latar belakang diatas, dalam penelitian ini merancang dan membangun sebuah sistem untuk mendeteksi berita *hoax* atau fakta. Penelitian ini berfokus membangun

aplikasi yang mampu memverifikasi berita *hoax* dan fakta dengan menggunakan metode *Machine Learning* algoritma *Support Vector Machine*. Sumber data yang digunakan dalam penelitian ini berasal dari berbagai portal berita yang terpercaya. Sehingga dengan adanya penelitian ini dapat mempercepat proses verifikasi berita dan menyediakan berbagai sumber berita di dalam satu aplikasi. Hasil dari pengujian penelitian ini menunjukkan bahwa fungsi utama aplikasi, yaitu *searchNews()*, *fetchNewsDetail()*, dan *detectHoax()* berhasil diimplementasikan dengan baik. Aplikasi ini mampu menampilkan hasil pencarian berita sesuai kata kunci, memberikan detail berita yang lengkap dan menampilkan hasil deteksi dengan tingkat akurasi yang memandai. Aplikasi ini menyediakan antarmuka yang dilengkapi fitur *Scrolling*, *pop-up* hasil deteksi.

METODE PENELITIAN

Secara garis besar, gambaran umum dari tahapan penelitian pada proses membuat sistem deteksi berita menggunakan metode *machine learning* algoritma *support vector machine* untuk klasifikasi data sesuai kategori yang telah ditentukan terdapat pada gambar blok alur penelitian berikut.



Gambar 1 Alur penelitian

Gambar 1 merupakan alur dari penelitian ini untuk mencapai tujuan penelitian. Berikut adalah alur penelitian yang meliputi:

1. Identifikasi Masalah: Melakukan identifikasi masalah dari objek penelitian serta menentukan batasan-batasan masalah, sehingga dapat menyelesaikan masalah yang ada dengan arah yang jelas.

2. Studi Literatur: Mengumpulkan penelitian terdahulu untuk dijadikan *revrensi* yang berkaitan dengan penelitian ini untuk membantu dalam penyelesaian masalah. Informasi ini membangun landasan yang kuat untuk penelitian ini.
3. Identifikasi *Tools* yang Dibutuhkan: Melakukan persiapan *tools* yang dibutuhkan
4. Pengumpulan Data: Mengumpulkan data berita dari berbagai sumber untuk melatih model sesuai kategori yang telah ditentukan
5. Pembagian Data: Data yang telah dikumpulkan akan dibagi menjadi 2, yaitu data latih dan data uji. Data latih di bagi menjadi 80% dan data uji di bagi menjadi 20%
6. Pelatihan Model Menggunakan Metode SVM: SVM berkerja mencari *hyperlane* terbaik yang memisahkan data dari kelas atau kategori berbeda dalam ruang fitur
7. Implementasi Aplikasi: Menentukan beberapa hal yang harus dilakukan dalam membangun sebuah aplikasi.
8. Pengujian dan Peluncuran: Pengujian model menggunakan *accuracy*, *precision*, *recall*, *F1-score* dan ROC-AUC. Setelah pengujian model, maka model bisa digunakan pada aplikasi untuk mendeteksi berita dengan tingkat akurat yang tinggi.

Pengumpulan Data

Pengumpulan data bertujuan untuk mendapatkan informasi yang akurat dan relevan yang akan menjadi dasar bagi proses klasifikasi. Data yang dikumpulkan harus mencakup berbagai aspek yang sesuai dengan tujuan klasifikasi. Pada tahapan ini, dilakukan pengumpulan data untuk melatih model dalam bentuk teks, serta melatih model untuk menyesuaikan teks dalam kategori yang sudah ditentukan, yaitu *hoax* (0) dan fakta (1). Pengumpulan data dilakukan dengan teknik *Web Scraping*. *Web Scraping* adalah teknik untuk mengekstraksi data dari internet dan *web scraping* sebagai teknik yang efisien dan kuat untuk mengumpulkan big data lalu disimpan ke file atau database untuk kebutuhan analisis data.[8] Jenis *web scraping* yang digunakan adalah *Selenium Webdriver*. *Selenium Webdriver* merupakan salah satu versi yang dapat digunakan untuk melakukan *web scraping* dikarenakan memiliki kemampuan untuk otomatisasi web dan berinteraksi dengan elemen web. *Selenium Webdriver* saat ini mendukung sebagian besar browser populer. Pada proses *web scraping* dilakukan dengan identifikasi url web[9] dan elemen HTML. Elemen html yang diambil menyesuaikan kebutuhan seperti, judul, tanggal, isi dari konten. Tabel 1 adalah sample data yang sudah dikumpulkan melalui teknik *web scraping*

