

## BAB 2

### LANDASAN TEORI

#### 2.1 Penelitian Sebelumnya

Tabel 2.1. Tabel Penelitian Terdahulu

Penulis	Judul	Jurnal	Metode	Hasil
Kaliyar, R.K., et al. (2021)	Fake News Detection in Social Media with a BERT-Based Deep Learning Approach	Multimedia Tools and Applications (Springer) [7]	Fine-tuning BERT untuk klasifikasi hoax	BERT mencapai peningkatan dengan akurasi sebesar 98% dibandingkan metode tradisional.
Ramzan, A., Ali, R.H., Ali, N. (2024)	Enhancing Fake News Detection Using BERT: A Comparative Analysis	IEEE ICIT Conference [8]	Perbandingan BERT dengan <i>Machine Learning</i> klasik	BERT unggul signifikan pada akurasi dan F1-score dibanding SVM, LR, dan LSTM.

Penulis	Judul	Jurnal	Metode	Hasil
Samadi, M., et al. (2023)	Fake News Detection: Deep Semantic Representation with Content-Based Features	Springer Journal [9]	Model BERT-like + fitur linguistik tambahan	Peningkatan sebesar 4,16% (akurasi) dan 4,02% (F1) pada dataset COVID-19 Inggris, serta 2,13% (akurasi) dan 1,6% (F1) pada dataset TAJ Persia dibanding model biasa
Ni'mah, I., et al. (2025)	A Simple Contrastive Embedding Framework for Low-Resource Fake News Detection (Contrast-BERT)	Soft Computing (Springer) [10]	Contrastive learning + BERT encoder	Peningkatan dalam deteksi dataset rendah sumber daya ditunjukkan dengan <i>precision</i> naik hingga 26,64%, serta akurasi dan recall tetap di atas 75% meski terdapat imbalance.

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

Penulis	Judul	Jurnal	Metode	Hasil
Suresh, S. (2025)	Transforming Fake News Detection: Leveraging DistilBERT for Efficient Classification	Procedia Computer Science [11]	Fine-tuning DistilBERT	DistilBERT base-cased-distilled-squad mencapai akurasi 0,977 dan F1-Score 0,97 dengan komputasi lebih rendah dibanding BERT.

Penelitian ini dilakukan dengan menggunakan metode yang sudah pernah diterapkan, namun dengan objek dan studi kasus yang berbeda. Penelitian ini memanfaatkan gabungan tiga bahasa, dengan dua di antaranya tergolong bahasa rendah sumber daya dalam konteks deteksi *hoax*, yaitu Indonesia, Inggris, dan Melayu. Untuk klasifikasi berita *Hoax* atau *Real*, penelitian ini menggunakan DistilBERT. Berbeda dengan penelitian-penelitian sebelumnya, model ini diterapkan secara khusus pada dataset multibahasa tersebut untuk mengeksplorasi kemampuan DistilBERT dalam menangani variasi linguistik dari ketiga bahasa sekaligus terutama pada bahasa Asia Tenggara yang tergolong rendah sumber dayanya, yaitu Indonesia dan Melayu.

## 2.2 Hoax

Berita Bohong atau biasa disebut *hoax* dalam bahasa Inggris dapat diartikan sebagai suatu informasi palsu yang disebarluaskan secara luas dengan adanya tujuan atau maksud tertentu. Chen, Yong, & Ishak (2014) menyatakan bahwa *hoax* merupakan informasi yang dapat menipu persepsi manusia. Hoax mampu memengaruhi banyak orang hingga merusak citra dan kredibilitas korban [12].

Menurut Tajrian, Rahman, Kabir, & Islam (2023), *hoax* pada umumnya memiliki karakteristik bahasa yang khas. Ciri-ciri tersebut antara lain penggunaan bahasa yang bersifat provokatif, ketidakjelasan sumber informasi, penggunaan struktur kalimat yang dirancang untuk memanipulasi persepsi pembaca, serta

menggunakan gaya retorika yang dapat menimbulkan ketegangan atau keraguan, sehingga memperkuat dampak persuasif dari informasi yang disampaikan [13]. Penyebaran *hoax* ini dapat mengakibatkan timbulnya rasa takut, dan kebingungan pada masyarakat, serta mengakibatkan terjadinya perpecahan yang memungkinkan terjadinya kerusuhan antar masyarakat [14].

