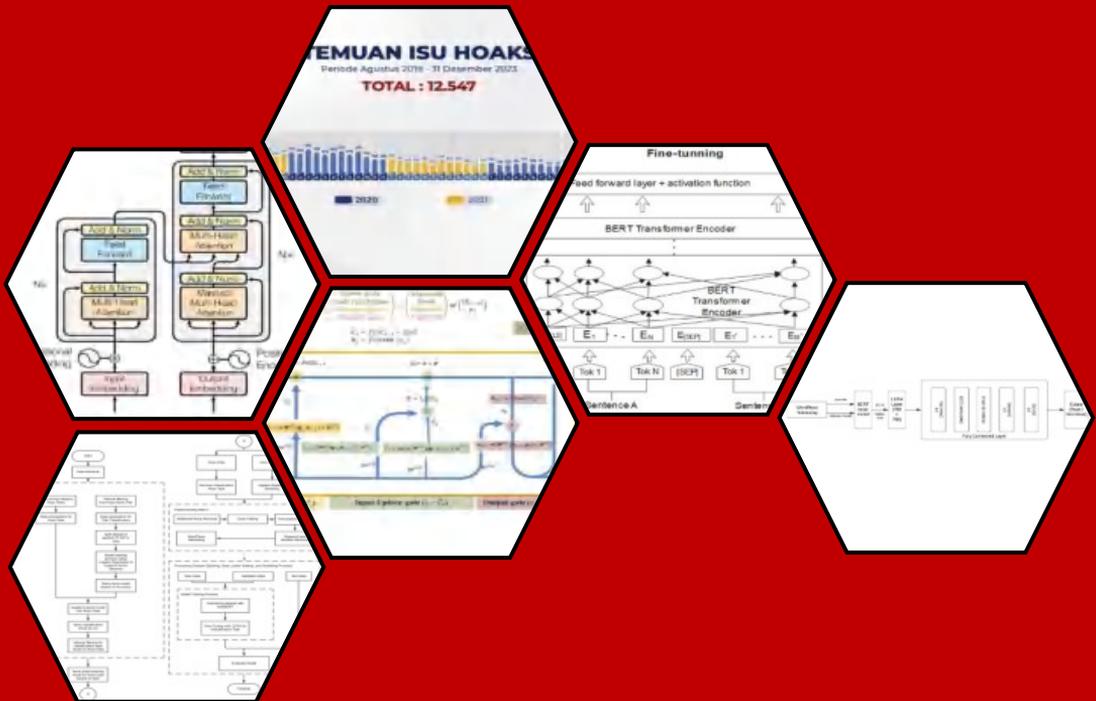


PENERAPAN MODEL INDOBERT-LSTM UNTUK KLASIFIKASI BERITA HOAX BERBAHASA INDONESIA



HARDIANTO TANDI SENO
D121201035



PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS HASANUDDIN
MAKASSAR
2024



Optimized using
trial version
www.balesio.com

**PENERAPAN MODEL INDOBERT-LSTM UNTUK KLASIFIKASI BERITA
HOAX BERBAHASA INDONESIA**

HARDIANTO TANDI SENO

D121 20 1035



PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS HASANUDDIN

MAKASSAR

2024



Optimized using
trial version
www.balesio.com

**PENERAPAN MODEL INDOBERT-LSTM UNTUK KLASIFIKASI BERITA
HOAX BERBAHASA INDONESIA**

HARDIANTO TANDI SENO

D121 20 1035

Skripsi

sebagai salah satu syarat untuk mencapai gelar sarjana

Program Studi Teknik Informatika

pada

PROGRAM STUDI TEKNIK INFORMATIKA

EPARTEMEN TEKNIK INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS HASANUDDIN

MAKASSAR

2024



SKRIPSI

PENERAPAN MODEL INDOBERT-LSTM UNTUK KLASIFIKASI BERITA HOAX
BERBAHASA INDONESIA

HARDIANTO TANDI SENO

D121201035

Skripsi,

telah dipertahankan di depan Panitia Ujian Sarjana Teknik Informatika pada 02
Oktober 2024 dan dinyatakan telah memenuhi syarat kelulusan
pada

Program Studi Teknik Informatika
Departemen Teknik Informatika
Fakultas Teknik
Universitas Hasanuddin
Makassar

Mengesahkan:

Pembimbing tugas akhir,

Mengetahui:

Ketua Program Studi,



min, S.T., M.T.

07 4 001



Prof. Dr. Ir. Indrabayu, S. T., M. T., M.

Bus., Sys., IPM, ASEAN. Eng.

NIP 19750716 200212 1 004

PERNYATAAN KEASLIAN SKRIPSI DAN PELIMPAHAN HAK CIPTA

Dengan ini saya menyatakan bahwa, skripsi berjudul "Penerapan Model IndoBERT-LSTM untuk Klasifikasi Berita *Hoax* Berbahasa Indonesia" adalah benar karya saya dengan arahan dari pembimbing Ir. Anugrayani Bustamin, S.T., M.T. sebagai pembimbing utama. Karya ilmiah ini belum diajukan dan tidak sedang diajukan dalam bentuk apa pun kepada perguruan tinggi mana pun. Sumber informasi yang berasal atau dikutip dari karya yang diterbitkan maupun tidak diterbitkan dari penulis lain telah disebutkan dalam teks dan dicantumkan dalam Daftar Pustaka skripsi ini. Apabila di kemudian hari terbukti atau dapat dibuktikan bahwa sebagian atau keseluruhan skripsi ini adalah karya orang lain, maka saya bersedia menerima sanksi atas perbuatan tersebut berdasarkan aturan yang berlaku.

Dengan ini saya melimpahkan hak cipta (hak ekonomis) dari karya tulis saya berupa skripsi ini kepada Universitas Hasanuddin.

Makassar, 02 Oktober 2024



HARDIANTO TANDI SENO
NIM D121201035



Optimized using
trial version
www.balesio.com

Ucapan Terima Kasih

Puji syukur kepada Tuhan Yesus Kristus karena atas kasih karunia dan anugerah-Nya yang telah Ia limpahkan kepada penulis sehingga dimampukan untuk dapat menyelesaikan skripsi yang berjudul “Penerapan Model IndoBERT-LSTM untuk Klasifikasi Berita *Hoax* Berbahasa Indonesia” dengan baik sebagai salah satu syarat untuk bisa menyelesaikan jenjang Strata-1 di Departemen Teknik Informatika, Fakultas Teknik, Universitas Hasanuddin.

Selama proses penyusunan skripsi ini, banyak pihak yang telah memberikan dukungan, bantuan, dan bimbingan sehingga skripsi ini akhirnya dapat terselesaikan. Oleh karena itu, ucapan terima kasih yang mendalam penulis sampaikan kepada:

1. Tuhan Yesus sebagai satu-satunya Tuhan, Raja, dan Juruslamat atas segala kasih dan penyertaan yang selalu nyata dalam titik terbaik hingga titik terendah kehidupan penulis.
2. Orang tua penulis, Bapak Titus Tandi Seno dan Ibu Albertin Palinggi yang selalu mendoakan penulis di setiap waktu yang dimiliki, selalu memberikan kasih sayang, dukungan, motivasi, dan mengajarkan arti kehidupan yang tidak pernah berhenti kepada penulis.
3. Ibu Ir. Anugrayani Bustamin S.T., M.T. sebagai pembimbing yang senantiasa meluangkan waktu, tenaga, dan pikiran untuk bisa selalu memberikan bimbingan dan arahan yang sangat berharga kepada penulis selama proses penyelesaian skripsi.
4. Segenap dosen dan staf Departemen Teknik Informatika, Fakultas Teknik, Universitas Hasanuddin yang telah memberikan setiap ilmu pengetahuan, pengalaman, dan membantu penulis selama masa perkuliahan.
5. Saudara penulis, Eka dan Sela yang juga selalu mendampingi, memberikan semangat, dan mendoakan penulis selama ini.
6. Teman-teman PPGT Jemaat Satria Kasih, khususnya teman-teman pengurus dan penghuni sekret untuk setiap doa, kebersamaan, dan dukungan yang diberikan kepada penulis.
7. Seluruh teman-teman Rezolver 20 yang telah bersama-sama dengan penulis menghadapi berbagai keadaan (suka dan duka) selama masa perkuliahan.
8. Seluruh pihak yang tidak dapat disebutkan satu per satu yang tanpa sadar telah menjadi inspirasi dan membantu penulis selama penyelesaian skripsi.

Akhir kata, penulis menyadari bahwa penulisan skripsi ini masih jauh dari sempurna sehingga penulis mengharapkan kritik dan saran yang membangun agar

lebih baik lagi dan memberi manfaat bagi pembaca untuk pengetahuan yang akan datang.



Penulis,

Hardianto Tandi Seno

ABSTRAK

HARDIANTO TANDI SENO. **Penerapan Model IndoBERT-LSTM untuk Klasifikasi Berita Hoax Berbahasa Indonesia** (dibimbing oleh Anugrayani Bustamin).