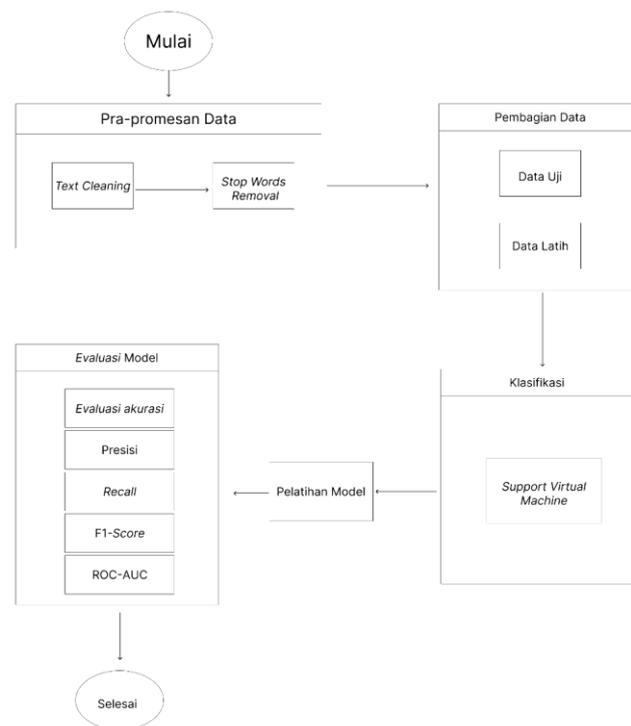
Table 1 Sample pengumpulan data

No	Judul	Tanggal	Narasi	Label	Sumber
1	Artificial Intelligence dan Otoritarianisme Baru	Rabu, 05 Feb 2025 13:40	artificial intelligence ai pencapaian terbesar umat manusia abad 21 ai diartikan kecerdasan buatan..	1	Newsdetik.com
2	Panduan Doa Nisfu Syaban, Pilih Tempat yang Tenang dan Nyaman	13 Februari 2025, 14:38 WIB	Malam Nisfu Syaban merupakan malam yang istimewa bagi umat Islam...	1	Pikiranrakyat.com
3	Akun Facebook “mr.terimakasih berbagi Indonesia”	Januari 6, 2025	akun facebook mr.terimakasih berbagi indonesia 30 desember 2024 mengunggah foto arsip disukai...	0	Trunbackhoax.com
4	Erling Haaland Bakal Perkuat Timnas Indonesia	2025-02- 10	Berita Kanal YouTube “Sport Starting” pada Minggu (19/01/2025) mengunggah video [arsip] disertai narasi: ” DIRESTUI KELUARGA HAALAND SEMAKIN MANTAP DENGAN KEPUTUSANNYA PERKUAT TIMNAS” ...	0	Cekfakta.com

Dataset akan dibagi menjadi dua yaitu data latih dan data uji. Data di bagi menjadi 80% data latih dan 20% data uji. Data latih adalah Kumpulan data yang digunakan untuk melatih model *meachine learning*. Model mempelajari hubungan antara *input* dan *output* dari data. Tujuan utama data latih adalah untuk memungkinkan model mengenali pola dan korelasi dalam data, sehingga dapat membuat prediksi dengan akurat. Data uji bertujuan untuk menguji metode klasifikasi SVM dengan memasukkan data baru , lalu metode tersebut akan melakukan klasifikasi yang tepat terhadap data baru.