Menurut Wardle dan Derakhshan [15], *hoax* dapat diklasifikasikan menjadi tiga, yaitu misinformasi, disinformasi, dan malinformasi. Misinformasi merupakan penyebaran informasi yang salah tanpa adanya unsur kejahatan. Disinformasi merupakan penyebaran informasi yang salah secara sengaja dengan adanya tujuan untuk menipu. Malinformasi merupakan penyebaran informasi namun disebarluaskan dengan tujuan untuk merugikan pihak tertentu.

### 2.3 Data Preprocessing

*Data preprocessing* merupakan tahapan awal yang penting pada proses analisis data, khususnya pada bidang *Natural Language Processing* (NLP). Tujuan utamanya adalah untuk membersihkan, dan menyiapkan data yang mentah supaya dapat diolah secara efektif oleh model yang digunakan. Menurut Camacho-Collados dan Pilehvar [16], proses ini membantu mengurangi *noise*, menghapus duplikat, serta menstandarkan format teks sehingga model dapat mengenali pola bahasa dengan lebih akurat.

Tahapan *data preprocessing* umumnya meliputi beberapa proses utama seperti *text cleaning*, *tokenization*, *stopword removal*, *stemming* atau *lemmatization*, serta *vectorization*. Pada tahap *text cleaning*, dilakukan penghapusan karakter yang tidak relevan seperti menghapus tanda baca, angka, simbol khusus, dan URL. Selanjutnya, proses *tokenization* memecahkan teks menjadi suatu unit kata atau sub-kata agar dapat dipelajari secara terstruktur oleh model. Proses *stopword removal* dilakukan untuk menghapus kata umum seperti “dan”, “yang”, dan “atau” yang tidak memberikan informasi penting terhadap konteks teks. Kemudian, tahap *stemming* atau *lemmatization* digunakan untuk mengembalikan kata ke bentuk dasarnya agar mengurangi variasi kata yang berlebihan [17].

Dalam konteks pemodelan berbasis *transformer*, seperti BERT dan DistilBERT, tahap tambahan seperti *padding* dan *truncation* juga diterapkan untuk menyamakan panjang urutan teks sebelum dimasukkan ke dalam model. Penelitian oleh Siino, Tinnirello, dan La ca [18] menunjukkan bahwa meskipun model modern seperti *transformers* mampu memahami konteks secara mendalam,

proses *preprocessing* tetap memiliki pengaruh signifikan terhadap performa model, terutama dalam menjaga kualitas dan konsistensi input teks. Dengan demikian, penerapan tahapan *data preprocessing* yang tepat menjadi fondasi utama bagi keberhasilan proses pelatihan dan evaluasi model deteksi berita *hoax* berbasis teks.

## 2.4 Natural Languange Processing

*Natural Language Processing* (NLP) merupakan cabang *artificial intelligence* yang memungkinkan komputer untuk mempelajari, memahami, menganalisis, serta menghasilkan bahasa manusia secara efektif. Proses ini menggabungkan prinsip linguistik, metode statistik, dan teknik pembelajaran mesin yang saling mendukung guna mengekstraksi makna serta informasi dari teks maupun ujaran dalam bahasa alami [19].

Menurut Sharma dan Nagashree (2022), NLP tidak hanya dapat memproses kata per kata, namun juga memahami makna kontekstual secara menyeluruh, sehingga meningkatkan akurasi dalam berbagai aplikasi seperti *sentiment analysis*, *machine translation*, dan *fake news detection* [20].