Latar belakang. Berdasarkan data penyebaran isu *hoax* yang ditemukan oleh KOMINFO pada periode Agustus 2018 – 31 Desember 2023, terdapat 12.547 konten isu *hoax* yang tersebar di media digital. Selain itu, tingkat literasi di Indonesia masih rendah yang berdampak pada kemampuan untuk mengevaluasi informasi dan membedakan suatu berita itu *hoax* atau tidak masih kurang. **Tujuan.** Penelitian ini bertujuan untuk mengetahui cara mengimplementasikan dan tingkat performa model IndoBERT-LSTM dalam mengklasifikasikan berita *hoax*. **Metode.** Penelitian dilakukan dengan beberapa tahapan utama, seperti pengumpulan *dataset* berita, proses *preprocessing data* pada berita, perancangan model IndoBERT-LSTM, dan evaluasi serta pengujian terhadap hasil pemodelan. **Hasil.** Penelitian dilakukan dengan beberapa skenario penggunaan *Dropout Rate* (0.2 dan 0.6) dan *Batch Size* (32 dan 64). Pada *dataset* gabungan (Politik & Pemerintahan, Kesehatan) dengan *Dropout Rate* 0.6 dan *Batch Size* 64, diperoleh nilai *Loss* 0.1174, *Accuracy* 0.9731, *Precision* 0.9749, *Recall* 0.9719, dan *F1-Score* 0.9734. Pada *dataset* topik Politik & Pemerintahan dengan *Dropout Rate* 0.6 dan *Batch Size* 32, diperoleh nilai *Loss* 0.0694, *Accuracy* 0.9812, *Precision* 0.9915, *Recall* 0.9710, dan *F1-Score* 0.9811. Pada *dataset* topik Kesehatan dengan *Dropout Rate* 0.2 dan *Batch Size* 32, diperoleh nilai *Loss* 0.3675, *Accuracy* 0.9256, *Precision* 0.8807, *Recall* 0.9873, dan *F1-Score* 0.9309. **Kesimpulan.** IndoBERT-LSTM dapat diimplementasikan untuk tugas klasifikasi berita *hoax* berbahasa Indonesia dengan hasil tertinggi pada *dataset* topik Politik & Pemerintahan berdasarkan jumlah data dan konteks data yang lebih homogen. Model yang dilatih dengan *dataset* gabungan kedua topik memberikan hasil evaluasi di atas 0.97 dan nilai *Loss* di bawah 0.2 meskipun ada perbedaan konteks antara kedua topik.

Kata kunci: *hoax*; klasifikasi; IndoBERT-LSTM; *Dropout Rate*; *Batch Size*



ABSTRACT

HARDIANTO TANDI SENO. **Implementation of IndoBERT-LSTM Model for Indonesian Language Hoax News Classification** (supervised by Anugrayani Bustamin).

Background. Based on spread of hoax issues data found by KOMINFO during August 2018 – December 31, 2023, there are 12.547 hoax issue content spread on digital media. Besides, literacy level in Indonesia is low which affects ability to evaluate information and distinguish news from hoax or not. **Purpose.** This research aims to find how to implement and performance IndoBERT-LSTM model in classifying hoax news. **Methods.** Research conducted in several stages, such as collection of news datasets, preprocessing data on news, design IndoBERT-LSTM models, and evaluation and testing of modelling results. **Result.** The research was conducted using several scenarios using Dropout Rate (0.2 and 0.6) and Batch Size (32 and 64). On combined datasets (Political & Governance, Health) with Dropout Rate 0.6 and Batch Size 64, obtain the Loss value 0.1174, Accuracy 0.9731, Precision 0.9749, Recall 0.9719, and F1-Score 0.9734. On datasets Political & Government topics with Dropout Rate 0.6 and Batch Size 32, obtain Loss value 0.0694, Accuracy 0.9812, Precision 0.9915, Recall 0.9710, and F1-Score 0.911. On Health topic datasets with Dropout Rate 0.2 and Batch Size 32, obtain Loss value 0.3675, Accuracy 0.9256, Precision 0.8807, Recall 0.9873, and F1-Score 0.9309. **Conclusion.** IndoBERT-LSTM can be implemented for Indonesian language hoax news classification task with highest result on Political & Government topic datasets based on large amounts of data and more homogeneous data contexts. A model trained with combined dataset of both topics provides evaluation results above 0.97 and Loss values below 0.2 despite the differences in context between two topics.

Keywords: hoax; classification; IndoBERT-LSTM; Dropout Rate; Batch Size



DAFTAR ISI

| | Halaman |
|--|---------|
| UCAPAN TERIMA KASIH | i |
| ABSTRAK | ii |
| ABSTRACT | iii |
| DAFTAR ISI | iv |
| DAFTAR TABEL | vi |
| DAFTAR GAMBAR | viii |
| DAFTAR LAMPIRAN | x |
| DAFTAR SINGKATAN, ISTILAH, DAN LAMBANG | xi |
| BAB I PENDAHULUAN..... | 1 |
| 1.1 Latar Belakang..... | 1 |
| 1.2 Landasan Teori..... | 4 |
| 1.2.1 Berita <i>Hoax</i> | 4 |
| 1.2.2 <i>Natural Language Processing (NLP)</i> | 5 |
| 1.2.3 <i>Text Classification</i> | 5 |
| 1.2.4 MAFINDO (Masyarakat Antifitnah Indonesia) | 5 |
| 1.2.5 IndoBERT (<i>Indonesia Bidirectional Encoder Representations from Transformers</i>)..... | 6 |
| 1.2.6 LSTM (<i>Long-Short Term Memory</i>) | 8 |
| 1.2.7 <i>Early Stopping</i> | 11 |
| 1.2.8 <i>Batch Normalization</i> dan <i>Dropout</i> | 11 |
| 1.2.9 API (<i>Application Programming Interface</i>) | 12 |
| 1.2.10 <i>Scraping</i> dengan <i>BeautifulSoup</i> | 12 |
| 1.2.11 <i>Scikit-learn</i> | 12 |
| 1.2.12 TF-IDF (<i>Term Frequency – Inverse Document Frequency</i>) | 13 |
| 1.2.13 <i>Stratified Sampling</i> | 13 |
| 1.2.14 <i>Hugging Face Transformers</i> | 13 |
| 1.2.15 <i>PyTorch</i> | 14 |
| 1.2.16 <i>Preprocessing Data</i> | 14 |
| 1.2.17 Metrik Evaluasi..... | 15 |
| 1.3 Rumusan Masalah | 16 |
| 1.4 Tujuan dan Manfaat | 16 |
| 1.5 Ruang Lingkup..... | 16 |
| BAB II METODE PENELITIAN..... | 17 |
| 2.1 Tahapan Penelitian | 17 |
| 2.2 Waktu dan Lokasi Penelitian..... | 17 |
| 2.3 Instrumen Penelitian | 17 |
| 2.4 Teknik Pengambilan Data..... | 18 |
| 2.5 Perancangan dan Implementasi Model IndoBERT-LSTM | 19 |
| Data Tahap 1 | 20 |
| Metode Analisis Klasifikasi Berita <i>Hoax</i> dan Pengaplikasian <i>Stratified Sampling</i> pada Berita <i>Non-Hoax</i> | 21 |
| Data Tahap 2 | 22 |
| Perancangan dan Implementasi Model dengan IndoBERT – LSTM | 23 |
| Evaluasi Model & Pengujian pada <i>Test Data</i> | 24 |
| PEMBAHASAN | 25 |
| Kesimpulan | 25 |



| | |
|---|----|
| 3.1.1 Pengambilan Data | 25 |
| 3.1.2 <i>Preprocessing Data</i> Tahap 1 | 25 |
| 3.1.3 Penghilangan Jenis Klasifikasi Berita <i>Hoax</i> dan Pengaplikasian <i>Stratified Sampling</i> pada Berita <i>Non-Hoax</i> | 32 |
| 3.1.4 <i>Preprocessing Data</i> Tahap 2 | 34 |
| 3.1.5 Proses Pelatihan Model dengan IndoBERT-LSTM | 52 |
| 3.2 Pembahasan..... | 59 |
| 3.2.1 Hasil Pelatihan dan Pengujian Model dengan <i>Dataset</i> Gabungan Kedua Topik (Topik Politik & Pemerintahan dan Kesehatan) | 59 |
| 3.2.2 Hasil Pelatihan dan Pengujian Model dengan <i>Dataset</i> topik Politik & Pemerintahan | 62 |
| 3.2.3 Hasil Pelatihan dan Pengujian Model dengan <i>Dataset</i> topik Kesehatan..... | 64 |
| 3.2.4 Interpretasi Data Hasil Pengujian Model | 66 |
| BAB IV KESIMPULAN DAN SARAN | 77 |
| 4.1 Kesimpulan | 77 |
| 4.2 Saran | 77 |
| DAFTAR PUSTAKA..... | 79 |
| LAMPIRAN | 83 |