Pelatihan model menggunakan algoritma SVM

Pada tahapan ini, SVM bekerja dengan mencari *hyperline* terbaik yang memisahkan data dari kelas yang berbeda dalam ruang fitur. Ada beberapa langkah langkah yang harus dilakukan dalam tahapan ini. Berikut adalah langkah-langkah dalam pelatihan model menggunakan algoritma SVM:

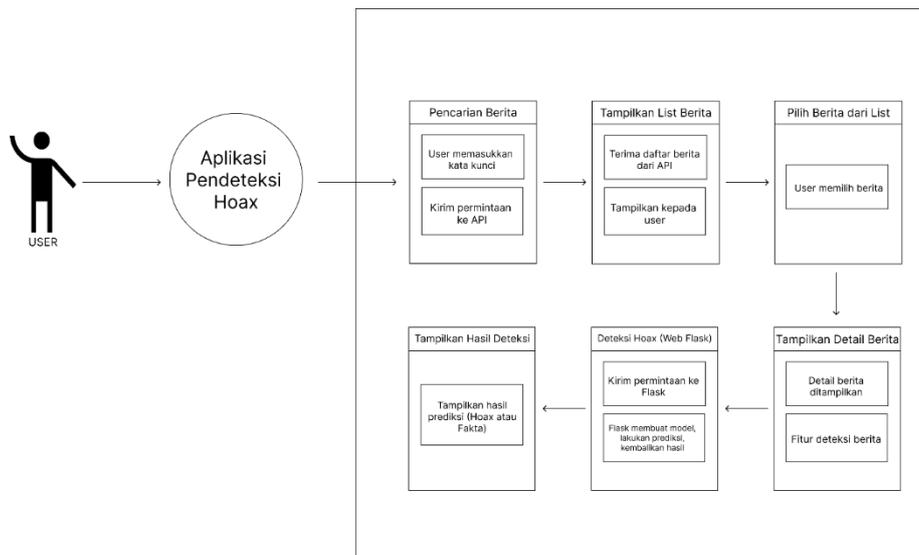


Gambar.2 Tahapan Pelatihan Model

Implementasi Aplikasi

Tahapan ini, harus ada beberapa yang harus dilakukan. Seperti:

- Pemilihan *platform*, pada penelitian ini memilih *platform mobile*
- Backend*, menggunakan *webflask* sebagai *backend* untuk memproses teks dan membuat model klasifikasi menggunakan algoritma SVM
- Frontend*, merancang dan membangun antar muka menggunakan *Flutter* dengan menggunakan bahasa pemrograman dart.
- Integrasi, integrasikan *backend* dan *frontend* sehingga *input* dari user dapat dikirim ke *backend*, *backend* akan memproses model SVM dan *backend* akan mengirimkan *output* ke user melalui *frontend*.
- Arsitektur desain penggunaan aplikasi.



Gambar 3 Arsitektur desain

HASIL DAN PEMBAHASAN

Hasil Percobaan Integrasi *Backend* dengan *Frontend*

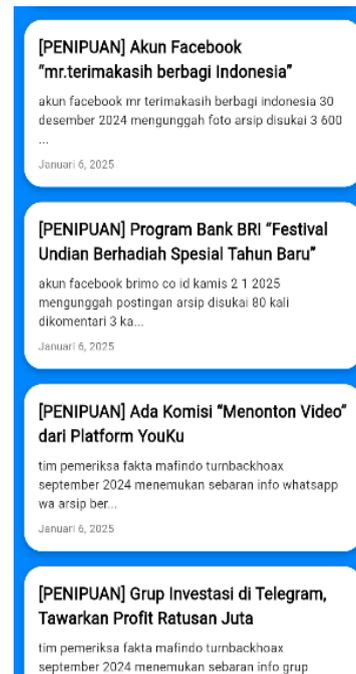
Integrasi *backend dengan frontend* berhasil memanfaatkan API *flask* secara optimal dan mendukung fitur utama dari aplikasi ini, yaitu pencarian berita berdasarkan kata kunci “*searchNews()*”, menampilkan detail berita “*fetchNewsDetail()*” dan mendeteksi berita fakta atau *hoax* “*detectHoax()*”.