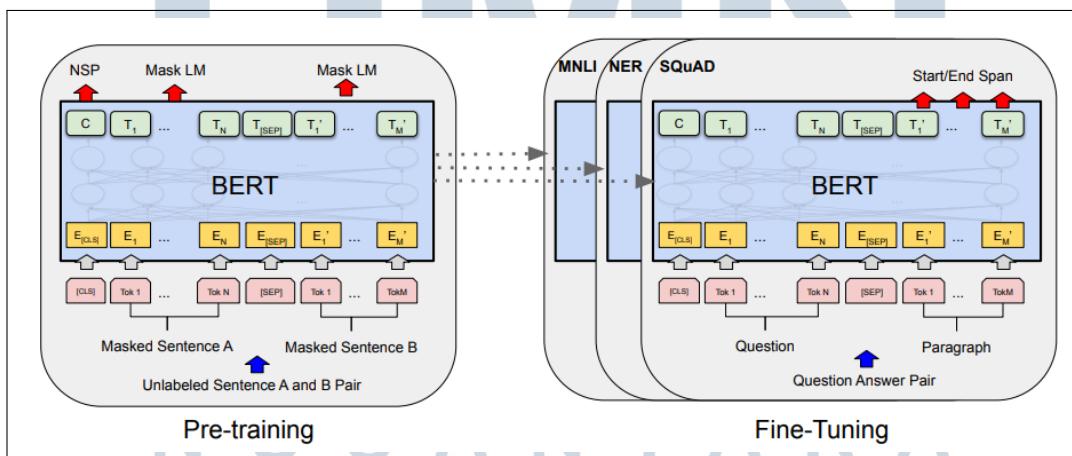
## 2.5 BERT

BERT atau *Bidirectional Encoder Representations from Transformers* adalah model *Natural Language Processing* yang dibuat oleh Google AI pada tahun 2018. Model ini menggunakan arsitektur *Transformer* yang untuk mempelajari representasi konteks dari teks secara dua arah (bidirectional), yaitu dari kiri ke kanan dan dari kanan ke kiri secara bersamaan [21]. Ini memungkinkan BERT untuk memahami makna kata secara lebih kontekstual dalam kalimat.

Input	[CLS]	I	like	dogs	[SEP]	he	likes	cats	[SEP]
<hr/>									
Token Embedding	E <sub>[CLS]</sub>	E <sub>I</sub>	E <sub>like</sub>	E <sub>dogs</sub>	E <sub>[SEP]</sub>	E <sub>he</sub>	E <sub>likes</sub>	E <sub>cats</sub>	E <sub>[SEP]</sub>
	+	+	+	+	+	+	+	+	+
Segment Embedding	E <sub>A</sub>	E <sub>A</sub>	E <sub>A</sub>	E <sub>A</sub>	E <sub>A</sub>	E <sub>B</sub>	E <sub>B</sub>	E <sub>B</sub>	E <sub>B</sub>
	+	+	+	+	+	+	+	+	+
Positions Embedding	E <sub>0</sub>	E <sub>1</sub>	E <sub>2</sub>	E <sub>3</sub>	E <sub>4</sub>	E <sub>5</sub>	E <sub>6</sub>	E <sub>7</sub>	E <sub>8</sub>

Gambar 2.1. Cara kerja tokenisasi BERT

Gambar 2.1 menunjukkan *Transformer blocks* model BERT dengan *hidden size* 768 dan 12–16 *self-attention heads*. BERT memproses teks melalui *tokenization* yang menghasilkan *classification token* [CLS] di awal sebagai representasi kalimat, dan *separate token* [SEP] sebagai pemisah antar kalimat. Setiap token direpresentasikan oleh tiga jenis embeddings: *token embedding*, *segment embedding*, dan *position embedding*. *Token embedding* memetakan kata ke dalam vektor, segment embedding membedakan kalimat pertama dan kedua (dengan nilai 0 dan 1), sedangkan position embedding menyimpan urutan posisi token dalam kalimat. Kombinasi ketiganya memungkinkan BERT memahami konteks kata secara menyeluruh dalam satu kalimat [22].



Gambar 2.2. Tahap *pre-trained & fine-tuning* BERT

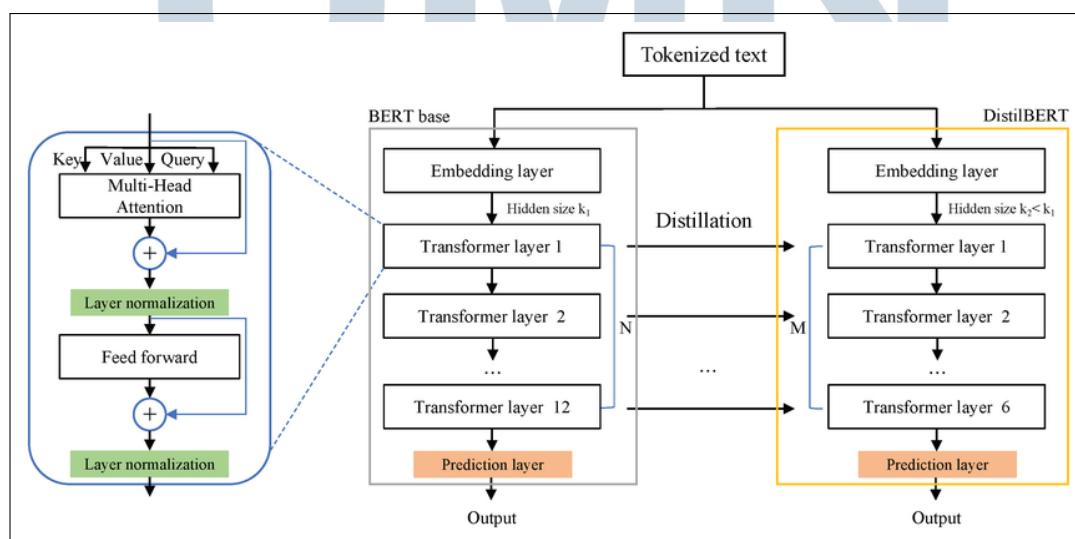
Gambar 2.2 menunjukkan proses pembelajaran BERT yang terdiri atas dua tahap, yaitu *pre-training* dan *fine-tuning*. Saat tahap pre-training, BERT dilatih