DAFTAR TABEL

| Nomor urut | Halaman |
|---|---------|
| 1. Contoh judul berita non- <i>hoax</i> yang tidak sesuai dengan topik..... | 26 |
| 2. Perbandingan judul sebelum dan setelah persiapan data judul berita | 26 |
| 3. Sampel data yang diperoleh dari <i>dataset</i> Politik & Pemerintahan..... | 27 |
| 4. Perbandingan jumlah data sebelum dan sesudah penghapusan data duplikat .. | 27 |
| 5. Perbandingan sebelum dan sesudah pembersihan data judul berita non- <i>hoax</i> .. | 28 |
| 6. Hasil pembagian <i>dataset</i> | 29 |
| 7. Contoh hasil beberapa nilai tertinggi frekuensi kemunculan kata pada <i>training</i> dan <i>test data</i> dengan TF-IDF | 29 |
| 8. Hasil pemodelan dengan <i>Logistic Regression</i> dan SVM | 30 |
| 9. Contoh judul yang tidak sesuai dengan topik dan narasi yang tidak menggambarkan isi konten berita pada berita <i>hoax</i> | 31 |
| 10. Perbandingan jumlah data sebelum dan sesudah penyaringan pada masing-masing topik | 31 |
| 11. Hasil <i>Stratified Sampling</i> berdasarkan bulan terbit berita non- <i>hoax</i> pada topik Politik & Pemerintahan | 32 |
| 12. Hasil <i>Stratified Sampling</i> berdasarkan bulan terbit berita non- <i>hoax</i> pada topik Kesehatan | 33 |
| 13. Perbandingan teks awal dan teks terbaru setelah penghilangan <i>noise</i> tambahan | 34 |
| 14. Hasil dari proses <i>case folding</i> pada teks berita <i>hoax</i> dan non- <i>hoax</i> | 38 |
| 15. Perbandingan sebelum dan setelah melakukan <i>Punctuation Removal</i> pada teks narasi berita <i>hoax</i> dan non- <i>hoax</i> | 41 |
| 16. Hasil penghilangan <i>stopword</i> dan angka pada teks narasi berita <i>hoax</i> dan non- <i>hoax</i> | 45 |
| 17. Hasil <i>WordPiece Tokenizing</i> dalam teks narasi berita <i>hoax</i> dan non- <i>hoax</i> | 49 |
| 18. Proporsi pembagian <i>dataset</i> | 53 |
| 19. Skenario pemodelan yang akan dilakukan pada IndoBERT-LSTM | 53 |
| 20. Hasil pelatihan model pada seluruh skenario dengan <i>dataset</i> gabungan | 54 |
| asi pelatihan model pada seluruh skenario dengan 1 kedua topik | 55 |
| <i>matrix</i> pelatihan model pada seluruh skenario dengan 1 kedua topik | 55 |



| | |
|--|----|
| 23. Hasil pelatihan model pada seluruh skenario dengan <i>dataset</i> topik Politik & Pemerintahan..... | 55 |
| 24. Hasil metrik evaluasi pelatihan model pada seluruh skenario dengan <i>dataset</i> topik Politik & Pemerintahan | 55 |
| 25. Hasil <i>confusion matrix</i> pelatihan model pada seluruh skenario dengan <i>dataset</i> topik Politik & Pemerintahan | 56 |
| 26. Hasil pelatihan model pada seluruh skenario dengan <i>dataset</i> topik Kesehatan..... | 56 |
| 27. Hasil metrik evaluasi pelatihan model pada seluruh skenario dengan <i>dataset</i> topik Kesehatan | 56 |
| 28. Hasil <i>confusion matrix</i> pelatihan model pada seluruh skenario dengan <i>dataset</i> topik Kesehatan | 56 |
| 29. Hasil <i>Loss</i> dan <i>Accuracy</i> seluruh skenario pada pengujian model | 57 |
| 30. Metrik evaluasi hasil pengujian model pada seluruh skenario dengan <i>dataset</i> gabungan kedua topik..... | 57 |
| 31. <i>Confusion matrix</i> hasil pengujian model pada seluruh skenario dengan <i>dataset</i> gabungan kedua topik..... | 58 |
| 32. Metrik evaluasi hasil pengujian model pada seluruh skenario dengan <i>dataset</i> topik Politik & Pemerintahan | 58 |
| 33. <i>Confusion matrix</i> hasil pengujian model pada seluruh skenario dengan <i>dataset</i> topik Politik & Pemerintahan | 58 |
| 34. Metrik evaluasi hasil pengujian model pada seluruh skenario dengan <i>dataset</i> topik Kesehatan | 58 |
| 35. <i>Confusion matrix</i> hasil pengujian model pada seluruh skenario dengan <i>dataset</i> topik Kesehatan | 59 |
| 36. Contoh teks berita topik Politik & Pemerintahan dari hasil pengujian model dengan <i>Dropout Rate</i> 0.6 dan <i>Batch Size</i> 64 pada <i>dataset</i> gabungan kedua topik..... | 67 |
| 37. Contoh teks berita topik Kesehatan dari hasil pengujian model dengan <i>Dropout Rate</i> 0.6 dan <i>Batch Size</i> 64 pada <i>dataset</i> gabungan kedua topik..... | 72 |



DAFTAR GAMBAR

| Nomor urut | Halaman |
|---|---------|
| 1. Temuan isu hoaks KOMINFO periode Agustus 2018 – 31 Desember 2023..... | 1 |
| 2. Bentuk model BERT | 6 |
| 3. Struktur yang terdapat dalam <i>cell</i> pada model LSTM | 8 |
| 4. Alur <i>input gate</i> dengan dua bagian (Lapisan <i>sigmoid</i> dan Lapisan <i>tanh</i>)..... | 9 |
| 5. Proses <i>memory update</i> dengan <i>old memory</i> melalui <i>forget gate</i> dan <i>new memory</i> melalui <i>input gate</i> | 9 |
| 6. Alur <i>output</i> informasi dengan <i>sigmoid</i> σ dan <i>tanh</i> pada <i>output gate</i> | 10 |
| 7. Alur <i>forget gate</i> dalam penentuan informasi yang akan disimpan pada <i>cell state</i> | 11 |
| 8. Tahapan penelitian yang akan dilakukan | 17 |
| 9. Dokumentasi API milik MAFINDO | 18 |
| 10. Alur perancangan dan implemetasi model | 19 |
| 11. Alur pemodelan IndoBERT-LSTM..... | 23 |
| 12. Hasil penghapusan kategori berita <i>hoax</i> yang tidak sesuai pada kolom <i>classification</i> | 25 |
| 13. Hasil penghapusan kategori berita <i>hoax</i> yang tidak sesuai pada kolom status | 26 |
| 14. Hasil pengubahan label pada kolom klasifikasi berita sesuai dengan topik yang akan dilatih oleh model..... | 28 |
| 15. Pengambilan kolom yang diperlukan untuk proses pemodelan | 28 |
| 16. Hasil <i>encoding</i> pada kolom klasifikasi berita non- <i>hoax</i> | 29 |
| 17. Hasil penerapan model pada berita <i>hoax</i> untuk klasifikasi topik | 30 |
| 18. Hasil dari penghilangan beberapa jenis klasifikasi berita <i>hoax</i> pada topik Politik & Pemerintahan dan topik Kesehatan | 32 |
| 19. Hasil dari proses <i>encoding</i> kolom klasifikasi berita pada beberapa <i>dataset</i> | 52 |
| 20. Arsitektur IndoBERT-LSTM dengan penambahan <i>Batch Normalization</i> dan <i>Dropout</i> pada <i>Fully Connected Layer</i> | 53 |
| 21. Perbandingan hasil akhir pelatihan model pada seluruh skenario dengan kedua topik | 59 |
| 22. Perbandingan metrik evaluasi pelatihan model pada seluruh skenario dengan kedua topik | 60 |
| 23. Perbandingan hasil pengujian model pada seluruh skenario dengan kedua topik | 61 |



| | |
|--|----|
| 24. Perbandingan hasil akhir pelatihan model pada seluruh skenario dengan <i>dataset</i> topik Politik & Pemerintahan | 62 |
| 25. Perbandingan metrik evaluasi pelatihan model pada seluruh skenario dengan <i>dataset</i> topik Politik & Pemerintahan..... | 62 |
| 26. Perbandingan hasil pengujian model pada seluruh skenario dengan <i>dataset</i> topik Politik & Pemerintahan | 63 |
| 27. Perbandingan hasil akhir pelatihan model pada seluruh skenario dengan <i>dataset</i> topik Kesehatan | 64 |
| 28. Perbandingan metrik evaluasi pelatihan model pada seluruh skenario dengan <i>dataset</i> topik Kesehatan..... | 65 |
| 29. Perbandingan hasil pengujian model pada seluruh skenario dengan <i>dataset</i> topik Kesehatan | 66 |



DAFTAR LAMPIRAN

| Nomor urut | Halaman |
|---|---------|
| 1. Permohonan akses data melalui API pada organisasi MAFINDO | 83 |
| 2. Contoh hasil pengambilan data berita non- <i>hoax</i> | 84 |
| 3. Contoh hasil pengambilan data berita <i>hoax</i> | 85 |
| 4. Grafik <i>Loss</i> pada seluruh skenario dengan <i>dataset</i> gabungan kedua topik..... | 86 |
| 5. Grafik <i>Accuracy</i> pada seluruh skenario dengan <i>dataset</i> gabungan kedua topik | 86 |
| 6. Grafik <i>Loss</i> pada seluruh skenario dengan <i>dataset</i> topik Politik & Pemerintahan..... | 87 |
| 7. Grafik <i>Accuracy</i> pada seluruh skenario dengan <i>dataset</i> topik Politik & Pemerintahan..... | 87 |
| 8. Grafik <i>Loss</i> pada seluruh skenario dengan <i>dataset</i> topik Kesehatan | 88 |
| 9. Grafik <i>Accuracy</i> pada seluruh skenario dengan <i>dataset</i> topik Kesehatan | 88 |
| 10. <i>Source code</i> lengkap klasifikasi berita <i>hoax</i> berbahasa Indonesia dengan menggunakan IndoBERT-LSTM..... | 89 |