Hasil percobaan Fitur *searchNews()*

Fitur *searchNews()* ini menggunakan *endpoint /search*, *endpoint /search* menerima parameter berupa kata kunci dan mengembalikan berupa daftar berita yang ada pada dataset. *Endpoint /search* bertanggung jawab untuk mencari berita dari dataset berdasarkan kata kunci yang diberikan user. *Output* yang dikembalikan API *Flask* kemudian ditampilkan pada tampilan aplikasi.



Gambar 4 Daftar berita (fakta)

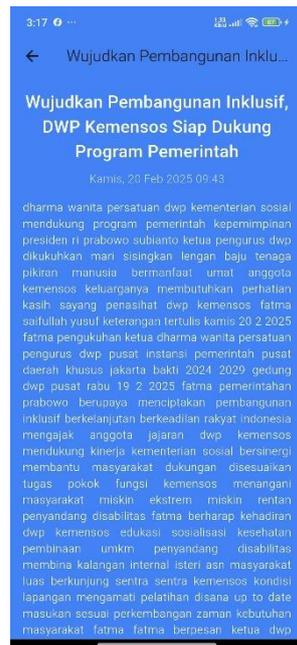


Gambar 5 Daftar berita (hoax)

Berdasarkan gambar 4 dan gambar 5, fungsi dari *searchNews()* bekerja dengan baik, API Flask mengembalikan daftar berita yang berisi informasi judul, deskripsi singkat, tanggal, hari dan sumber beritanya. Daftar berita yang ditampilkan sesuai dengan kata kunci yang dimasukkan pengguna dan respon yang diterima aplikasi *flutter* berupa data JSON di ubah kedalam format yang dapat ditampilkan di antar muka aplikasi.

Hasil Percobaan Fitur *fetchNewsDetail()*

Fitur *fetchNewsDetail()* ini menenggunakan *endpoint /news/id* agar API *flask* menyediakan data detail berita berdasarkan ID dari judul berita yang ditentukan user. Fungsi *fetchNewsDetail()* bekerja untuk melengkapi fitur *searchNews()*, *detectHoax()* dan menampilkan detail berita pada tampilan aplikasi.



Gambar 6 Detail berita (fakta) Gambar 7 Detail berita (hoax)

Berdasarkan gambar 6 dan gambar 7, data berita yang diterima dari API akan digunakan untuk ditampilkan di halaman detail berita dalam aplikasi. Fungsi *fetchNewsDetail()* berkerja untuk melengkapi fungsi *searchNews()* dan *detectHoax()*. Detail berita yang ditampilkan berdasarkan ID berita yang dipilih dan dikirimkan ke fungsi *fetchNewsDetail()* yang memanggil API *Flask* pada *endpoint /news/id*. Pada halaman ini juga akan menampilkan tautan dari sumber berita.

Hasil Percobaan Fitur *detectHoax()*

Fitur *detectHoax()* menggunakan *endpoint /detect* untuk mendeteksi sebuah berita tersebut termasuk didalam kategori *hoax* atau fakta. Aplikasi *flutter* mengirimkan permintaan HTTP ke API *flask* menggunakan *package http* untuk melakukan deteksi berita. API memproses data menggunakan algoritma dan model yang telah diintegrasikan. Respon yang diterima aplikasi berupa data JSON diubah ke dalam format yang dapat ditampilkan di antarmuka aplikasi dalam bentuk *pop-up*.



Gambar 8 Hasil deteksi



Gambar 9 Hasil deteksi

Berdasarkan gambar 8 dan gambar 9, hasil deteksi sesuai dengan kategori berita yang ada didalam data. Aplikasi mengirimkan permintaan HTTP ke API Flask menggunakan *package* ‘http’ untuk melakukan pencarian berita atau deteksi berita. API Flask memproses data menggunakan algoritma dan model yang telah diintegrasikan, respon yang diterima aplikasi berupa data *JSON* di ubah ke dalam format yang dapat ditampilkan di antar muka aplikasi.

