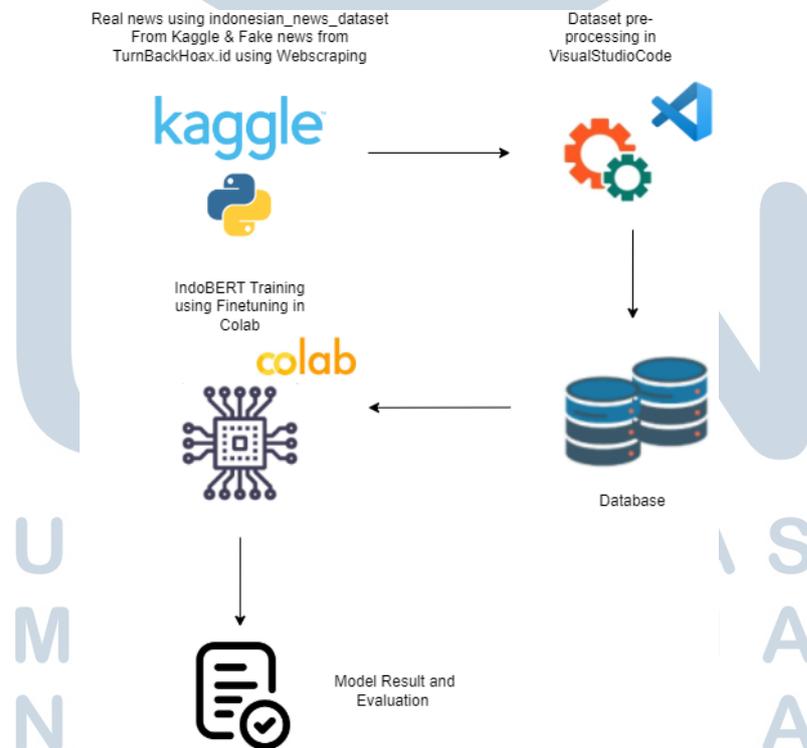


BAB 3 METODOLOGI PENELITIAN

3.1 Desain Penelitian Diteksi Berita Hoaks

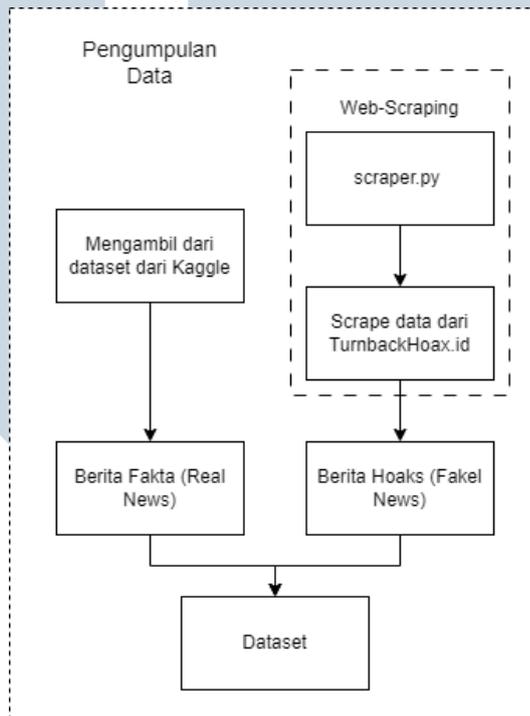
Penelitian ini dilaksanakan melalui serangkaian tahapan yang sistematis, seperti yang digambarkan pada Gambar 3.1. Alur penelitian diawali dengan Pengumpulan Data dari berbagai sumber. Data yang terkumpul kemudian memasuki tahap *pre-processing* data untuk pembersihan dan standardisasi. Setelah bersih, data dibagi (*split*) menjadi data latih (80%) dan data uji (20%) sehingga menghasilkan hasil dataset yang optimal. Tahap inti adalah Pelatihan Model IndoBERT, yang terdiri dari penggunaan model IndoBERT yang telah dilatih sebelumnya (*pre-trained*) dan proses *fine-tuning* hiperparameter menggunakan data latih. Terakhir, model yang telah di-*fine-tuning* akan melalui tahap Evaluasi Model menggunakan data uji untuk mengukur performa akhirnya. Setelah melewati semua proses tersebut maka dikeluarkan hasil akhir berupa deephoaks atau fakta.



