

DETEKSI BERITA HOAKS BERBAHASA INDONESIA MENGGUNAKAN CNN-LSTM DAN INDOBERT DENGAN DUKUNGAN *EVIDENCE RETRIEVAL*

Junaedy Susanto, Taswanda Taryo, Abu Khalid Rivai

Teknik Informatika, Universitas Pamulang

Jl. Suryakencana No.1, Pamulang, Tangerang Selatan, Banten.

Junsusanto19@gmail.com

ABSTRAK

Penyebaran berita hoaks berbahasa Indonesia di media digital semakin meningkat dan berpotensi menimbulkan dampak sosial yang signifikan. Karakteristik berita hoaks yang menyerupai berita valid menyebabkan proses verifikasi informasi menjadi sulit dilakukan secara manual. Penelitian ini bertujuan mengembangkan sistem deteksi berita hoaks berbahasa Indonesia dengan membandingkan kinerja model IndoBERT dan CNN-LSTM serta mengintegrasikan *evidence retrieval* untuk mendukung verifikasi informasi. *Dataset* yang digunakan berasal dari Kaggle "Deteksi Berita Hoaks Indo" yang terdiri dari 23.942 berita dari TurnBackHoax.id, AntaraneWS, Kompas, dan Detik.com. Setelah proses pembersihan data dan deduplikasi diperoleh 21.130 data yang dibagi dengan rasio 70:15:15 untuk *data training*, *validation*, dan *testing*. Model CNN-LSTM dan IndoBERT dievaluasi menggunakan metrik *accuracy*, *precision*, *recall*, dan *F1-score*, kemudian diimplementasikan pada aplikasi web berbasis *Flask* yang mendukung input teks dan URL berita. Hasil penelitian menunjukkan bahwa IndoBERT memperoleh performa terbaik dengan *accuracy* 99,12% dan *F1-score* 0,99, sedangkan CNN-LSTM memperoleh *accuracy* 97% dan *F1-score* 0,97. Integrasi *evidence retrieval* berhasil menyediakan sumber informasi eksternal yang relevan sebagai pendukung hasil deteksi. Hasil penelitian menunjukkan bahwa IndoBERT lebih efektif dibandingkan CNN-LSTM dalam mendeteksi berita hoaks berbahasa Indonesia serta mampu mendukung proses verifikasi informasi yang lebih transparan.

Kata kunci : Deteksi berita hoaks, IndoBERT, CNN-LSTM, klasifikasi teks, *evidence retrieval*

1. PENDAHULUAN

Penyebaran informasi saat ini banyak terjadi melalui media daring yang memungkinkan setiap pengguna menyebarkan informasi tanpa melalui proses penyuntingan yang ketat. Kondisi ini menyebabkan berita hoaks mudah tersebar, terutama pada momen penting seperti pemilihan umum, bencana alam, dan krisis kesehatan. Informasi palsu sering disajikan secara meyakinkan sehingga mudah dipercaya dan dibagikan oleh masyarakat [1].

Perkembangan pemrosesan bahasa alami telah mendorong penggunaan model Transformer seperti IndoBERT, yang dirancang khusus untuk bahasa Indonesia dan terbukti memiliki performa tinggi dalam berbagai tugas klasifikasi teks, termasuk deteksi berita hoaks. [2].

Sebelum era *Transformer*, kombinasi CNN dan LSTM banyak digunakan karena mampu mengekstraksi fitur lokal serta memodelkan hubungan sekuensial antar kata secara efektif dan masih sering dijadikan model pembandingan pada penelitian terkini [3].

Selain klasifikasi berbasis teks, penelitian deteksi hoaks juga mulai mengadopsi pendekatan *evidence-based fact checking* yang memanfaatkan *evidence retrieval* dari sumber tepercaya untuk mendukung proses verifikasi dan meningkatkan transparansi hasil prediksi [4].

Meskipun IndoBERT terbukti efektif dalam deteksi berita hoaks, sebagian besar penelitian masih berfokus pada klasifikasi teks tanpa memanfaatkan *evidence retrieval* dan hanya menghasilkan label

prediksi tanpa bukti pendukung, sehingga transparansi serta kepercayaan pengguna terhadap sistem masih terbatas. [5].

Berdasarkan permasalahan tersebut, penelitian ini bertujuan membandingkan kinerja model IndoBERT dan CNN-LSTM dalam mendeteksi berita hoaks berbahasa Indonesia serta mengembangkan sistem deteksi berita hoaks berbasis web yang mampu menerima input berupa teks maupun URL berita. Sistem juga dilengkapi mekanisme *evidence retrieval* untuk memperoleh informasi pendukung dari sumber eksternal yang relevan.

Rencana penyelesaian masalah dilakukan melalui beberapa tahapan, yaitu pengumpulan dan pembersihan dataset berita berbahasa Indonesia, pelatihan model CNN-LSTM dan IndoBERT, evaluasi kinerja model menggunakan metrik *accuracy*, *precision*, *recall*, dan *F1-score*, serta integrasi model terbaik ke dalam aplikasi web berbasis *Flask*. Selanjutnya, sistem diintegrasikan dengan mekanisme *evidence retrieval* berbasis Google Custom Search API untuk menampilkan sumber informasi pendukung yang dapat digunakan sebagai dasar verifikasi hasil deteksi berita.

2. TINJAUAN PUSTAKA

2.1. Penelitian Terdahulu

Berbagai penelitian sebelumnya telah menunjukkan efektivitas model berbasis *Transformer*, khususnya IndoBERT, dalam deteksi berita hoaks berbahasa Indonesia. Salah satu studi yang membandingkan IndoBERT Base, IndoBERT Large,

CNN-LSTM, dan beberapa algoritma *machine learning* klasik menunjukkan bahwa IndoBERT *Large* mampu mencapai akurasi hingga 98% setelah proses *oversampling*, jauh melampaui CNN-LSTM yang mengalami *overfitting* dan hanya memperoleh akurasi 44% [2].

Penelitian lain yang membandingkan IndoBERT dan mBERT pada deteksi hoaks politik berbahasa Indonesia juga menunjukkan keunggulan IndoBERT dengan akurasi 95,6% dan ROC-AUC 98,5%, yang mengindikasikan kemampuan lebih baik dalam memahami konteks linguistik bahasa Indonesia [5].

Temuan serupa diperoleh pada studi yang membandingkan IndoBERT, metode *ensemble*

machine learning, dan CNN-LSTM menggunakan lebih dari enam ribu artikel berita, di mana IndoBERT mencapai akurasi 92,24% serta menunjukkan stabilitas pelatihan yang lebih baik dibandingkan CNN-LSTM [6].