Hasil Pengujian Model Klasifikasi

Pengujian model bertujuan untuk mengetahui seberapa akurat model dalam mendeteksi, pengujian dilakukan dengan klasifikasi menggunakan algoritma *Support Vector Machine*, pengujian terdiri dari akurasi, presisi, recall, dan *F1-Score* dan dibantu dengan *ROC-AUC* dan *Cross-Validation*. *ROC-AUC* memberikan pemahaman lebih terhadap model dalam membedakan antar kelas positif dan negative pada berbagai ambang batas, *ROC-AUC* yang lebih tinggi menunjukkan model lebih baik dalam melakukan klasifikasi. Sedangkan *Cross-Validation* berguna untuk menghindari *overfitting* dengan melatih dan menguji model pada berbagai subset data. Pengujian model dilakukan sebanyak 10 kali untuk melihat nilai dari pengujian model.

Table 2 Hasil Pengujian Model

No	Akurasi	Presisi	Recall	F1-Score	ROC-AUC	Mean Accuracy (Cross-Validation)
1	0.9648	0.9640	1.000	0.9817	0.9991	0.9954
2	0.9743	0.9737	1.000	0.9867	0.9994	0.9931
3	0.9743	0.9737	1.000	0.9867	0.9975	0.9949
4	0.9766	0.9758	1.000	0.9878	0.9985	0.9955
5	0.9797	0.9791	1.000	0.9895	0.9962	0.9953
6	0.9804	0.9799	1.000	0.9898	0.9961	0.9958
7	0.9804	0.9798	1.000	0.9898	0.9963	0.9955
8	0.9804	0.9799	1.000	0.9898	0.9966	0.9958

9	0.9805	0.9800	1.000	0.9899	0.9963	0.9947
10	0.9776	0.9770	1.000	0.9884	0.9971	0.9959

Berdasarkan tabel 2. Pengujian model dilakukan sebanyak 10 kali dengan hasil evaluasi berdasarkan beberapa metrik utama. Berikut adalah analisis terhadap hasil pengujian

a. Akurasi

Akurasi model berkisaran 0.9648 hingga 0.9805, dengan nilai tertinggi 0.9805 dan nilai terendah 0.9648. Nilai akurasi yang tinggi menunjukkan bahwa model mampu mengklasifikasikan berita *hoax* dan fakta dengan baik

b. Presisi

Presisi model berkisaran 0.9640 hingga 0.9800, yang menunjukkan bahwa model mampu mengidentifikasi berita *hoax* dan fakta dengan tingkat kesalahan yang rendah. Presisi yang tinggi berarti model jarang memberikan *false positive*, artinya ketika model memprediksi sebuah berita sebagai *hoax* atau fakta, kemungkinan besar berita tersebut memang *hoax* atau fakta

c. Recall

Nilai recall pada 10 kali pengujian pertama mencapai 1.000, yang berarti model berhasil mendeteksi berita tanpa ada yang terlewat. Nilai recall yang sempurna mengidentifikasikan bahwa model sangat sensitif terhadap berita *hoax* dan tidak melewatkan satupun berita *hoax* dalam dataset pengujian

d. F1-Score

F1-Score berkisaran antara 0.9817 hingga 0.9899, yang menunjukkan keseimbangan antara presisi dan recall. Nilai yang tinggi menunjukkan bahwa model memiliki performa yang sangat baik dalam mengklasifikasikan berita *hoax* dan fakta secara seimbang

e. ROC-AUC

ROC-AUC model berada dalam kisaran 0.9961 hingga 0.9994, yang hampir mendekati 1.0, menunjukkan bahwa model memiliki kemampuan klasifikasi yang sangat baik. ROC-AUC yang tinggi berarti model dapat dengan baik membedakan antara berita *hoax* dan fakta.

f. Mean Accuracy (Cross-Validation)

Hasil uji Cross-Validation menunjukkan nilai rata-rata akurasi antara 0.9931 hingga 0.9959, yang memperkuat bahwa model memiliki generalizability yang baik. Mean

accuracy yang tinggi menandakan bahwa model tetap konsisten dalam berbagai subset data pelatihan dan pengujian.