menggunakan dua tugas utama: *Masked Language Modeling* (MLM) dan *Next Sentence Prediction* (NSP). Dalam MLM, sebagian kata dalam kalimat diganti dengan token [MASK], dan model dilatih untuk memprediksi kata yang hilang berdasarkan konteks sekitarnya. Hal ini memungkinkan BERT memahami relasi semantik antar kata secara mendalam. Sementara pada NSP, model diberikan dua kalimat dan diminta menebak apakah kalimat kedua merupakan kelanjutan logis dari kalimat pertama, sehingga BERT mampu memahami hubungan antar kalimat [21].

Tahap selanjutnya adalah fine-tuning, di mana model BERT yang telah melalui pre-training disesuaikan dengan tugas spesifik, seperti *text classification*, *Named Entity Recognition* (NER), atau *Question Answering* (QA). Pada tahap ini, semua parameter BERT diperbarui menggunakan dataset yang lebih kecil dan sesuai dengan domain yang diinginkan, tanpa mengubah struktur arsitektur model. Dengan strategi dua tahap ini, BERT dapat mencapai performa tinggi di berbagai tugas NLP karena kemampuannya menghasilkan representasi konteks kata yang kaya dan mendalam [21].

## 2.6 DistilBERT

DistilBERT merupakan versi ringan dan efisien dari BERT yang dikembangkan oleh tim Hugging Face pada tahun 2019. Model ini dibangun dengan menggunakan teknik *knowledge distillation*, yaitu proses di mana *teacher model*, mentransfer pengetahuannya ke *student model*, tanpa kehilangan banyak performa [23].



Gambar 2.3. Arsitektur dari DistilBERT

Gambar 2.3 menunjukkan arsitektur untuk model DistilBERT. Pada sisi kiri, terlihat arsitektur BERT yang terdiri atas beberapa komponen utama, yaitu *Embedding Layer* yang berfungsi mengubah teks hasil tokenisasi menjadi representasi vektor, 12 lapisan *Transformer* dengan ukuran *hidden state* sebesar  $k_1$ , serta *Prediction Layer* yang menghasilkan keluaran berdasarkan representasi akhir dari lapisan *Transformer*. Sementara itu, di sisi kanan ditampilkan DistilBERT, yang merupakan versi lebih ringkas dari BERT dengan struktur serupa, namun hanya memiliki enam lapisan *Transformer* dengan ukuran *hidden state*  $k_2$  yang lebih kecil dibandingkan  $k_1$ , dan tetap dilengkapi dengan *Prediction Layer* yang menghasilkan keluaran setara dengan model aslinya.

Arsitektur DistilBERT mempertahankan sebagian besar struktur BERT namun hanya menggunakan 6 lapisan *Transformer*, setengah dari jumlah lapisan pada BERT Base yang memiliki 12 lapisan. DistilBERT mampu mempertahankan sekitar 97% performa BERT pada berbagai tugas NLP seperti *text classification*, *sentiment analysis*, dan *question answering*, namun dengan kecepatan inferensi yang lebih tinggi dan ukuran model yang lebih efisien [23].

## 2.7 Evaluasi Metrik

Evaluasi metrik adalah alat ukur yang digunakan untuk mengukur kinerja model dalam melakukan klasifikasi [24]. Dalam penelitian mengenai deteksi *hoax*, evaluasi metrik digunakan untuk menilai efektivitas model dalam membedakan antara berita yang asli dan berita yang palsu. Beberapa metrik yang umum digunakan antara lain *accuracy*, *precision*, *recall*, dan *F1-score*.