Secara umum, penelitian-penelitian tersebut menunjukkan bahwa IndoBERT secara konsisten mengungguli model *machine learning* tradisional maupun CNN-LSTM dalam tugas deteksi berita hoaks berbahasa Indonesia. Namun, sebagian besar penelitian masih berfokus pada klasifikasi teks secara langsung tanpa mengintegrasikan mekanisme *evidence retrieval* untuk memverifikasi klaim berita menggunakan sumber eksternal yang tepercaya.

Tabel 1. Perbandingan Penelitian Terdahulu

Peneliti & Tahun	Model	Akurasi	Dataset
Tobing et al. (2025)	IndoBERT vs MBERT	95,60%	Kaggle Fake News (20.800 data)
Ridho et al. (2024)	IndoBERT Large + SMOTE	98%	600 berita Indonesia (diseimbangkan)
Pekandi et al. (2025)	IndoBERT vs CNN-LSTM	92,24%	6.120 artikel (seimbang)
Junaedy Susanto (2026) — Penelitian ini	IndoBERT + CNN-LSTM + Evidence Retrieval	99,12%	21.130 rita Indonesia (Kaggle)

2.2. Deteksi Berita Hoaks

Proses deteksi berita hoaks merupakan proses mengidentifikasi informasi yang tidak benar atau menyesatkan untuk membedakannya dari berita valid. Hoaks termasuk bentuk disinformasi yang dapat menyebar dengan cepat melalui media sosial dan portal berita daring, sehingga berpotensi menurunkan kepercayaan publik. Oleh karena itu, deteksi hoaks otomatis menjadi salah satu bidang penting dalam *Natural Language Processing* (NLP) dan *machine learning* [7].

Sistem deteksi hoaks umumnya menerima input berupa teks berita yang kemudian diproses untuk menghasilkan prediksi label, seperti hoaks atau valid. *Dataset* yang digunakan biasanya berupa kumpulan berita yang telah diberi label kebenaran secara manual [8]. Secara umum, pendekatan deteksi hoaks terdiri atas:

- Machine Learning* Tradisional, seperti *Naive Bayes*, *SVM*, *Random Forest*, dan *Logistic Regression* yang menggunakan fitur tekstual seperti *TF-IDF* atau *Bag-of-Words*.
- Deep Learning*, seperti CNN dan LSTM yang mampu mempelajari representasi fitur secara otomatis serta menangkap pola lokal dan hubungan sekuensial dalam teks.
- Pretrained Language Models (Transformer)*, seperti BERT dan turunannya, yang mampu memahami konteks teks secara lebih baik melalui representasi dua arah sehingga menghasilkan akurasi yang lebih tinggi pada tugas deteksi hoaks.

2.3. Natural Language Processing (NLP)

Natural Language Processing (NLP) merupakan bidang kecerdasan buatan yang berfokus pada pemrosesan bahasa manusia oleh mesin. Dalam klasifikasi teks, NLP mengubah teks mentah menjadi representasi numerik melalui tahapan seperti *text*

normalization, *tokenization*, dan *transformasi* vektor. Kualitas representasi teks sangat memengaruhi performa sistem deteksi hoaks [9]. Pada sistem deteksi berita hoaks, NLP berfungsi mengubah dokumen teks yang dirumuskan sebagai berikut:

$$d = (w_1, w_2, \dots, w_n), \quad (1)$$

menjadi vektor fitur x melalui fungsi representasi $\phi(d)$, yang kemudian digunakan sebagai masukan model klasifikasi. Pada pendekatan tradisional, representasi teks umumnya menggunakan *Bag-of-Words* atau *TF-IDF* yang menghitung bobot kemunculan kata dalam dokumen. Meskipun efisien, metode ini bersifat *context-free* sehingga kurang mampu memahami makna kata berdasarkan konteks kalimat.

Perkembangan NLP modern menghadirkan *word embedding* dan *contextual embedding*. Model berbasis *Transformer* seperti BERT mampu menghasilkan representasi kontekstual yang lebih baik dibandingkan metode sebelumnya. Dalam bahasa Indonesia, model yang dilatih khusus pada korpus lokal, seperti IndoBERT, terbukti lebih efektif dibandingkan model multibahasa karena mampu menangkap karakteristik linguistik bahasa Indonesia dengan lebih baik [2]. Oleh karena itu, NLP menjadi komponen penting dalam meningkatkan akurasi sistem deteksi berita hoaks berbasis IndoBERT maupun CNN-LSTM.

2.4. IndoBERT

Model IndoBERT merupakan adaptasi dari BERT (*Bidirectional Encoder Representations from Transformers*) yang dilatih menggunakan korpus berbahasa Indonesia. Berbeda dengan model sekuensial seperti RNN dan LSTM, IndoBERT memanfaatkan arsitektur *Transformer* dengan mekanisme *self-attention* sehingga mampu memahami konteks kata dari arah kiri dan kanan secara bersamaan [10].

Pada *Transformer*, setiap token direpresentasikan sebagai vektor *embedding* dan diproses menggunakan mekanisme *scaled dot product attention* yang dirumuskan sebagai:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \quad (2)$$

Keterangan Rumus:

Q = Query

K = Key

V = Value

d_k = dimensi vektor key

Mekanisme ini memungkinkan setiap token memperhatikan token lain dalam kalimat sehingga konteks global dapat dipahami dengan lebih baik. BERT dilatih menggunakan dua tugas utama, yaitu *Masked Language Modeling* (MLM) dan *Next Sentence Prediction* (NSP), dengan fungsi kerugian:

$$L = L_{MLM} + L_{NSP} \quad (3)$$

Keterangan Rumus:

L_{MLM} = loss untuk prediksi token yang disembunyikan

L_{NSP} = loss untuk prediksi hubungan antar kalimat

IndoBERT mengadopsi mekanisme tersebut dan dilatih menggunakan berbagai sumber teks berbahasa Indonesia sehingga lebih mampu menangkap karakteristik linguistik lokal, seperti imbuhan, variasi morfologi, dan kosakata khas Indonesia. Berbagai penelitian menunjukkan bahwa IndoBERT memberikan performa lebih baik dibandingkan model multibahasa seperti mBERT pada berbagai tugas NLP berbahasa Indonesia [6] [2].