Gambar 3.1. Desain penelitian diteksi deephoaks

3.2 Pengumpulan Data

Kualitas dan kuantitas data merupakan faktor krusial dalam membangun model *deep learning* yang andal. Seperti yang diilustrasikan pada Diagram 3.2, proses pengumpulan data melibatkan dua alur utama untuk memperoleh dataset yang seimbang antara kelas fakta dan hoaks.



Gambar 3.2. Pengumpulan Data

1. Data Berita Fakta (*Real News*): Kumpulan data berita fakta (sebanyak 32.000 artikel berita) diperoleh dengan mengunduh langsung dari platform Kaggle, melalui dataset "Indonesian News Dataset" yang diunggah oleh Iqbal Maulana. Dataset ini berisi artikel berita dari berbagai sumber media terkemuka di Indonesia dan digunakan sebagai representasi kelas "Fakta". Contoh dari dataset berita fakta dapat dilihat pada Gambar 3.3 [32].

id	source	title	content	date	created_at	updated_at
83	tempo	Depo Plumpang Terbakar Anggota DPR Minta Pertamina Pastikan Pasokan BBM Tak Terganggu	TEMPO.CO, Jakarta - Anggota Komisi VII DPR RI Rofik Hananto menyayangkan terjadinya insiden kebakaran...	2023-03-04 06:18:13+00	2023-03-04 07:03:39.039332	2023-03-04 07:03:39.039332
84	tempo	Jokowi Perintahkan Wapres Ma'ruf Amin Tinjau Lokasi Kebakaran Depo Plumpang	TEMPO.CO, Jakarta - Presiden Joko Widodo atau Jokowi memerintahkan Wakil Presiden Ma'ruf Amin untuk ...	2023-03-04 06:04:38+00	2023-03-04 07:03:39.039332	2023-03-04 07:03:39.039332
85	tempo	HW Mendukung Jasaah Umroh First Travel Dapatkan Haknya	INFO NASIONAL - Wakil Ketua WPR RI Dr. H. M. Hidayat Nur Wahid MA., atau HW menerima kunjungan perw...	2023-03-04 06:15:04+00	2023-03-04 07:03:39.039332	2023-03-04 07:03:39.039332

Gambar 3.3. Contoh dari Dataset Fakta - diambil dari Kaggle

2. Data Berita Hoaks (*Fake News*): Kumpulan data berita hoaks dikumpulkan melalui proses *web scraping* yang menargetkan situs Masyarakat Anti Fitnah Indonesia (MAFINDO), yaitu TurnBackHoax.id. Proses ini dirancang untuk mengambil artikel-artikel klarifikasi hoaks secara otomatis. Dalam pengumpulan, didapatkan data sebanyak 12.540 artikel berita hoaks melalui metode *web scraping*. Contoh dari dataset berita hoaks dapat dilihat pada Gambar 3.4 [33].



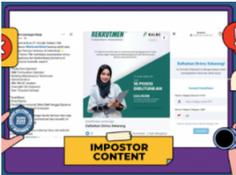
[SALAH] TikTok Bakal Tutup Juni 2025
 June 24, 2025 | Pemeriksa Fakta Junior | 0

Tidak ada pemberitahuan atau pengumuman resmi di laman resmi TikTok mengenai tutupnya platform tersebut



[SALAH] Simbol Tiga Garis di Grup WhatsApp Itu Tanda Kehadiran Hacker
 June 24, 2025 | Bentang Febrilian | 0

Faktanya, ikon tersebut adalah tombol Chat Audio yang terenkripsi end-to-end, bukan sebagai penanda terdapat hacker yang dapat menguras mobile banking.



[PENIPUAN] Tautan Pendaftaran Lowongan Kerja Kalbe Farma
 June 24, 2025 | Pemeriksa Fakta Junior | 0

Perusahaan menegaskan informasi rekrutmen dapat dilihat di laman kalbe.co.id dan media sosial resmi Kalbe Farma.