DAFTAR SINGKATAN, ISTILAH, DAN LAMBANG

| Lambang/singkatan | Arti dan Penjelasan |
|-------------------|--|
| NLP | <i>Natural Language Processing</i> |
| IndoBERT | <i>Indonesia Bidirectional Encoder Representation from Transformer</i> |
| MLM | <i>Masked Language Model</i> |
| NSP | <i>Next Sentence Prediction</i> |
| LSTM | <i>Long-Short Term Memory</i> |
| API | <i>Application Programming Interface</i> |
| TF-IDF | <i>Term Frequency – Inverse Document Frequency</i> |
| i_t | <i>Input Gate</i> |
| o_t | <i>Output Gate</i> |
| f_t | <i>Forget Gate</i> |
| C_t | <i>Cell State</i> |
| h_t | <i>Hidden State</i> |
| p_t | <i>Indeks current input timestep</i> |
| \tilde{C}_t | <i>Vektor nilai candidate</i> |
| \hat{y}_t | <i>Hasil prediksi model LSTM</i> |
| \odot | <i>Hadamard Product</i> |
| W | <i>Bobot matriks</i> |
| b | <i>Vektor bias</i> |
| σ | <i>Sigmoid</i> |
| n_i | <i>Jumlah sampel yang diambil dari strata i</i> |
| N_i | <i>Jumlah populasi dalam strata i</i> |
| N | <i>Jumlah populasi keseluruhan</i> |
| n | <i>Jumlah sampel yang dibutuhkan pada keseluruhan populasi</i> |
| \tanh | <i>Hyperbolic tangen</i> |
| FC | <i>Fully Connected</i> |
| TP | <i>True Positive</i> |
| FP | <i>False Positive</i> |
| TN | <i>True Negative</i> |
| FN | <i>False Negative</i> |



BAB I PENDAHULUAN

1.1 Latar Belakang

Dengan hadirnya era digitalisasi dalam kehidupan manusia, berbagai kemudahan telah muncul, salah satunya yaitu kemudahan mencari suatu informasi. Salah satu bentuk informasi yang sering diterima oleh manusia yaitu berita. Menurut KBBI, berita adalah cerita atau karangan mengenai kejadian atau peristiwa yang hangat. Berdasarkan survei yang dilakukan oleh APJII (Asosiasi Penyelenggara Jasa Internet Indonesia), jumlah pengguna internet di Indonesia hingga awal tahun 2024 mencapai 221 juta pengguna (Asosiasi Penyelenggara Jasa Internet Indonesia, 2024), sehingga dengan banyaknya pengguna internet ini membuat penyebaran berita semakin cepat. Media digital (*website*, media sosial, blog, dll.) selalu digunakan untuk menyebarkan berita dalam waktu yang singkat. Namun, berita yang tersebar di media digital terkadang tidak diseleksi dan diambil dari berbagai macam sumber sehingga ada kemungkinan kita bisa menerima informasi palsu (Beauvais, 2022).

Berita palsu (*hoax*) merupakan suatu berita yang bertujuan untuk memutarbalikkan fakta dengan menggunakan informasi yang seolah-olah meyakinkan namun kebenarannya belum dapat dipastikan (Hamzah, 2020). Berita *hoax* ini diklasifikasikan menjadi 7 jenis menurut (Wardle, 2019), yaitu *Satire or Parody*, *False Connection*, *Misleading Content*, *False Context*, *Imposter Content*, *Manipulated Content*, dan *Fabricated Content*. Klasifikasi ini juga yang digunakan oleh organisasi MAFINDO (Masyarakat Antifitnah Indonesia) untuk mengklasifikasikan berita *hoax* yang ada sesuai dengan jenisnya.



Gambar 1. Temuan isu hoaks KOMINFO periode Agustus 2018 – 31 Desember 2023

Sumber: (KOMINFO, 2024)



temuan pada **Gambar 1**, KOMINFO menemukan 12.547 hoaks yang tersebar di media digital. Selain itu, terdapat 4 topik dengan jumlah hoaks terbanyak, diantaranya kesehatan (2.357 hoaks), penipuan (2.210 hoaks), dan politik (1.628 hoaks) (KOMINFO, 2024). Dengan demikian, masyarakat Indonesia harus lebih berhati-hati lagi ketika

membaca suatu berita di berbagai media digital yang ada. Hal ini dikarenakan efek yang ditimbulkan dari penyebaran berita *hoax* yaitu dapat menggiring opini dan membuat masyarakat bisa mengambil keputusan yang salah untuk mempercayai suatu informasi yang belum terbukti kebenarannya (Hamzah, 2020).

Penelitian untuk mengukur tingkat literasi negara di dunia pernah dilakukan oleh *Central Connecticut University* pada tahun 2016. Hasilnya menunjukkan bahwa dari 61 negara yang berpartisipasi, Indonesia menempati peringkat 60 (Nayoga et al., 2021). Selain itu, penilaian juga dilakukan oleh *Organization for Economic Cooperation and Development* (OECD) dalam *Programme for International Student Assessment* (PISA) 2022. Salah satu penilaian yang dilakukan yaitu menilai literasi membaca pelajar di beberapa negara. Penilaian dilakukan terhadap pelajar yang berusia 15 tahun dari 81 negara. Hasilnya menunjukkan bahwa Indonesia menempati peringkat 71 dari 81 negara yang berpartisipasi dengan skor 359 (OECD, 2023). Kedua hal ini menunjukkan bahwa literasi di Indonesia masih kurang jika dibandingkan dengan beberapa negara lainnya. Dampaknya akan mengarah ke buruknya kemampuan untuk dapat mengevaluasi informasi yang diterima dengan kritis dan membedakan berita yang termasuk dalam kategori *hoax* atau tidak (Nayoga et al., 2021).

Cara yang bisa dilakukan untuk mengetahui suatu berita itu *hoax* atau bukan yaitu dengan melakukan validasi terhadap berita yang telah dibaca dan melakukan pendekatan linguistik dari teks tersebut (ejaan, tata bahasa, dan gaya penulisan) karena biasanya kesalahan penggunaan bahasa terjadi pada teks berita *hoax* (Conroy et al., 2015). Selain itu, teknik berbasis kecerdasan buatan juga bisa digunakan untuk memudahkan kita dalam mengetahui suatu berita itu *hoax* atau tidak (Aimeur et al., 2023). NLP (*Natural Language Processing*) merupakan cabang dari kecerdasan buatan yang memungkinkan komputer untuk bisa memahami, menghasilkan, dan memanipulasi bahasa manusia.

Terkait dengan penerapan NLP, khususnya dalam klasifikasi berita, terdapat penelitian dari (Biantong, 2019) yang melakukan penelitian terkait dengan deteksi *clickbait* pada artikel berita berbahasa Indonesia dengan jumlah data yang digunakan sebanyak 500 dan penggunaan beberapa metode (TF-IDF, *WordNet*, dan *Cosine Similarity*). Hasil penelitiannya menunjukkan bahwa dengan menggunakan metode tersebut untuk mendeteksi suatu berita itu *clickbait* atau tidak, nilai *Accuracy* yang diperoleh mencapai 78%.

Selain metode tersebut, BERT merupakan model yang sering digunakan untuk tugas-tugas NLP dan memiliki arsitektur *transformer* yang terdiri dari struktur *encoder* saja. Berdasarkan penelitian yang telah dilakukan oleh (Choudhary & Arora,



klasifikasi berita palsu dengan beberapa dataset yang berbeda *ws dataset*, *Twitter Fake News dataset*, *Kaggle-BS Detector*, *oCop*), hasilnya menyatakan bahwa nilai metrik evaluasi yang BERT lebih baik pada 4 dataset dibandingkan dengan model LSTM, *Attention Based Bi-LSTM*). Nilai *Training Accuracy* yang BERT berkisar antara 0.86-0.91 dan nilai *Validation Accuracy*-nya 0.86-0.90 (nilainya bergantung pada dataset yang digunakan).

Sedangkan untuk *Kaggle-BS Detector dataset*, nilai *Training* (0.97) dan *Validation Accuracy* (0.98) tertinggi didapatkan oleh model *Attention Based Bi-LSTM* (Choudhary & Arora, 2024).

Dalam perkembangannya, BERT telah dikembangkan untuk bisa memahami dan menghasilkan representasi bahasa Indonesia yang baik untuk tugas-tugas NLP. IndoBERT merupakan model monolingual dalam bahasa Indonesia yang mengikuti konfigurasi BERT (Rahmawati et al., 2022). Pada penelitian yang telah dilakukan oleh (Rahmawati et al., 2022) dalam tugas klasifikasi berita *hoax* berbahasa Indonesia, hasilnya menunjukkan bahwa model IndoBERT memiliki nilai metrik evaluasi yang lebih tinggi (*Accuracy*, *Precision*, *Recall*, dan *F1-Score*) dengan nilai yang sama (90%) dibandingkan dengan model lainnya (SVM & *Naïve Bayes*).