2.5. CNN-LSTM

Model CNN-LSTM merupakan kombinasi dua arsitektur *deep learning* yang memanfaatkan keunggulan CNN dalam mengekstraksi pola lokal (kata atau frasa) dan LSTM dalam memodelkan hubungan sekuensial antar kata. Kombinasi ini banyak digunakan dalam klasifikasi teks sebelum berkembangnya model berbasis *Transformer* [11]. Dokumen teks terlebih dahulu direpresentasikan sebagai urutan vektor *embedding*:

$$X = (x_1, x_2, \dots, x_n), \quad x_i \in \mathbb{R}^d, \quad (4)$$

Keterangan:

X = representasi dokumen

x_i = vektor *embedding* token ke- i

d = dimensi *embedding*

CNN kemudian melakukan ekstraksi fitur lokal melalui operasi konvolusi:

$$c_i = \sigma(W \cdot X_{i:i+h-1} + b), \quad (5)$$

Keterangan:

c_i = fitur hasil konvolusi

W = bobot filter

b = bias

σ = fungsi aktivasi

h = ukuran filter

Hasil konvolusi diringkas menggunakan *pooling* untuk menghasilkan fitur yang lebih kompak sebelum diteruskan ke LSTM. Pada LSTM, informasi diproses melalui mekanisme *input gate*, *forget gate*, dan *output gate* untuk mempertahankan informasi penting serta

mengurangi informasi yang tidak relevan. Secara umum, pembaruan memori dan *hidden state* dirumuskan sebagai:

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t, \quad (6)$$

$$h_t = o_t \odot \tanh(c_t) \quad (7)$$

Keterangan:

i_t = *input gate*

f_t = *forget gate*

o_t = *output gate*

c_t = *cell state*

h_t = *hidden state*

$\odot$ = perkalian elemen per elemen

Secara keseluruhan, alur CNN-LSTM dapat dinyatakan sebagai:

$$F = \text{CNN}(X), \quad (8)$$

$$H = \text{LSTM}(F), \quad (9)$$

$$\hat{y} = g(H), \quad (10)$$

Keterangan:

F = fitur hasil ekstraksi CNN

H = representasi sekuensial hasil LSTM

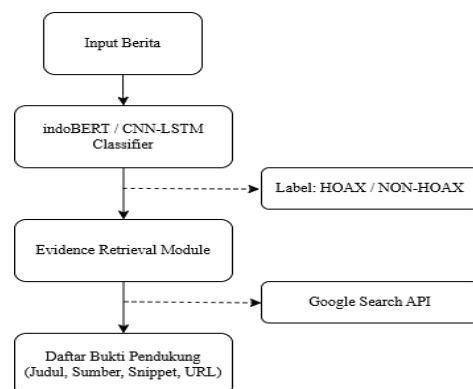
$\hat{y}$ = hasil prediksi kelas

$g(\cdot)$ = fungsi klasifikasi (*dense layer* dan *softmax*)

Berbagai penelitian menunjukkan bahwa CNN-LSTM mampu memberikan performa lebih baik dibandingkan CNN atau LSTM tunggal karena dapat menangkap pola lokal sekaligus hubungan sekuensial dalam teks [10] [11]. Namun, model ini masih memiliki keterbatasan dalam memahami konteks global dan relasi semantik yang kompleks pada teks panjang. Oleh karena itu, penelitian ini menggunakan CNN-LSTM sebagai model pembanding terhadap IndoBERT untuk menganalisis perbedaan kinerja antara pendekatan *deep learning* konvensional dan *Transformer* dalam deteksi berita hoaks berbahasa Indonesia.

2.6. Evidence Retrieval

Evidence retrieval merupakan pendekatan *fact-checking* yang memverifikasi klaim dalam berita dengan membandingkannya terhadap informasi dari sumber tepercaya. Berbeda dengan pendekatan *content-based* yang hanya mengandalkan isi berita, *evidence retrieval* memanfaatkan bukti eksternal sehingga sistem tidak hanya menghasilkan label hoaks atau non-hoaks, tetapi juga menyediakan dasar faktual yang mendukung hasil prediksi [12].



Gambar 1. Arsitektur Integrasi *Evidence retrieval*

Proses *evidence retrieval* dimulai dengan mengekstraksi klaim utama dari berita, kemudian menggunakannya sebagai *query* untuk mencari dokumen relevan melalui *Google Search API*. Hasil pencarian berupa artikel dari sumber tepercaya yang diseleksi berdasarkan relevansinya terhadap klaim yang dianalisis.

Dalam penelitian ini, *evidence retrieval* digunakan setelah proses klasifikasi oleh IndoBERT atau CNN-LSTM. Sistem menampilkan artikel, sumber, cuplikan isi, dan tautan yang mendukung atau menyangkal klaim berita. Dengan demikian, sistem tidak hanya memberikan prediksi, tetapi juga menyajikan bukti yang meningkatkan transparansi, keandalan, dan kepercayaan pengguna terhadap hasil deteksi hoaks.

3. METODE PENELITIAN

3.1. Dataset Penelitian

Tabel 2. Rincian *Dataset* Tiap Sumber

No	Sumber	Jumlah Data
1	TurnBackHoax.id	12.744
2	AntaraneWS	4.200
3	Kompas	3.500
4	Detik.com	3.298

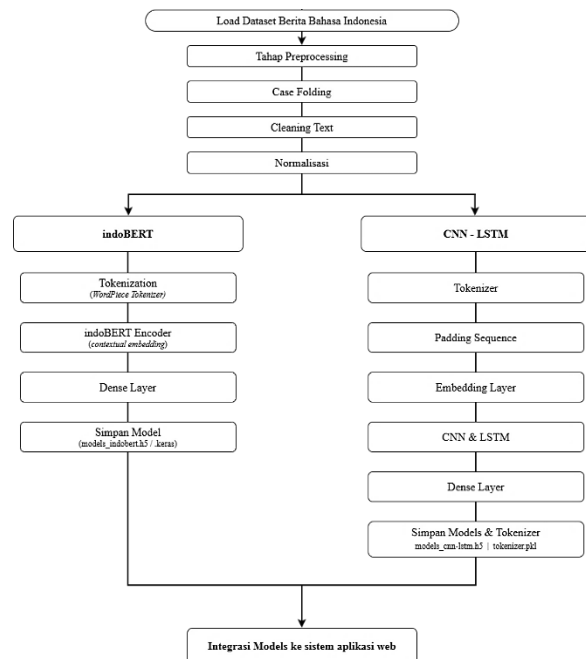
Dataset penelitian diperoleh dari platform Kaggle dengan judul "Deteksi Berita Hoaks Indo" yang dikembangkan oleh Mochamad Abdul Azis. *Dataset* ini mencakup 23.942 data berita dari empat sumber: TurnBackHoax.id (12.744 data hoaks yang telah diverifikasi), AntaraneWS (4.200 data), Kompas (3.500 data), dan Detik.com (3.298 data). Setelah tahap pembersihan data yang meliputi penghapusan *missing values* dan *deduplication*, jumlah data berkurang menjadi 21.130. Distribusi kelas relatif seimbang, dengan label 1 (hoaks) berjumlah sekitar 12.700 data dan label 0 (non-hoaks) sekitar 11.200 data. *Dataset* kemudian dibagi dengan rasio 70:15:15 menjadi 14.791 *data training*, 3.169 *data validation*, dan 3.170 *data testing*. Atribut narasi berita dipilih sebagai input utama karena mengandung informasi kontekstual yang lebih lengkap dibandingkan judul.