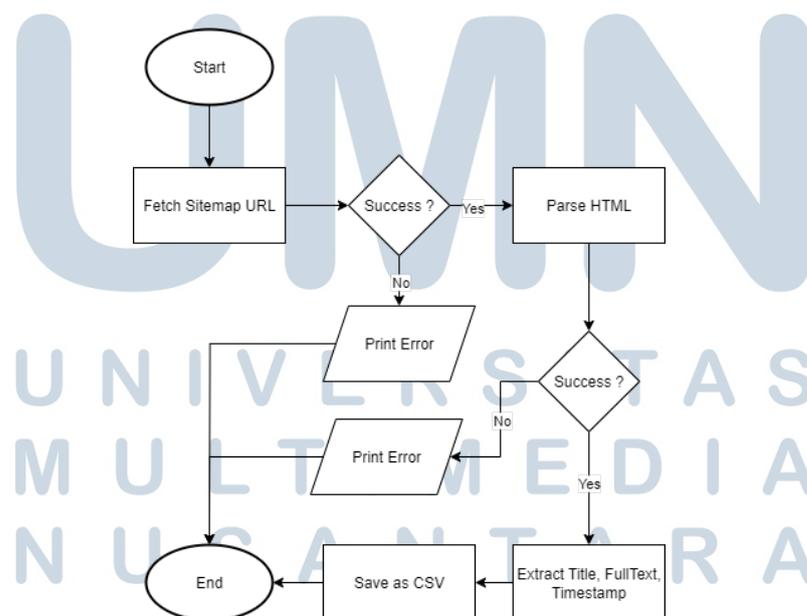
Gambar 3.4. Contoh dari Dataset Hoaks - diambil dari TurnBackHoax.id

3.2.1 Web Scraping

Pengambilan data dari TurnBackHoax.id dilakukan dengan menggunakan skrip kustom scraper.py dan memperhatikan etika scraping. Langkah pertama adalah memeriksa berkas robots.txt pada situs tersebut untuk memastikan bahwa proses scraping tidak melanggar kebijakan yang ditetapkan oleh pemilik situs.

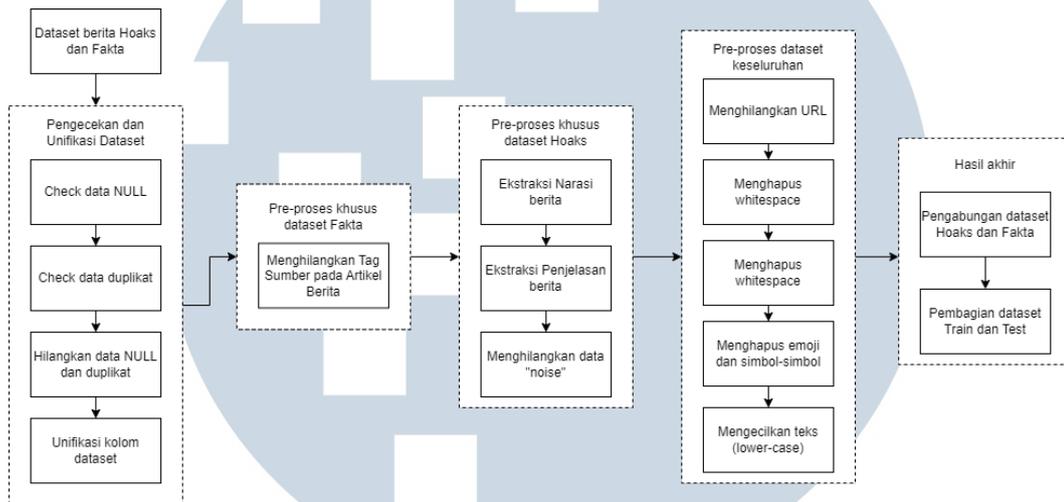
Proses scraping dilakukan secara asinkron untuk meningkatkan efisiensi waktu pengambilan data. Cara kerja dari proses scraping dapat dilihat pada Diagram 3.5. Skrip Python yang digunakan memanfaatkan beberapa pustaka utama, di antaranya:

1. [asyncio] dan [aiohttp]: Untuk menangani permintaan HTTP secara asinkron, memungkinkan pengiriman banyak permintaan secara bersamaan tanpa harus menunggu setiap permintaan selesai.
2. [BeautifulSoup]: Untuk melakukan parsing konten HTML dari halaman web yang diunduh, sehingga memudahkan ekstraksi informasi spesifik seperti judul, narasi, dan penjelasan artikel.
3. [csv] dan [aiofiles]: Untuk menyimpan data yang telah diekstraksi ke dalam format berkas CSV secara asinkron.