1. *Accuracy* merupakan metrik yang mengukur prediksi model yang benar terhadap seluruh data uji.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (2.1)$$

2. *Precision* merupakan metrik yang mengukur ketepatan model dalam mengklasifikasikan data positif secara benar.

$$\text{Precision} = \frac{TP}{TP + FP} \quad (2.2)$$

3. *Recall* merupakan metrik yang mengukur kemampuan model dalam menemukan seluruh data positif yang sebenarnya.

$$Recall = \frac{TP}{TP + FN} \quad (2.3)$$

4. *F1-score* merupakan metrik yang menunjukkan nilai rata-rata harmonis yang memberikan keseimbangan antara keduanya dari *precision* dan *recall*.

$$F1\text{-Score} = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (2.4)$$

Evaluasi Metrik yang didapatkan berasal dari *confusion matrix*, yang sekaligus juga menjadi dasar untuk memahami kinerja model dalam mengklasifikasi berita asli atau berita palsu.

## 2.8 Confusion Matrix

*Confusion Matrix* merupakan tabel yang menampilkan perbandingan antara nilai prediksi dengan nilai sebenarnya yang dihasilkan oleh model setelah proses pelatihan [25]. Tabel 2.2 menunjukkan struktur *confusion matrix* yang biasa digunakan untuk penelitian mengenai deteksi *hoax*:

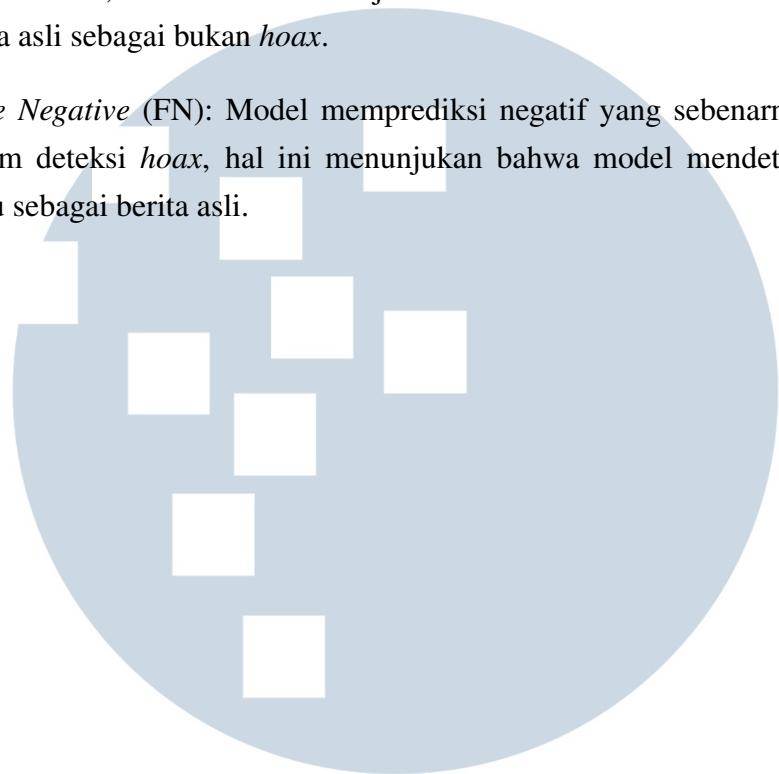
Tabel 2.2. Struktur *confusion matrix*

	Prediksi <i>hoax</i>	Prediksi Asli
Aktual <i>hoax</i>	True Positive (TP)	False Negative (FN)
Aktual Asli	False Positive (FP)	True Negative (TN)

Keterangan :

- *True Positive* (TP): Model memprediksi positif dan benar positif. Dalam deteksi *hoax*, ini mengartikan bahwa model berhasil mengidentifikasi berita palsu sebagai *hoax*.
- *False Positive* (FP): Model memprediksi positif yang sebenarnya negatif. Dalam deteksi *hoax*, hal ini berarti model salah mengklasifikasikan berita yang asli sebagai *hoax*.

- *True Negative* (TN): Model memprediksi negatif dan benar negatif. Dalam deteksi *hoax*, kondisi ini menunjukkan bahwa model berhasil mengenali berita asli sebagai bukan *hoax*.
- *False Negative* (FN): Model memprediksi negatif yang sebenarnya positif. Dalam deteksi *hoax*, hal ini menunjukkan bahwa model mendeteksi berita palsu sebagai berita asli.



UMN  
UNIVERSITAS  
MULTIMEDIA  
NUSANTARA