3.2. Tahap Penelitian

Penelitian ini diawali dengan pengumpulan *dataset* berita berbahasa Indonesia yang diperoleh dari platform Kaggle dengan judul Deteksi Berita Hoaks Indo. *Dataset* terdiri dari 23.942 berita yang bersumber dari TurnBackHoax.id, AntaraneWS, Kompas, dan Detik.com. Selanjutnya dilakukan proses pembersihan data (*data cleaning*) yang meliputi penghapusan *missing values* dan data duplikat (*deduplication*), sehingga diperoleh 21.130 data yang siap digunakan dalam penelitian.

Tahap berikutnya adalah teks yang mencakup *case folding*, *cleaning*, *tokenizer*, dan penyesuaian panjang teks. Pada penelitian ini, atribut narasi berita digunakan sebagai input utama karena memiliki informasi kontekstual yang lebih lengkap

dibandingkan judul berita. *Dataset* kemudian dibagi menggunakan rasio 70:15:15 menjadi *data training*, *validation*, dan *testing* untuk mendukung proses pelatihan dan evaluasi model secara optimal.



Gambar 2. Tahap Penelitian

Data hasil kemudian diproses menggunakan dua pendekatan klasifikasi yang berbeda, yaitu CNN-LSTM dan IndoBERT. Pada model CNN-LSTM dilakukan tokenisasi, pembentukan *padding sequence*, proses *embedding*, serta pembelajaran menggunakan kombinasi lapisan CNN dan LSTM. Sementara itu, pada model IndoBERT dilakukan tokenisasi menggunakan *WordPiece Tokenizer*, ekstraksi representasi kontekstual melalui *IndoBERT Encoder*, dan klasifikasi menggunakan *dense layer*. Setelah proses pelatihan selesai, masing-masing model disimpan untuk digunakan pada tahap implementasi sistem.

Tahap berikutnya adalah integrasi model CNN-LSTM dan IndoBERT ke dalam sistem deteksi berita hoaks berbasis *web* menggunakan *framework Flask*. Sistem yang dikembangkan mendukung input berupa teks maupun url berita serta dilengkapi dengan fitur *evidence retrieval* untuk memperoleh informasi pendukung dari sumber eksternal. Pada tahap akhir dilakukan pengujian sistem untuk memastikan seluruh fungsi deteksi berita dapat berjalan dengan baik sesuai kebutuhan penelitian.

3.3. Tahap Preprocessing

Tahap dilakukan untuk meningkatkan kualitas data teks sebelum digunakan dalam proses pelatihan model CNN-LSTM dan IndoBERT. Proses ini bertujuan mengurangi noise serta menyeragamkan format teks sehingga informasi yang relevan dapat dipelajari secara lebih efektif oleh model. Pada penelitian ini, dilakukan pada atribut narasi berita

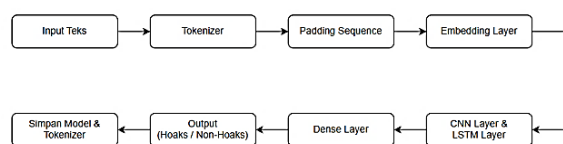
karena memiliki informasi kontekstual yang lebih lengkap dibandingkan atribut lainnya.

Tahapan meliputi *case folding* untuk mengubah seluruh huruf menjadi huruf kecil, *cleaning* untuk menghapus karakter khusus, tanda baca, angka, url, dan simbol yang tidak diperlukan, serta tokenisasi untuk memecah teks menjadi unit kata atau *token*. Selanjutnya dilakukan penyesuaian panjang teks (*padding dan truncation*) agar seluruh data memiliki ukuran input yang seragam sesuai kebutuhan model. Hasil kemudian digunakan sebagai masukan pada proses pelatihan dan evaluasi model CNN-LSTM maupun IndoBERT.

3.4. CNN-LSTM

Kolaborasi model CNN-LSTM merupakan model *deep learning* yang menggabungkan CNN untuk mengekstraksi pola penting pada teks dan LSTM untuk mempelajari hubungan sekuensial antar kata. Kombinasi kedua arsitektur ini memungkinkan model menangkap informasi lokal sekaligus mempertahankan konteks dalam narasi berita. Teks hasil terlebih dahulu dikonversi menjadi urutan token dan diseragamkan menggunakan padding hingga 200 token. Selanjutnya, lapisan CNN melakukan ekstraksi fitur, kemudian hasilnya diproses oleh LSTM untuk mempelajari keterkaitan antar kata. Pada lapisan keluaran digunakan fungsi aktivasi *sigmoid* untuk menghasilkan probabilitas klasifikasi ke dalam kategori hoaks atau non-hoaks.

Proses pelatihan dilakukan menggunakan *data training* dan *validation* untuk memantau kinerja model serta mengurangi risiko *overfitting*. Setelah pelatihan selesai, model dievaluasi menggunakan *data testing* berdasarkan metrik *accuracy*, *precision*, *recall*, dan *F1-score*. Dalam penelitian ini, CNN-LSTM digunakan sebagai model pembandingan terhadap IndoBERT pada tugas deteksi berita hoaks berbahasa Indonesia.