KESIMPULAN DAN SARAN

Berdasarkan hasil pengujian dan analisis, aplikasi yang dikembangkan berhasil mengintegrasikan API berbasis Flask dengan Flutter, memungkinkan pengguna untuk mencari berita, melihat detail berita, serta mendeteksi hoaks secara efektif. Fungsi utama seperti pencarian berita, penampilan detail berita, dan klasifikasi hoaks/fakta bekerja dengan baik, didukung oleh antarmuka yang intuitif serta respon API yang cepat dan akurat. Evaluasi performa model menunjukkan hasil yang sangat baik, dengan akurasi mencapai 0.9805, presisi tinggi 0.9800, recall sempurna 1.000, *F1-Score* hingga 0.9899 dan ROC-AUC mendekati 1.0, yang mengindikasikan bahwa sistem memiliki kemampuan klasifikasi yang sangat handal. Untuk pengembangan lebih lanjut, disarankan agar model *machine learning* dilatih dengan dataset yang lebih besar dan beragam, serta menambahkan fitur seperti *bookmark*, riwayat pencarian, dan notifikasi berita terbaru. Selain itu, pengguna layanan *cloud* untuk hosting API, penerapan autentikasi pengguna dan enkripsi data juga penting untuk meningkatkan performa, stabilitas, serta keamanan aplikasi kedepannya.

DAFTAR PUSTAKA

- Nur Elyta Febriyanty “*DETEKSI BERITA HOAX DARI MEDIA ONLINE INDONESIA MENGGUNAKAN ALGORITMA NAIVE BAYES DAN SUPPORT VECTOR MACHINE*” 2023.
- E. I. Setiawan, S. Johanes, A. T. Hermawan, and Y. Yamasari, “*Deteksi Validitas Berita pada Media Sosial Twitter dengan Algoritma Naive Bayes,*” *Journal of Intelligent System and Computation*, vol. 3, no. 2, pp. 55–60, 2021, doi: 10.52985/insyst.v3i2.164.
- H. Mustofa and A. A. Mahfudh, “*Klasifikasi Berita Hoax Dengan Menggunakan Metode Naive Bayes,*” *Walisongo Journal of Information Technology*, vol. 1, no. 1, p. 1, 2019, doi: 10.21580/wjit.2019.1.1.3915.
- T. Mulyana, “*Deteksi Hoax Pada Media Berita Indonesia Menggunakan Metode Support Vector Machine Skripsi,*” pp. 1–14, 2023.
- N. Agustina, A. Adrian, and M. Hermawati, “*Implementasi Algoritma Naive Bayes Classifier untuk Mendeteksi Berita Palsu pada Sosial Media,*” *Faktor Exacta*, vol. 14, no. 4, p. 206, 2022, doi: 10.30998/faktorexacta.v14i4.11259

- B. Imran, M. N. Karim, and N. I. Ningsih, “Klasifikasi Berita Hoax Terkait Pemilihan Umum Presiden Republik Indonesia Tahun 2024 Menggunakan Naïve Bayes Dan Svm,” *Dinamika Rekayasa*, vol. 20, no. 1, pp. 1–9, 2024, doi: 10.20884/1.dinarek.2024.20.1.27.
- I. A. Ropikoh, R. Abdulhakim, U. Enri, and N. Sulistiyowati, “Penerapan Algoritma Support Vector Machine (SVM) untuk Klasifikasi Berita Hoax Covid-19,” *Journal of Applied Informatics and Computing*, vol. 5, no. 1, pp. 64–73, 2021, doi: 10.30871/jaic.v5i1.3167.
- B. Pendidikan, D. Pelatihan Keuangan, K. Keuangan, M. Djufri, dan P. Pajak, “JURNAL BPPK PENERAPAN TEKNIK WEB SCRAPING UNTUK PENGGALIAN POTENSI PAJAK (Studi Kasus pada Online Market Place Tokopedia, Shopee dan Bukalapak).”
- A. Z. Rizquina dan C. I. Ratnasari, “Implementasi Web Scraping untuk Pengambilan Data Pada Website E-Commerce,” *J. Teknol. Dan Sist. Inf. Bisnis*, vol. 5, no. 4, hal. 377–383, 2023, doi: 10.47233/jteksis.v5i4.913.
- S. Rabbani, D. Safitri, N. Rahmadhani, A. A. F. Sani, dan M. K. Anam, “Perbandingan Evaluasi Kernel SVM untuk Klasifikasi Sentimen dalam Analisis Kenaikan Harga BBM,” *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 3, no. 2, hal. 153–160, 2023, doi: 10.57152/malcom.v3i2.897.