Selain BERT dan IndoBERT, LSTM merupakan jenis algoritma *deep learning* yang cukup sering digunakan untuk tugas yang berkaitan dengan NLP. LSTM merupakan algoritma lebih khusus dari *Recurrent Neural Network* (RNN) yang mampu mempelajari suatu dependensi bersifat jangka panjang dan mempertahankan informasi dalam jangka waktu yang lama. Arsitektur LSTM terdiri dari 3 bagian, yaitu *input gate*, *forget gate*, dan *output gate*. Berdasarkan penelitian yang telah dilakukan oleh (Gupta et al., 2021) dalam mendeteksi dan mengklasifikasikan *toxic comments* pada Wikipedia, hasilnya membuktikan bahwa dengan menggunakan LSTM (98.05%) dan Bi-LSTM (98.07%), nilai *Validation Accuracy* yang dihasilkan cenderung tinggi.

Selain itu, terdapat beberapa algoritma yang bisa dikombinasikan untuk mendapatkan hasil yang lebih optimal lagi, seperti menggabungkan IndoBERT dengan LSTM. Hal ini pernah dilakukan oleh (Putra et al., 2022), dimana peneliti membuat model klasifikasi untuk *tweet* berbahasa Indonesia menggunakan kombinasi IndoBERT – LSTM. Penelitian ini menyimpulkan bahwa model IndoBERT-LSTM berhasil mendapatkan *F1-Score* sebesar 98,90% pada dataset yang tidak termodifikasi (peningkatan 0,70% dari model *Word2Vec-LSTM* dan 0,40% dari model *fine-tuned* IndoBERT) dan 92,83% pada dataset yang telah termodifikasi (peningkatan 4,51% dari model *Word2Vec-LSTM* dan 0,69% dari model *fine-tuned* IndoBERT).

Berdasarkan urgensi dan penelitian yang telah dilakukan sebelumnya, penulis melakukan penelitian dengan judul **“Penerapan Model IndoBERT-LSTM untuk Klasifikasi Berita Hoax Berbahasa Indonesia”** dengan mempertimbangkan bahwa model LSTM memiliki performa yang baik dalam melakukan tugas klasifikasi teks dan algoritma ini dapat digabung dengan IndoBERT berdasarkan penelitian yang telah dilakukan sebelumnya untuk pemahaman konteks yang lebih baik pada klasifikasi berita *hoax* berbahasa Indonesia.



1.2 Landasan Teori

1.2.1 Berita Hoax

Berita palsu (*hoax*) adalah berita yang bertujuan untuk memutarbalikkan fakta dengan merekayasa informasi yang seolah-olah meyakinkan. Namun, karena hasil rekayasa telah menutupi sebagian atau seluruh informasi yang sebenarnya, kebenarannya tidak dapat dipastikan (Hamzah, 2020). Penyebaran berita ini dapat membahayakan masyarakat dan mengakibatkan ketegangan sosial, kerusakan hubungan interpersonal, dan menjadi ancaman bagi stabilitas demokrasi. Selain itu, informasi yang terkait dengan kesehatan yang tidak akurat, seperti informasi yang salah tentang vaksinasi, pengobatan alternatif/metode pengobatan yang belum terbukti efektif, atau berita palsu tentang penyakit/pandemi, nantinya dapat membahayakan nyawa dan kesejahteraan masyarakat (M & Hairunnisa, 2023). Menurut (Wardle, 2019), berita *hoax* dapat diklasifikasikan menjadi 7 jenis, yaitu:

1. *Satire or Parody*

Informasi jenis ini banyak mengandung ironi atau humor sebagai bentuk seni terhadap suatu keadaan atau orang yang berpotensi membodohi pembaca karena informasi tersebar dari satu tempat ke tempat lainnya sehingga dapat mengaburkan makna yang awalnya hanya ingin menyindir atau menertawakan suatu hal menjadi hal yang dianggap serius bagi pembaca (Zannettou et al., 2019).

2. *False Connection*

Informasi ini sering disebut dengan "*clickbait*", dimana judul berita dibuat dengan bahasa yang bersifat sensasional sehingga mendorong pembaca untuk membuka informasi tersebut, namun bentuk visual atau teks yang terdapat dalamnya tidak mendukung konten pada judul berita. Kebutuhan terhadap lalu lintas dan jumlah *click* pada industri berita menandakan bahwa ini tidak akan pernah hilang, tetapi akan ada dampak jangka panjang yang ditimbulkan antara pembaca dan situs berita yang dikunjungi terkait dengan kualitas berita (Wardle, 2019).

3. *Misleading Content*

Jenis ini dapat dikatakan sebagai informasi yang menyesatkan, dikarenakan dibuat dengan cara membingkai ulang cerita dalam bentuk berita (teks dan visual), menggunakan kutipan dari sumber lain untuk mendukung poin tertentu, mengutip statistik yang sesuai dan lain sebagainya. Tentunya tindakan ini akan membuat pembaca dapat menerima dan mempercayai informasi yang sesat (Wardle, 2019).



si ini biasanya mendeskripsikan konten yang asli, tetapi isi telah membingkai ulang hal tersebut dengan konteks ah (Wardle, 2019).

Kategori ini merupakan informasi yang seringkali meniru sumber asli dengan menggunakan logo berita terkenal, berita dari jurnalis ternama, atau akun yang mengatasnamakan tokoh terkenal untuk menyebarkan berita palsu yang dapat mengelabui/menipu pembaca (Wardle, 2019).

6. *Manipulated Content*

Informasi ini memanipulasi aspek dari konten asli ke dalam bentuk suatu konten, biasanya gambar atau video untuk mengecoh pembaca (Wardle, 2019).

7. *Fabricated Content*

Bentuk informasi jenis ini adalah konten yang direkayasa secara penuh oleh pembuat berita, sehingga isi informasi dalam konten tersebut dipastikan 100% salah (Wardle, 2019).

1.2.2 *Natural Language Processing (NLP)*

Natural Language Processing (NLP) adalah pendekatan interdisipliner terhadap *Machine Learning*, yang menggabungkan linguistik, kecerdasan buatan, dan ilmu komputer untuk mengajarkan bahasa alami kepada mesin. Pendekatan ini bertujuan agar komputer memiliki kemampuan dalam memahami, memproses, mensimulasikan, menghasilkan bahasa manusia, dan melakukan percakapan alami dengan manusia. *Machine Translation, Question and Answer Systems, Sentiment Analysis, Text Classification* merupakan beberapa contoh aplikasi yang dapat diwujudkan melalui teknik NLP. (Liu et al., 2024).

1.2.3 *Text Classification*

Salah satu jenis metode NLP yang sering dilakukan yaitu *Text Classification*. Dalam proses NLP, *Text Classification* bertujuan untuk memberikan label atau kategori tertentu ke data tekstual. Ini dapat dilakukan dalam berbagai konteks, seperti penyaringan berita, analisis opini, deteksi *spam*, pengorganisasian dokumen, perbaikan teks, dan penemuan topik. Metode ini bertanggung jawab untuk memberikan label, kategori, atau *tag* ke berbagai bagian teks, seperti pertanyaan, kalimat, dokumen, dan paragraf. Beberapa pendekatan *Machine Learning* dapat dilakukan untuk tugas *Text Classification*, seperti *Supervised Learning, Unsupervised Learning, dan Deep Learning (BERT, CNN, LSTM, dll.)* (Jamshidi et al., 2024).

1.2.4 **MAFINDO (Masyarakat Antifitnah Indonesia)**

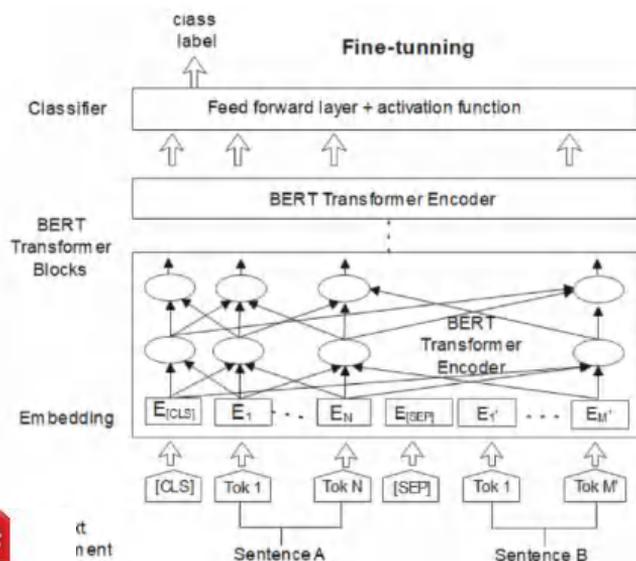


(Masyarakat Antifitnah Indonesia) merupakan komunitas anti-*hoax* pada tanggal 19 November 2016 yang terdiri lebih dari 95.000 anggota dari 1.000 relawan, dan lebih dari 20 cabang di seluruh Indonesia untuk mensosialisasikan bahaya berita *hoax*, melawan *hoax*, dan menciptakan imunitas terhadap *hoax* dikalangan masyarakat. Organisasi ini telah tersertifikasi secara internasional oleh

International Fact Checking Network (IFCN). MAFINDO bekerja dengan mengumpulkan seluruh berita *hoax* yang bersumber dari berbagai media digital yang ada dan mengklasifikasikan setiap berita yang didapat menjadi 7 jenis menurut (Wardle, 2019). Data yang dikumpulkan bisa berbentuk teks, gambar, dan video. Untuk narasi selain teks, biasanya tulisan yang terdapat pada gambar atau suara dalam video diubah menjadi bentuk teks narasi biasa. Selain itu, organisasi ini memiliki prioritas dalam pemeriksaan berita *hoax* dengan 2 kriteria, yaitu berbahaya - perlu dan tidak berbahaya - tidak perlu. Untuk kriteria yang tidak berbahaya - tidak perlu, jenis beritanya seperti *satire* dan lain-lain, maka akan disanggah jika dirasa perlu saja dan jika ada *resource* yang bisa melakukannya (Mafindo, 2024).