Gambar 3. Arsitektur Model CNN-LSTM

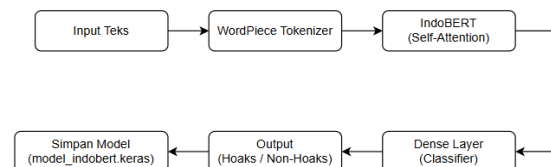
Arsitektur diatas merupakan kombinasi model CNN-LSTM yang digunakan pada penelitian ini. Teks hasil terlebih dahulu diubah menjadi token numerik menggunakan *tokenizer* dan diseragamkan panjangnya melalui *padding sequence*. Selanjutnya, *embedding layer* mengonversi token menjadi representasi vektor yang diproses oleh lapisan CNN untuk mengekstraksi fitur penting dan lapisan LSTM untuk mempelajari hubungan sekuensial antar kata. Hasil pembelajaran kemudian diteruskan ke *dense layer* untuk menghasilkan prediksi hoaks atau non-hoaks beserta nilai probabilitasnya. Setelah pelatihan selesai, model dan *tokenizer* disimpan sehingga dapat digunakan

kembali pada tahap implementasi sistem tanpa perlu dilakukan pelatihan ulang.

3.5. IndoBERT

Model IndoBERT digunakan sebagai pendekatan berbasis *Transformer* untuk melakukan klasifikasi berita hoaks berbahasa Indonesia. teks hasil prapemrosesan diproses menggunakan *WordPiece Tokenizer* untuk mengubah teks menjadi sub-kata (*subword*) yang sesuai dengan kosakata model. Proses ini menghasilkan input IDs, *attention mask*, dan padding sebagai masukan bagi arsitektur *Transformer*.

Token yang dihasilkan kemudian diproses oleh IndoBERT *Encoder* untuk membentuk representasi vektor kontekstual dua arah, sehingga makna setiap token dipengaruhi oleh konteks sebelum dan sesudahnya dalam kalimat. Representasi tersebut diteruskan ke *dense layer* yang berfungsi sebagai pengklasifikasi untuk menghasilkan probabilitas kelas hoaks dan non-hoaks melalui proses *fine-tuning*.



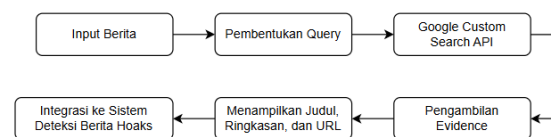
Gambar 4. Pelatihan Model IndoBERT

Model selanjutnya menghasilkan output prediksi berupa label klasifikasi beserta nilai probabilitas yang menunjukkan tingkat keyakinan prediksi. Setelah pelatihan selesai, model disimpan dalam format *model_indobert.keras* atau *model_indobert.h5* sehingga dapat digunakan kembali dan diintegrasikan ke dalam aplikasi *web* berbasis *Flask* tanpa perlu dilakukan pelatihan ulang.

3.6. Evidence Retrieval

Untuk mendukung proses verifikasi, sistem dilengkapi fitur *evidence retrieval* yang memanfaatkan *Google Custom Search API* untuk mencari informasi eksternal yang relevan berdasarkan judul atau isi berita. Hasil pencarian berupa artikel dari berbagai sumber daring kemudian dipilih sebagai *evidence* yang berkaitan dengan topik berita yang dianalisis.

Informasi yang ditampilkan meliputi judul artikel, ringkasan konten, dan tautan sumber asli. *Evidence* tersebut tidak memengaruhi hasil prediksi CNN-LSTM maupun IndoBERT, melainkan berfungsi sebagai informasi pendukung untuk membantu pengguna memverifikasi kebenaran berita.



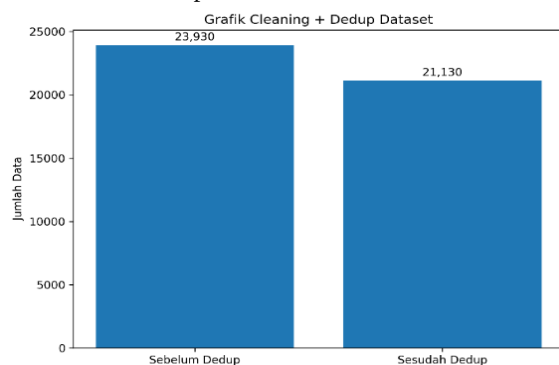
Gambar 5. Alur Evidence Retrieval

Fitur *evidence retrieval* diintegrasikan ke dalam aplikasi *Flask* dan dijalankan secara otomatis setiap kali proses deteksi dilakukan. Dengan demikian, sistem tidak hanya menghasilkan klasifikasi hoaks atau non-hoaks, tetapi juga menyediakan sumber informasi tambahan yang dapat dijadikan bahan pertimbangan oleh pengguna.

4. HASIL DAN PEMBAHASAN

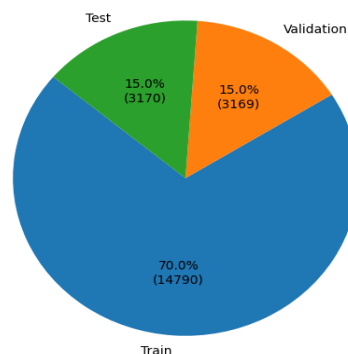
4.1. Hasil Data

Tahap dilakukan untuk meningkatkan kualitas *dataset* sebelum digunakan dalam pelatihan CNN-LSTM dan IndoBERT. *Dataset* awal berjumlah 23.942 berita yang berasal dari TurnBackHoax.id, Antaraneews, Kompas, dan Detik.



Gambar 6. Proses Cleaning Duplicat *Dataset*

Melalui proses *data cleaning* berupa deleting *missing values* dan data duplikat, sebanyak 2.812 data dihapus sehingga tersisa 21.130 data.



Gambar 7. Proporsi Pembagian *Dataset*

Selanjutnya dilakukan teks yang meliputi *case folding*, *cleaning*, *tokenizer*, dan penyesuaian panjang teks untuk menghasilkan data yang lebih terstruktur. *Dataset* kemudian dibagi secara *stratified* dengan rasio 70:15:15 menjadi *data training* (14.791 data), *validation* (3.169 data), dan *testing* (3.170 data). Pembagian data dilakukan secara *stratified* untuk menjaga proporsi masing-masing kelas pada setiap subset data.

Hasil menunjukkan bahwa *dataset* memiliki jumlah data yang memadai dan distribusi kelas yang relatif seimbang, sehingga dapat mendukung proses pelatihan dan evaluasi model secara optimal.

Tabel 3. Proporsi *Dataset*

Tahap <i>Dataset</i>	Jumlah Data	Persentase
<i>Dataset</i> Awal	23.942	100%
Data Terhapus	2.812	11,74%
<i>Dataset</i> Final	21.130	88,26%
Training Set	14.791	70%
Validation Set	3.169	15%
Testing Set	3.170	15%

4.2. Hasil Pelatihan CNN-LSTM

Model CNN-LSTM dilatih menggunakan 14.791 *data training* dan 3.169 *data validation* selama 10 *epoch*.