Gambar 3.5. Proses Web-Scraping

Setelah data berhasil dikumpulkan, tahap selanjutnya adalah *pre-processing*. Seperti yang ditunjukkan pada Gambar 3.6, tahap ini bertujuan untuk membersihkan, mentransformasi, dan menyiapkan data agar sesuai dengan format yang dibutuhkan oleh model IndoBERT dan untuk menghilangkan *noise* yang dapat mengganggu proses pelatihan.



Gambar 3.6. *pre-processing* Data

Langkah-langkah *pre-processing* yang dilakukan adalah sebagai berikut:

1. Pengecekan dan Unifikasi Dataset: Melihat dan menghilangkan data berita yang memiliki nilai NULL atau duplikat untuk memastikan dataset dapat digunakan dalam pelatihan model. Pada bagian ini juga dilakukan penyamaan tabel kolom antara kedua dataset.
2. Penghapusan Tag Sumber pada Berita Fakta: Dilakukan pembersihan dan penghapusan sumber berita ("TEMPO.CO, Jakarta -") dari isi artikel berita asli yang dapat menjadi bias pada pelatihan model.
3. Ekstraksi Konten Spesifik (Data Hoaks): Untuk data dari TurnBackHoax.id, hanya bagian teks dari "Narasi" dan "Penjelasan" yang diekstraksi. Menghapus data-data yang berpotensi menjadi *noise* seperti bagian "REFERENSI".
4. Pembersihan Teks Umum: Melakukan pembersihan teks secara menyeluruh pada seluruh dataset, yang mencakup:
 - (a) Menghilangkan URL dan tautan.

- (b) Menghapus whitespace / spasi berlebihan.
 - (c) Menghapus emoji dan simbol-simbol non-tekstual.
 - (d) Mengubah seluruh teks menjadi huruf kecil (*lowercasing*).
5. Penggabungan dan Penyeimbangan Dataset: Menggabungkan kedua dataset menjadi satu dataset akhir dengan rasio kelas yang seimbang (50:50). Dataset akhir terdiri dari sekitar 12.540 data hoaks dan 12.600 data fakta, dengan total sekitar 25.140 data.
 6. Pembagian Data: Dataset yang telah bersih dibagi menjadi tiga bagian untuk keperluan pelatihan dan evaluasi model, dengan proporsi 80% untuk data pelatihan dan 20% untuk data pengujian. Dari 80% data pelatihan tersebut, 10% di antaranya dialokasikan sebagai data validasi untuk memonitor performa model pada setiap *epoch* pelatihan.

3.3 Implementasi Model

Penelitian ini menggunakan model IndoBERT dengan arsitektur BERT yang telah dilatih sebelumnya pada korpus Bahasa Indonesia yang besar. Implementasi dilakukan dengan bantuan *framework* dan *library* dari Hugging Face.

1. Model: Model yang digunakan adalah [indolem/indobert-base-uncased], yang tersedia di repositori Hugging Face. Model ini dipilih karena telah memiliki pemahaman mendalam tentang struktur dan semantik Bahasa Indonesia, sehingga sangat cocok untuk di-*fine-tuning* pada tugas spesifik seperti deteksi berita hoaks.
2. Lingkungan Pengembangan dan Tools: Proses pengembangan dibagi menjadi dua lingkungan utama. Tahap *pre-processing* data dilakukan secara lokal menggunakan Visual Studio Code (VSC). Sementara itu, tahap pelatihan dan *fine-tuning* model yang membutuhkan sumber daya komputasi tinggi dilakukan di Google Colaboratory (Colab) untuk memanfaatkan akses ke unit pemrosesan grafis (T4-GPU). Implementasi model memanfaatkan *library* [transformers] dari Hugging Face yang menyediakan berbagai kelas penting, seperti:
 - (a) [AutoTokenizer]: Untuk memuat tokenizer yang sesuai dengan model IndoBERT.