1.2.5 IndoBERT (*Indonesia Bidirectional Encoder Representations from Transformers*)

Model IndoBERT hadir untuk mengatasi masalah kurangnya kumpulan beranotasi, kelangkaan sumber daya bahasa, dan kurangnya standarisasi sumber daya untuk model NLP yang dapat menangani bahasa Indonesia saja. Model ini adalah suatu *pre-trained* model monolingual dalam bahasa Indonesia yang mengikuti konfigurasi dari BERT-Base sehingga struktur dari IndoBERT sama dengan BERT. IndoBERT dilatih dengan menggunakan lebih dari 220 juta kata dari tiga sumber utama, yaitu Wikipedia bahasa Indonesia (74 juta kata), artikel berita dari Kompas, Tempo, dan Liputan6 (55 juta kata), dan Korpus Web Indonesia (90 juta kata) (Koto et al., 2020).



Gambar 2. Bentuk model BERT
Sumber: (Choudhary & Arora, 2024)



irectional Encoder Representations from Transformers) yang memiliki arsitektur *transformer* yang hanya terdiri dari

struktur *encoder* saja. Model ini dinyatakan sebagai *pre-trained language representation* yang bisa menangkap makna semantik dan makna kontekstual penuh *input sequence* dari dua arah sekaligus selama proses pemodelan berlangsung. Model BERT memiliki beberapa varian, salah satunya yaitu BERT-Base yang terdiri dari 12 lapisan *encoder blocks*, 768 *hidden layer size*, *multi-head attention heads* yang berjumlah 12, dan jumlah total parameter sebesar 110 juta (Choudhary & Arora, 2024).

BERT mengharapkan representasi *input sequence* yang dibangun dengan menggabungkan kombinasi dari tiga *embedding*, yaitu *token embedding* (memberikan token tambahan untuk menandakan awal dan akhir kalimat), *segment embedding* (mempelajari karakteristik kalimat pertama dan kedua untuk membantu model membedakan keduanya), dan *positional embedding* (mempelajari informasi urutan dalam teks) (Sabharwal & Agrawal, 2021).

Terdapat dua strategi yang dimiliki BERT untuk mempelajari informasi kontekstual pada data, yaitu MLM (*Masked Language Modeling*) dan NSP (*Next Sentence Prediction*). Beberapa kata dalam teks ditutupi secara acak dalam tugas MLM. Kata [MASK] diprediksi dengan kata-kata konteks yang mengelilingi token [MASK]. Ketika urutan kata dimasukkan ke dalam BERT, 15% kata dari setiap urutan diganti dengan token [MASK] dan dipilih secara acak. Dari jumlah ini, 80% ditutupi, 10% diganti dengan kata acak, dan 10% dipertahankan. Ini dilakukan karena jika 100% merupakan kata bertopeng, maka model tidak selalu menghasilkan representasi token yang baik untuk kata yang tidak bertopeng. Dengan demikian, kinerja model akan lebih baik karena tidak terlalu berfokus pada posisi atau token tertentu (Sabharwal & Agrawal, 2021).

NSP lebih ke memprediksi kata berikutnya dalam suatu kalimat. Untuk memastikan bahwa model dapat memahami bagaimana kalimat yang berbeda dalam korpus teks saling terkait, BERT juga dilatih untuk tugas NSP dan menggunakan pasangan kalimat sebagai *input* dalam pemodelan. Untuk melakukan ini, 50% *input*-an diambil sedemikian rupa sehingga kalimat kedua dalam pasangan tersebut adalah kalimat berikutnya dalam dokumen asli dan 50% lainnya terdiri dari pasangan di mana kalimat kedua dipilih secara acak dari dokumen asli (Sabharwal & Agrawal, 2021).

Dalam BERT terdapat token tambahan yang diperlukan selama pemodelan berlangsung, yaitu token [CLS] yang ditambahkan pada awal kalimat untuk bisa mempelajari konteks pada keseluruhan kalimat & digunakan selama proses *Next Sentence Prediction* untuk memprediksi kalimat berikutnya berdasarkan kalimat sebelumnya dan token [SEP] sebagai pembatas antar kalimat untuk menekankan sebut (Sharma et al., 2022). Token akan melalui lapisan *self*-*attention* ke *feed-forward neural network* sebelum sampai ke ada setiap blok yang ditumpuk (Choudhary & Arora, 2024).

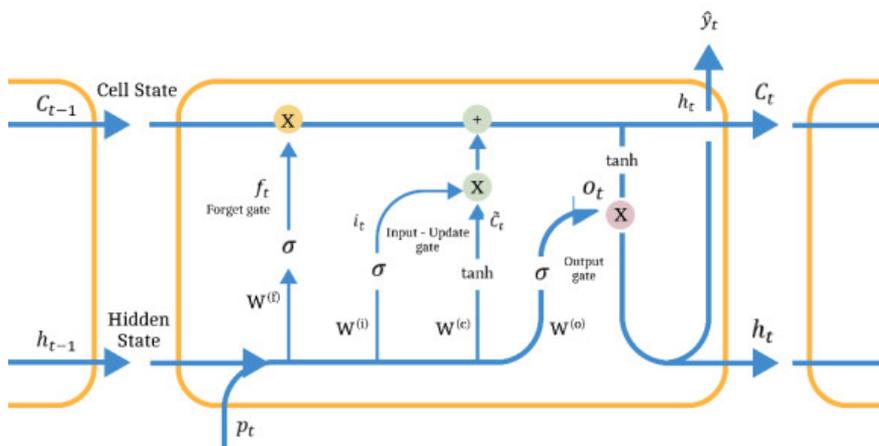
n *encoder* mengambil daftar *token embedding* dan *attention Output* dari proses ini yaitu berupa jumlah *embedding* dan ma. Satu vektor yang menggambarkan seluruh kalimat *input* e pengklasifikasi dan *hidden state* pada token pertama [CLS]



dari *output* model digunakan untuk menggambarkan seluruh kalimat yang akan digunakan untuk klasifikasi (Rai et al., 2022).

1.2.6 LSTM (Long-Short Term Memory)

LSTM (*Long-Short Term Memory*) merupakan salah satu model *deep learning* yang lebih khusus dari RNN (*Reccurent Neural Network*) yang mampu mempelajari suatu dependensi bersifat jangka panjang dan mempertahankan informasi dalam jangka waktu yang lama (Gupta et al., 2021). Model ini dirancang untuk menyelesaikan masalah *vanishing gradien* pada RNN yang berpengaruh pada kemampuan untuk bisa mempelajari dan memahami urutan maupun konteks yang panjang (Rai et al., 2022).



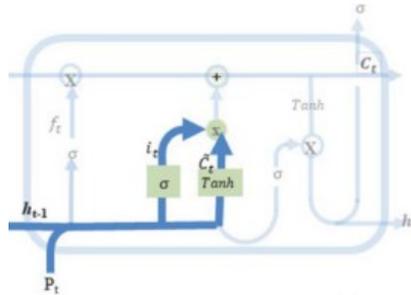
Gambar 3. Struktur yang terdapat dalam *cell* pada model LSTM
Sumber: (Okut, 2021)

LSTM terdiri dari modul berulang bernama *cell* yang didalamnya terdapat 4 *neural network* yang terhubung dengan cara khusus. Setiap *cell* akan meneruskan dua status ke *cell* berikutnya, yaitu *cell state* (C_t) yang berguna untuk menyimpan dan mengelola informasi penting dari satu *time step* ke *time step* lainnya dan *hidden state* (h_t) yang berfungsi untuk menyimpan informasi relevan yang tersimpan pada *time step* tertentu (Okut, 2021).

Dalam *cell* terdapat beberapa *gate* untuk membantu dalam mengetahui dependensi panjang dalam suatu urutan. Berikut ini adalah jenis-jenis *gate* yang terdapat dalam LSTM:



tugas untuk mengontrol informasi baru yang akan ditambahkan *current input*. *Gate* ini juga menjaga agar isi memori tidak *out* yang tidak penting. Terdapat fungsi aktivasi *sigmoid* dalam *gate* ini untuk menghasilkan nilai *input* dan mengubah informasi (Okut, 2021).