=== HISTORY TRAINING ===

	loss	accuracy	val_loss	val_accuracy	epoch
0	0.142983	0.951930	0.072758	0.979489	1
1	0.041950	0.988304	0.084307	0.971600	2
2	0.025763	0.993239	0.115680	0.966235	3
3	0.019135	0.994997	0.089259	0.976964	4
4	0.010705	0.997228	0.104883	0.978227	5
5	0.007560	0.998175	0.143509	0.970338	6
6	0.004906	0.998986	0.116391	0.974755	7
7	0.002613	0.999459	0.116536	0.979804	8
8	0.001690	0.999797	0.136551	0.977911	9
9	0.003134	0.999459	0.133063	0.976964	10

=== BEST EPOCH ===

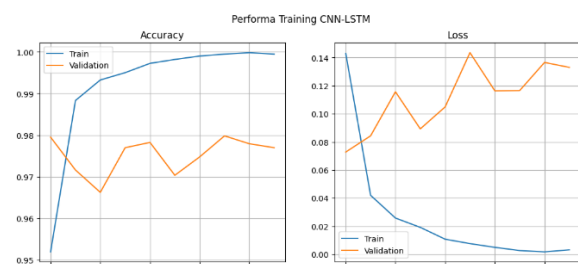
Epoch terbaik: 8

Validation Accuracy: 0.9798

Validation Loss: 0.1165

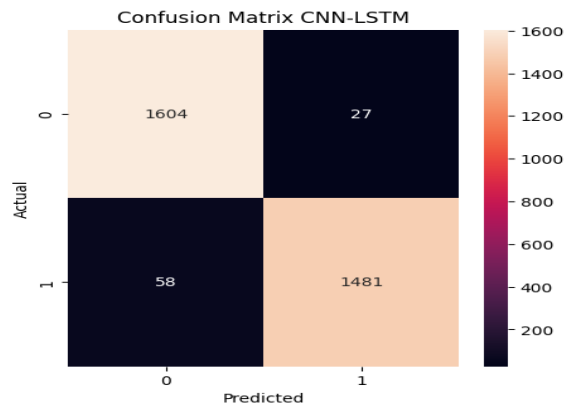
Gambar 8. Proporsi Pembagian *Dataset*

Hasil pelatihan menunjukkan peningkatan *training accuracy* dari 95,19% menjadi 99,95%, disertai penurunan *training loss* dari 0,1429 menjadi 0,0031. Sementara itu, *validation accuracy* berada pada rentang 96,62%–97,98%, dengan nilai terbaik sebesar 97,98% pada epoch ke-8.



Gambar 9. Proporsi Pembagian *Dataset*

Berdasarkan kurva pelatihan, *training accuracy* terus meningkat sedangkan *validation accuracy* relatif stabil di sekitar 97%. Selisih keduanya pada akhir pelatihan sekitar 2,25%, yang mengindikasikan *overfitting* ringan, namun model tetap menunjukkan performa yang konsisten pada data validasi. Pengujian menggunakan 3.170 *data testing* menghasilkan *accuracy*, *precision*, *recall*, dan *F1-score* sebesar 97%. Pada kelas non-hoaks, model memperoleh *precision* 97%, *recall* 98%, dan *F1-score* 97%, sedangkan pada kelas hoaks diperoleh *precision* 98%, *recall* 96%, dan *F1-score* 97%, menunjukkan kemampuan klasifikasi yang seimbang pada kedua kelas.



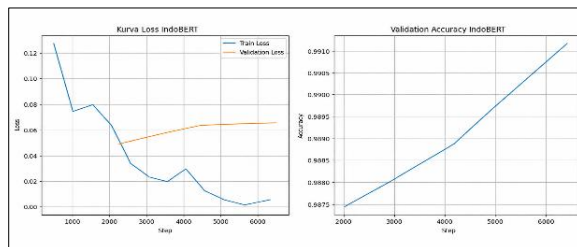
Gambar 10. Proporsi Pembagian Dataset

Hasil *confusion matrix* menunjukkan 3.085 dari 3.170 data berhasil diklasifikasikan dengan benar, dengan rincian 1.604 *true negative* dan 1.481 *true positive*. Kesalahan klasifikasi hanya terjadi pada 85 data, terdiri dari 27 *false positive* dan 58 *false negative*, sehingga tingkat kesalahan model sebesar 2,68%.

Secara keseluruhan, CNN-LSTM menunjukkan performa yang sangat baik dalam mendeteksi berita hoaks berbahasa Indonesia, dengan kemampuan yang efektif dalam mengekstraksi fitur lokal dan mempelajari hubungan sekuensial antar kata.

4.3. Hasil Pelatihan IndoBERT

Model IndoBERT dilatih menggunakan *dataset* berita berbahasa Indonesia yang telah melalui tahap dan tokenisasi



Gambar 11. Proporsi Pembagian Dataset

=== HASIL EVALUASI PER EPOCH ===

	step	eval_loss	eval_accuracy	eval_f1
4	2113	0.048717	0.987459	0.987127
9	4226	0.063767	0.988878	0.988478
14	6339	0.065282	0.991245	0.990969

=== BEST PERFORMANCE ===

Step: 6339

Accuracy: 0.9912

F1-score: 0.9910

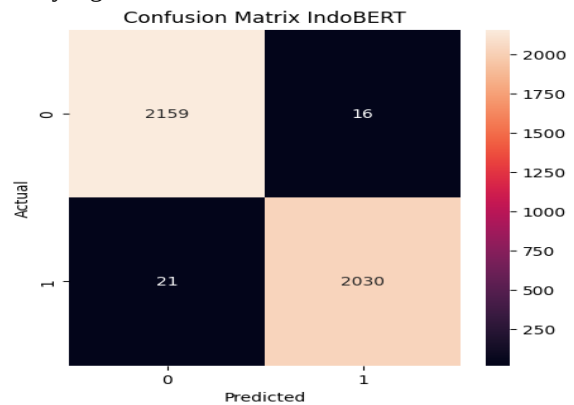
Loss: 0.0653

Gambar 11. Proporsi Pembagian Dataset

Hasil *fine-tuning* menunjukkan performa yang sangat baik dengan *validation accuracy* meningkat dari 98,75% menjadi 99,12%, sementara *training loss*

turun lebih dari 96% dari 0,127 menjadi 0,005. Nilai *validation loss* juga relatif stabil pada rentang 0,049–0,065, yang menunjukkan kemampuan generalisasi model yang baik.