- (b) `[AutoModelForSequenceClassification]`: Untuk memuat arsitektur IndoBERT dengan kepala klasifikasi di atasnya.
 - (c) `[TrainingArguments]` dan `[Training]`: Untuk mengelola seluruh proses pelatihan dan evaluasi dengan konfigurasi yang terstruktur.
3. Fine-Tuning dan Hiperparameter: Proses *fine-tuning* dilakukan dengan menyesuaikan bobot model IndoBERT pada dataset yang telah disiapkan. Konfigurasi hiperparameter yang digunakan dalam proses pelatihan diatur secara spesifik untuk mencapai performa optimal. Hiperparameter utama yang diatur adalah:
- (a) Ukuran Batch Pelatihan (Train Batch Size): 32
 - (b) Jumlah Epoch: 4
 - (c) Tingkat Pembelajaran (Learning Rate): 3×10^{-5}
 - (d) Penjadwal Tingkat Pembelajaran (Learning Rate Scheduler): Linear
 - (e) Weight Decay: 0.1
 - (f) Optimizer: AdamW (Adam with Weight Decay, defaultnya HuggingFace)

Pengaturan ini diimplementasikan menggunakan kelas `[TrainingArguments]`. Model dievaluasi pada data validasi di setiap akhir epoch, dan model dengan skor F1-Score tertinggi akan disimpan sebagai model terbaik.

3.4 Evaluasi Model

Evaluasi merupakan tahap krusial untuk mengukur performa dan keandalan model yang telah dilatih dalam mendeteksi berita hoaks. Evaluasi dilakukan pada data uji (20% dari total dataset) yang belum pernah dilihat oleh model selama proses pelatihan. Kinerja model diukur berdasarkan beberapa metrik standar yang dihitung dari *confusion matrix*.

1. Confusion Matrix: Sebuah tabel yang memvisualisasikan performa model dengan menunjukkan jumlah prediksi yang benar dan salah untuk setiap kelas. Komponen utamanya adalah:

- (a) True Positive (TP): Jumlah berita hoaks yang diprediksi dengan benar sebagai hoaks.
- (b) True Negative (TN): Jumlah berita fakta yang diprediksi dengan benar sebagai fakta.
- (c) False Positive (FP): Jumlah berita fakta yang salah diprediksi sebagai hoaks (Error Tipe I).
- (d) False Negative (FN): Jumlah berita hoaks yang salah diprediksi sebagai fakta (Error Tipe II).

2. Accuracy: Mengukur persentase prediksi yang benar dari keseluruhan data.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (3.1)$$

3. Precision: Mengukur tingkat ketepatan prediksi positif. Dari semua berita yang diprediksi sebagai hoaks, berapa persen yang benar-benar hoaks.

$$\text{Precision} = \frac{TP}{TP + FP} \quad (3.2)$$

4. Recall (Sensitivity): Mengukur kemampuan model untuk mengidentifikasi semua berita hoaks yang ada di dalam dataset.

$$\text{Recall} = \frac{TP}{TP + FN} \quad (3.3)$$

5. F1-Score: Rata-rata harmonik dari Precision dan Recall. Metrik ini memberikan skor tunggal yang menyeimbangkan kedua metrik tersebut dan sangat berguna ketika terjadi ketidakseimbangan kelas atau ketika kedua metrik sama-sama penting. Metrik ini menjadi acuan utama dalam memilih model terbaik.

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (3.4)$$

6. ROC-AUC (*Receiver Operating Characteristic - Area Under Curve*): Mengukur kemampuan model untuk membedakan antara kelas positif (hoaks) dan negatif (fakta). Kurva ROC memplot *True Positive Rate* (Recall) terhadap *False Positive Rate* di berbagai ambang batas klasifikasi. Nilai AUC yang mendekati 1.0 menunjukkan performa pemisahan kelas yang sangat baik.

Metrik-metrik ini dihitung menggunakan fungsi `[compute_metrics]` yang diintegrasikan ke dalam proses evaluasi, memastikan pengukuran yang konsisten dan komprehensif.