Gambar 4. Alur *input gate* dengan dua bagian (Lapisan *sigmoid* dan Lapisan *tanh*)

Sumber: (Okut, 2021)

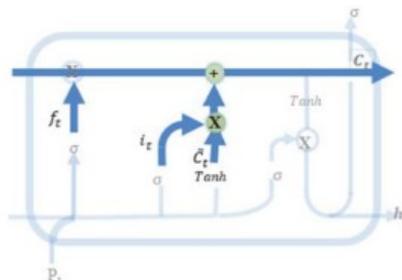
Secara matematis, persamaan untuk *input gate* (1) yaitu:

$$i_t = \sigma(W^{(i)}(p_t, h_{t-1}) + b^{(i)}) = \frac{1}{1 + e^{-(W^{(i)}(p_t, h_{t-1}) + b^{(i)})}} \quad (1)$$

dimana, $W^{(i)}$ dan $b^{(i)}$ merupakan bobot matriks dan vektor bias, p_t adalah indeks *current input timestep* dengan *timestep* sebelumnya h_{t-1} . Parameter *input gate* akan dipelajari dari *input training data*. Selain itu, *candidate cell state* dapat dihitung pada setiap *timestep* dengan informasi baru (p_t) (Okut, 2021). Selanjutnya, vektor nilai *candidate* baru (\tilde{C}_t) dibuat dengan menggunakan fungsi aktivasi *hyperbolic tanh* dengan rentang nilai (1,1). Persamaan untuk vektor ini (2) adalah sebagai berikut:

$$\tilde{C}_t = \tanh(W^{(c)}(p_t, h_{t-1}) + b^{(c)}) = \frac{e^{(W^{(c)}(p_t, h_{t-1}) + b^{(c)})} - e^{-(W^{(c)}(p_t, h_{t-1}) + b^{(c)})}}{e^{(W^{(c)}(p_t, h_{t-1}) + b^{(c)})} + e^{-(W^{(c)}(p_t, h_{t-1}) + b^{(c)})}} \quad (2)$$

Pada langkah berikutnya, nilai *input state* dan *cell candidate* digabungkan untuk membuat dan memperbarui *cell state*. Untuk mengubah *previous cell state* (C_{t-1}) menjadi *current cell state* (C_t), kombinasi linear antara *input gate* dan *forget gate* digunakan. *Input gate* (i_t) mengontrol jumlah data baru yang harus diperhitungkan melalui *candidate* (\tilde{C}_t), sementara *forget gate* (f_t) mengontrol jumlah *old memory cell content* (C_{t-1}) yang harus disimpan (Okut, 2021).



memory update* dengan *old memory* melalui *forget gate* dan *new memory* melalui *input gate

Sumber: (Okut, 2021)

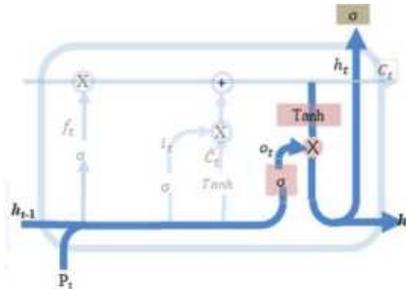


Dengan menggunakan perkalian *pointwise* ($\odot = \text{Hadamard Product}$), persamaan terbaru untuk *cell state* (3) yaitu:

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \quad (3)$$

2. Output Gate

Nilai *hidden state* (h_t) berikutnya ditentukan oleh *output gate* (o_t). Dengan kata lain, *output gate* mengontrol informasi mana yang akan diungkapkan dari *cell state* yang diperbarui (C_t) ke *output* dalam satu *timestep*. *Hidden state* mengandung data dari *input* sebelumnya. Selain itu, *calculated value* dalam *hidden state* pada *timestep* tertentu digunakan untuk prediksi ($\hat{y}_t = \text{softmax}(\cdot)$) (6) (Okut, 2021).



Gambar 6. Alur output informasi dengan sigmoid σ dan \tanh pada output gate

Sumber: (Okut, 2021)

Berikut ini adalah beberapa persamaan yang digunakan dalam *output gate* untuk menghasilkan prediksi dan *hidden state* baru:

$$o_t = \sigma(W^{(o)}(h_{t-1}, p_t) + b^{(o)}) = \frac{1}{1 + e^{-(W^{(o)}(h_{t-1}, p_t) + b^{(o)})}} \quad (4)$$

$$h_t = o_t \odot \tanh(C_t) = o_t \frac{e^{(f_t \odot C_{t-1} + i_t \odot \tilde{C}_t)} - e^{-(f_t \odot C_{t-1} + i_t \odot \tilde{C}_t)}}{e^{(f_t \odot C_{t-1} + i_t \odot \tilde{C}_t)} + e^{-(f_t \odot C_{t-1} + i_t \odot \tilde{C}_t)}} \quad (5)$$

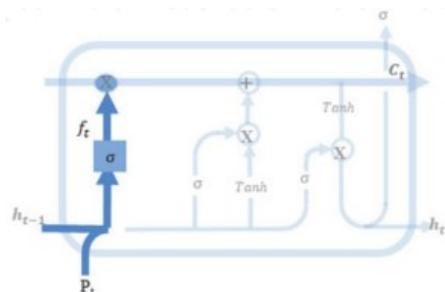
$$\hat{y}_t = \sigma(W^{(y)}h_t + b^{(y)}) = \frac{1}{1 - e^{-(W^{(y)}h_t + b^{(y)})}} \quad (6)$$

Pertama, fungsi *sigmoid* menerima *current input* dari *hidden state* sebelumnya (h_{t-1}). Kemudian, fungsi *tanh* memperbarui *cell state* yang baru (4). Terakhir, *output tanh* dikalikan dengan *output sigmoid* untuk mengetahui informasi apa yang harus dibawa oleh *hidden state* (5). Pembaruan *hidden state* adalah hasil akhir dari *output gate*, yang digunakan untuk memprediksi *time step* t. Oleh karena itu, tujuan *gate* ini adalah untuk membedakan *cell state* yang diperbarui (perbarui) dari *hidden state*. *Cell state* yang diperbarui (C_t) yakni informasi yang tidak perlu disimpan dalam *hidden state*. Namun, karena semua gerbang blok LSTM menggunakan σ dan \tanh diperbarui setiap saat, informasi ini sangat penting. Dengan *gate* melakukan penilaian mengenai bagian apa saja dari *cell state* yang akan disimpan dalam *hidden state* (h_t). *Cell state* dan *hidden state* yang baru akan digunakan pada *time step* berikutnya (Okut, 2021).



3. Forget Gate

Forget gate (f_t) berfungsi menentukan informasi apa yang harus dibuang atau disimpan dalam *cell state*. Fungsi aktivasi *sigmoid* melakukan proses ini dengan menghasilkan nilai antara 0 dan 1 dari *weighted input* ($W_f p_t$), *hidden state* sebelumnya (h_{t-1}), dan bias (b_f) (Okut, 2021)



Gambar 7. Alur forget gate dalam penentuan informasi yang akan disimpan pada cell state

Sumber: (Okut, 2021)

Persamaan pada *forget gate* (7) diuraikan sebagai berikut:

$$f_t = \sigma(W^{(f)}(p_t, h_{t-1}) + b^{(f)}) = \frac{1}{1 - e^{-(W^{(f)}(p_t, h_{t-1}) + b^{(f)})}} \quad (7)$$

dimana, σ merupakan fungsi aktivasi *sigmoid*, $W^{(f)}$ dan $b^{(f)}$ adalah bobot matriks dan vektor bias. Fungsi ini mengambil *old output* (h_{t-1}) pada *time* $t - 1$ dan *current input* (p_t) pada *time* t untuk menghitung komponen yang mengontrol *cell state* dan *hidden state* dari lapisan tersebut. Hasilnya berupa angka biner [0, 1], dimana 0 mewakili informasi yang dibuang dan 1 mewakili informasi yang harus tetap disimpan (Okut, 2021).

1.2.7 Early Stopping

Early stopping merupakan salah satu metode yang dapat digunakan untuk mencegah *overfitting* dengan memantau kinerja model pada *validation set* selama proses pemodelan berlangsung. Dalam *early stopping*, terdapat satu parameter (*patience*) yang menentukan jumlah iterasi tertentu untuk bisa menghentikan seluruh proses yang ada jika *validation set* tidak mengalami peningkatan (Hussein & Shareef, 2024).

1.2.8 Batch Normalization dan Dropout



metode modifikasi yang berguna untuk *neural network* karena *zation error* beriringan dengan menyeimbangkan *training error*. Untuk mengurangi *overfitting* selama proses *training* berlangsung 3). Contoh dari metode regulasi ini yaitu *Batch Normalization*

Batch Normalization. Metode ini bertugas dalam menormalisasi dan memusatkan *feature map* untuk meningkatkan kinerja pembelajaran dan inferensi. Ini dilakukan dengan menghitung rata-rata dan standar deviasi dalam *mini-batch*, lalu menggunakan parameter yang dapat dipelajari untuk menyesuaikan *feature map*. Tujuan utama dari *Batch Normalization* yaitu mengurangi masalah terkait dengan pergeseran kovariat internal, yang mengacu pada perubahan distribusi *network activation* akibat pembaruan bobot selama pelatihan (Balestrieri & Baraniuk, 2022).

Dropout. *Dropout* sebagai salah satu contoh regularisasi bekerja dengan menonaktifkan subset neuron secara acak di setiap *training iteration*. Hal ini akan mendorong jaringan untuk mempelajari fitur yang lebih tangguh dan umum karena tidak dapat lagi bergantung pada neuron tunggal manapun. Proses ini memaksa jaringan untuk membuat representasi data yang lebih terdistribusi. Pada akhirnya, metode *dropout* meningkatkan kinerja dan kekokohan model sehingga dapat digeneralisasi ke data yang tidak terlihat (Salehin & Kang, 2023).