Pengujian menggunakan 4.226 *data testing* menghasilkan *accuracy* sebesar 99,12%, dengan *precision*, *recall*, dan *F1-score* masing-masing mencapai 99% pada kelas hoaks maupun non-hoaks. Hasil ini menunjukkan bahwa model mampu mengidentifikasi kedua kelas secara seimbang tanpa bias yang berarti.



Gambar 12. Proporsi Pembagian Dataset

Berdasarkan *confusion matrix*, sebanyak 4.189 dari 4.226 data berhasil diklasifikasikan dengan benar, terdiri atas 2.159 *true negative* dan 2.030 *true positive*. Kesalahan klasifikasi hanya terjadi pada 37 data, yaitu 16 *false positive* dan 21 *false negative*, sehingga tingkat kesalahan model hanya 0,88%.

Secara keseluruhan, IndoBERT menunjukkan performa yang sangat tinggi dalam deteksi berita hoaks berbahasa Indonesia. Kemampuan *self-attention* dalam memahami konteks antar kata dan kalimat tercermin dari nilai evaluasi yang konsisten pada kisaran 99% serta rendahnya jumlah kesalahan klasifikasi pada data pengujian.

4.4. Perbandingan Kinerja Model

Perbandingan kinerja model dilakukan menggunakan metrik *accuracy*, *precision*, *recall*, dan *F1-score* berdasarkan hasil pengujian pada *data testing*.

Tabel 4. Perbandingan Hasil Model

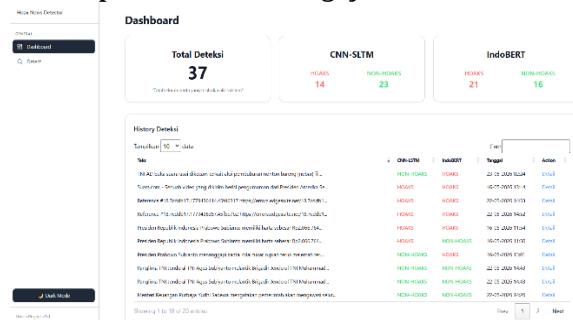
Metrik	CNN-LSTM	IndoBERT
Accuracy	97%	99,12%
Precision	97%	99%
Recall	97%	99%
F1-Score	97%	99%
Error Rate	2,68%	0,88%

Dari sisi tingkat kesalahan, CNN-LSTM menghasilkan *error rate* sebesar 2,68%, sedangkan IndoBERT hanya 0,88%. Hal ini menunjukkan bahwa IndoBERT mampu menurunkan kesalahan klasifikasi sebesar 1,80 poin persentase atau sekitar 67,16% dibandingkan CNN-LSTM. Hasil *confusion matrix* juga menunjukkan bahwa CNN-LSTM menghasilkan

85 kesalahan klasifikasi (27 *false positive* dan 58 *false negative*), sedangkan IndoBERT hanya menghasilkan 37 kesalahan (16 *false positive* dan 21 *false negative*). Dengan demikian, jumlah kesalahan IndoBERT lebih rendah sebanyak 48 data dibandingkan CNN-LSTM.

Perbedaan ini menunjukkan bahwa kemampuan *self-attention* pada IndoBERT lebih efektif dalam memahami konteks dan makna semantik teks dibandingkan CNN-LSTM. Meskipun demikian, CNN-LSTM tetap menunjukkan performa yang baik dengan *accuracy* di atas 97%, sehingga masih layak digunakan untuk tugas deteksi berita hoaks berbahasa Indonesia.

4.5. Implementasi dan Pengujian Sistem



Gambar 13. Proporsi Pembagian Dataset

Model CNN-LSTM dan IndoBERT yang telah dilatih kemudian diintegrasikan ke dalam sistem deteksi berita hoaks berbasis *web* menggunakan framework *Flask*.

Gambar 14. Proporsi Pembagian Dataset

Sistem dirancang untuk mendukung dua jenis masukan, yaitu teks narasi berita secara langsung dan url berita dari portal berita daring. Pada input url, sistem melakukan proses *web scraping* untuk mengambil isi berita secara otomatis sebelum dilakukan klasifikasi oleh kedua model.

Selain klasifikasi, sistem dilengkapi fitur *evidence retrieval* berbasis *Google Custom Search API* untuk memperoleh informasi pendukung dari sumber eksternal. Hasil prediksi, *confidence score*, *evidence*, dan riwayat deteksi kemudian ditampilkan serta disimpan ke dalam basis data.

Deteksi Berita

CNN-LSTM | IndoBERT | Evidence Retrieval

Gambar 15. Proporsi Pembagian Dataset

Pengujian diatas menunjukkan hasil deteksi berita menggunakan input url berita yang diproses melalui *web scraping* sebelum diklasifikasikan oleh model CNN-LSTM dan IndoBERT. url yang digunakan berasal dari situs resmi Komdigi yang memuat daftar berita yang telah terverifikasi sebagai hoaks. Hasil pengujian menunjukkan bahwa kedua model berhasil mengidentifikasi berita sebagai hoaks. CNN-LSTM menghasilkan *confidence score* sebesar 72,82%, sedangkan IndoBERT mencapai 100,00%. Hasil tersebut sesuai dengan label hoaks yang diberikan oleh Komdigi, sehingga menunjukkan kemampuan sistem dalam mengenali karakteristik berita hoaks pada sumber yang telah diverifikasi. Selain hasil klasifikasi, sistem juga menampilkan *evidence retrieval* berupa sumber informasi pendukung dari pihak eksternal. Hasil ini menunjukkan bahwa proses input url, *web scraping*, klasifikasi model, dan pencarian *evidence* dapat berjalan secara terintegrasi dalam sistem yang dikembangkan.

Tabel 5. Pengujian Sistem

No	Skenario Pengujian	Hasil
1	Input teks berita	Berhasil
2	Input url berita	Berhasil
3	Prediksi CNN-LSTM	Berhasil
4	Prediksi IndoBERT	Berhasil
5	Evidence Retrieval	Berhasil
6	Penyimpanan Riwayat	Berhasil
7	Detail History Deteksi	Berhasil

Berdasarkan hasil pengujian pada Tabel Pengujian Sistem, seluruh fungsi utama sistem berhasil dijalankan tanpa ditemukan kegagalan proses. Dengan demikian, tingkat keberhasilan pengujian fungsional mencapai 100%, yang menunjukkan bahwa sistem telah memenuhi kebutuhan fungsional yang ditetapkan pada penelitian ini.

5. KESIMPULAN DAN SARAN

Penelitian ini berhasil mengembangkan sistem deteksi berita hoaks berbahasa Indonesia menggunakan dua pendekatan klasifikasi, yaitu CNN-LSTM dan IndoBERT, yang diintegrasikan ke dalam aplikasi web berbasis *Flask* serta didukung oleh fitur *evidence retrieval*. Dataset yang digunakan terdiri dari 21.130 berita hasil yang kemudian dibagi menjadi *data training*, *validation*, dan *testing* dengan rasio 70:15:15. Hasil evaluasi menunjukkan bahwa kedua model mampu memberikan performa yang tinggi dalam mendeteksi berita hoaks. Model CNN-LSTM memperoleh *accuracy* sebesar 97,32%, dengan nilai *precision*, *recall*, dan *F1-score* masing-masing sebesar 97%. Sementara itu, model IndoBERT menghasilkan performa yang lebih baik dengan *accuracy* sebesar 99,12% serta nilai *precision*, *recall*, dan *F1-score* sebesar 99%. Selain memiliki tingkat akurasi yang lebih tinggi, IndoBERT juga menghasilkan tingkat kesalahan klasifikasi yang lebih rendah, yaitu 0,88%, dibandingkan CNN-LSTM sebesar 2,68%.

Hasil penelitian menunjukkan bahwa pendekatan berbasis *Transformer* melalui IndoBERT lebih efektif dalam memahami konteks dan makna semantik pada teks berita berbahasa Indonesia dibandingkan pendekatan CNN-LSTM. Integrasi model ke dalam sistem web juga berhasil dilakukan dengan baik, sehingga pengguna dapat melakukan deteksi berita melalui *input* teks maupun url berita secara otomatis.

Penelitian selanjutnya dapat mengembangkan sistem deteksi berita hoaks yang mampu melakukan pembelajaran secara berkelanjutan (*Continuous Learning*) atau *Incremental Learning*. Pada pendekatan ini, berita yang telah melalui proses deteksi dan verifikasi dapat disimpan sebagai data baru untuk memperkaya *dataset* yang telah ada. Selanjutnya, sistem dapat melakukan proses pelatihan ulang secara terjadwal sehingga model selalu diperbarui berdasarkan pola, topik, dan karakteristik hoaks terbaru yang beredar di masyarakat. Pendekatan tersebut berpotensi meningkatkan kemampuan adaptasi model terhadap perubahan informasi yang terjadi secara dinamis, mengingat pola penyebaran hoaks terus berkembang dari waktu ke waktu. Dengan adanya mekanisme pembaruan *dataset* dan model secara berkelanjutan, sistem diharapkan mampu mempertahankan tingkat akurasi klasifikasi yang tinggi serta mengurangi penurunan performa akibat munculnya jenis hoaks baru yang belum pernah dipelajari sebelumnya.

DAFTAR PUSTAKA

- [1] M. Candra and M. R. K. Dinata, "Penegakan Hukum terhadap Tindak Pidana Penyebaran Berita Hoaks melalui Media Sosial," *Arus Jurnal Sosial Dan Humaniora*, vol. 5, no. 2, pp. 1577–1586, 2025.
- [2] M. Y. Ridho and E. Yulianti, "From Text to Truth: Leveraging IndoBERT and Machine Learning Models for Hoax Detection in Indonesian News," *vol*, vol. 10, pp. 544–555, 2024.
- [3] F. Salim and A. I. Wahyudhy, "Klasifikasi Berita Palsu Menggunakan Pendekatan Hybrid CNN-LSTM," *Jurnal CoSciTech (Computer Science and Information Technology)*, vol. 6, no. 1, pp. 55–59, 2025.
- [4] H. Liao *et al.*, "Muser: A multi-step *evidence retrieval* enhancement framework for fake news detection," in *Proceedings of the 29th ACM SIGKDD conference on knowledge discovery and data mining*, 2023, pp. 4461–4472.
- [5] C. J. L. Tobing, I. G. N. L. Wijayakusuma, and L. P. I. Harini, "Perbandingan Kinerja IndoBERT dan MBERT untuk Deteksi Berita Hoaks Politik dalam Bahasa Indonesia," *JST (Jurnal Sains dan Teknologi)*, vol. 14, no. 1, pp. 114–123, 2025.
- [6] L. A. Pekandi, R. G. Widjaja, A. Ananta, J. Harefa, and K. Jingga, "Evaluating IndoBERT for Indonesian Hoax News Detection: A Comparative Study with Ensemble and CNN-LSTM Models," *Procedia Comput. Sci.*, vol. 269, pp. 1625–1633, 2025.
- [7] S. K. Hamed, M. J. Ab Aziz, and M. R. Yaakub, "A review of fake news detection approaches: A critical analysis of relevant studies and highlighting key challenges associated with the *dataset*, feature representation, and data fusion," *Heliyon*, vol. 9, no. 10, p. e20382, 2023, doi: <https://doi.org/10.1016/j.heliyon.2023.e20382>
- [8] C. Jocelynne, I. G. N. L. Wijayakusuma, and L. P. I. Harini, "Detection of Political Hoax News Using Fine-Tuning IndoBERT," *Journal of Applied Informatics and Computing*, vol. 9, no. 2, pp. 354–360, 2025.
- [9] S. M. Isa, G. Nico, and M. Permana, "Indobert for indonesian fake news detection," *ICIC Express Letters*, vol. 16, no. 3, pp. 289–297, 2022.
- [10] A. R. Hakim and A. H. Muhammad, "PERBANDINGAN MODEL TRANSFORMER, DEEP LEARNING, DAN MACHINE LEARNING UNTUK DETEKSI BERITA PALSU: STUDI KASUS PADA TEKS BERBAHASA INDONESIA," *Jurnal Manajemen Informatika dan Sistem Informasi*, vol. 8, no. 2, pp. 188–197, 2025.
- [11] M. Agbabiaka, E. Ogbuju, and F. Oladipo, "A Systematic Review of Deep Learning Approaches for Fake News Detection," *African Journal of Advances in Science and Technology Research*, vol. 21, no. 1, pp. 1–17, 2025.
- [12] P. Deka, A. Jurek-Loughrey, and D. P., "Evidence extraction to validate medical claims in fake news detection," in *International conference on health information science*, Springer, 2022, pp. 3–15.