1.2.9 API (Application Programming Interface)

API (*Application Programming Interface*) merupakan serangkaian aturan dan protokol yang memungkinkan aplikasi perangkat lunak untuk mengakses dan mengambil data dari sistem atau layanan lain. API yang dirancang untuk pengambilan data biasanya memungkinkan pengembang untuk bisa mengakses dan memanfaatkan data dari sumber eksternal dalam aplikasi mereka. Interaksi ini biasanya melibatkan pengiriman permintaan ke API dengan parameter tertentu, yang kemudian mengembalikan data yang diminta dalam format terstruktur, seperti JSON atau XML. Dengan API, pengembang dapat meningkatkan pengalaman pengguna saat menggunakan aplikasi web atau seluler (Russell et al., 2019).

1.2.10 Scraping dengan BeautifulSoup

Teknik *scraping* merupakan metode ekstraksi data secara otomatis pada situs web secara *real-time* dan menghasilkan data yang lebih menyeluruh, akurat, dan konsisten daripada ekstraksi data secara manual (Khder, 2021). *BeautifulSoup* adalah salah satu alat yang paling sering digunakan untuk teknik ini karena merupakan pustaka *Python* yang mampu untuk mengurai dan mengekstrak data dari berkas HTML dan XML. Pustaka ini membuat *parse tree* dari sumber halaman, yang kemudian dapat digunakan untuk menavigasi, mencari, dan mengubah data yang diuraikan. Pustaka ini sangat membantu dalam menemukan elemen tertentu dalam *source code* halaman selama proses *scraping* berlangsung (*The Bucharest Economic Studies, Romania et al., 2024*).



beberapa fitur tambahan yang merupakan bagian penting dari *Machine Learning pipeline*, seperti *data preprocessing*, *data resampling*, *splitting data*, parameter evaluasi, *hyperparameter tuning* untuk mengoptimalkan kinerja algoritma, dan lain-lain (Bisong, 2019).

1.2.12 TF-IDF (*Term Frequency – Inverse Document Frequency*)

Term Frequency – Inverse Document Frequency (TF-IDF) merupakan metrik numerik yang berguna untuk menentukan penting atau tidaknya sebuah kata berdasarkan frekuensi kemunculan kata tersebut dalam suatu teks. Jika sebuah kata atau istilah sering muncul dalam sebuah teks, maka hasilnya akan signifikan sehingga harus diberikan skor tinggi. Namun, jika kata itu muncul dalam terlalu banyak teks lain, kemungkinan besar itu bukan pengenalan unik yang harus diberi skor yang lebih rendah (Gothankar et al., 2022). Cara untuk menghitung masing-masing TF (8) dan IDF (9) yaitu:

$$TF(w) = \frac{(\text{Jumlah kemunculan istilah } w \text{ dalam kalimat})}{(\text{Jumlah keseluruhan istilah dalam kalimat})} \quad (8)$$

$$IDF(w) = \log_{10} \frac{(\text{Jumlah keseluruhan kalimat})}{(\text{Jumlah kalimat yang mengandung kata } w)} \quad (9)$$

Dengan demikian, untuk menghitung TF-IDF (10), persamaannya adalah:

$$TF - IDF(w) = TF(w) \times IDF(w) \quad (10)$$

1.2.13 *Stratified Sampling*

Stratified Sampling membagi populasi menjadi beberapa kategori atau strata, dan dipilih elemen dari setiap kategori. Rumus *sampling* ini melibatkan pemilihan sampel dari setiap kategori secara proporsional dengan ukuran populasi kategori tersebut untuk memastikan bahwa sampel akhir mewakili setiap strata secara memadai (Ilyasu & Etikan, 2021). Berikut ini adalah persamaan dari teknik *sampling* ini:

$$n_i = \frac{N_i}{N} \times n \quad (11)$$

n_i merupakan jumlah sampel yang diambil dari strata i , N_i adalah jumlah populasi dalam strata i , N menandakan jumlah populasi keseluruhan, dan n berupa jumlah sampel yang diinginkan pada keseluruhan populasi (AIRifai, 2020).

1.2.14 *Hugging Face Transformers*

Pustaka *Hugging Face Transformers* adalah perangkat lengkap untuk NLP yang memungkinkan integrasi *Transfer Learning* dengan model *Transformer* berskala besar. Tujuannya yaitu agar akses ke perangkat NLP tingkat lanjut menjadi lebih



Optimized using
trial version
www.balesio.com

terjangkau. Pustaka ini menyediakan *Unified API* yang menggabungkan berbagai model sebelumnya, sehingga pengguna bisa terlibat dalam tugas-tugas *classification*, *question answering*, dan lain sebagainya. Fitur utama pustaka ini selain *Unified API* yaitu penggunaan model terlatih & sumber daya, kompatibel dengan *PyTorch* & *TensorFlow* interoperabilitas mendalam antara dua pustaka tersebut, dan

memiliki komunitas aktif dari peneliti/praktisi & dokumentasi yang luas (Wolf et al., 2020).

1.2.15 PyTorch

PyTorch merupakan pustaka komputasi ilmiah berbasis *Python* yang menggantikan sebagian besar *low-level code* dari proyek *Torch* berbasis *Lua*. Pustaka *Deep Learning* ini memiliki API yang dapat digunakan dengan mudah oleh pengguna yang mau melakukan proses pelatihan model dengan berbagai macam jenis algoritma *Deep Learning*. Keunggulan yang dimiliki oleh *PyTorch* yaitu API-nya dapat digunakan dengan mudah, proses *debugging* mudah dilakukan, kode mudah dipahami, memiliki jumlah jenis lapisan yang sama dengan *Torch* (*Convolution*, *LSTM*, dan *GRU*), terdapat berbagai fungsi *Loss*, dianggap sebagai augmentasi *numpy* untuk GPU, dan lebih cepat daripada pustaka lain seperti *chainer* dan *dynet*. Selain itu, *PyTorch* memiliki tiga tingkat abstraksi, yaitu *Tensor* (*array* N-dimensi yang berjalan pada GPU), *Variable* (*node* dalam *computational graph* yang menyimpan data dan *graph*), dan *Module* (lapisan *neural network* yang akan menyimpan *state* atau *weights* yang dapat dipelajari) (Prakash & Kanagachidambaresan, 2021).

Dalam *PyTorch*, terdapat satu metode yang bernama *Data Loader*. Metode ini dapat mengulang set data dengan memanfaatkan *parallel processing*, *pre-fetching*, *batching*, dan teknik lainnya untuk mengurangi waktu *loading* dan *overhead* memori. Tujuan utama *Data Loader* yaitu memindahkan sampel data dari penyimpanan ke memori untuk proses *training* dan kemudian membentuk *batch* sampel yang akan dimasukkan ke dalam model (Ofeidis et al., 2022).

1.2.16 Preprocessing Data

Tahapan *preprocessing* merupakan suatu langkah penting untuk dilakukan pada data teks mentah agar data yang akan digunakan selama proses pelatihan model memiliki kualitas yang baik. Beberapa proses yang diterapkan dalam tahapan ini yaitu:

1. Case Folding

Case Folding berfungsi untuk mengubah semua ukuran huruf dalam teks menjadi huruf kecil (Pradana & Hayaty, 2019).

2. Punctuation Removal

Tanda baca memainkan peran penting dalam pemahaman teks dan semantik. Beberapa studi telah dilakukan oleh peneliti untuk mengetahui dampak penghilangan tanda baca pada model-model NLP (RoBERTa, BERT, dll.) dan ditemukan bahwa menghilangkan tanda baca dapat meningkatkan akurasi model kasus (Lamprou et al., 2023). Tanda baca merupakan karakter “.”, “&” dan lain sebagainya yang tidak diperlukan dalam tugas-na & Hayaty, 2019).



Number Removal

dalam bahasa Indonesia yaitu “yang”, “untuk”, “di”, “dari”, dan hapusnya kata-kata tersebut karena *stopword* merupakan kata

yang sering muncul dan umum digunakan namun tidak memberi pemaknaan yang berarti (Pradana & Hayaty, 2019). Selain itu, dengan dihapusnya *stopword* pada teks, akan mengurangi ukuran data teks yang terlihat berat (Rosid et al., 2020). Angka-angka yang terdapat dalam teks akan dihilangkan karena tidak memiliki makna yang berarti pada kasus tertentu dan dapat meningkatkan efisiensi ketika proses pemodelan berlangsung (Pradana & Hayaty, 2019)

4. *Stemming*

Stemming merupakan proses pemetaan dan penguraian kata yang berimbuhan menjadi bentuk kata dasarnya tanpa melihat konteks dari kata tersebut (Rosid et al., 2020).

5. *WordPiece Tokenizing*

WordPiece Tokenizing adalah algoritma tokenisasi yang bertujuan untuk membagi teks menjadi bagian yang lebih kecil (token), yang berupa kata atau subkata. Jika sebuah kata tidak dapat ditokenisasi sepenuhnya, itu akan dipetakan ke token tertentu yang disebut sebagai *unknown*. Pendekatan ini dapat memecah kata menjadi unit-unit subkata bermakna yang bisa mempertahankan makna linguistik (Song et al., 2021). Tokenisasi yang lebih cepat dan efektif dapat mengurangi latensi pemrosesan, yang meningkatkan kinerja BERT dalam berbagai tugas NLP. Secara keseluruhan, *WordPiece Tokenization* berguna secara optimal untuk mempersiapkan teks untuk model BERT dan meningkatkan kemampuannya dalam memahami konteks bahasa (Song et al., 2021).

1.2.17 Metrik Evaluasi

Terdapat beberapa evaluasi yang akan digunakan setelah pelatihan dan pengujian model, yaitu *Accuracy*, *Precision*, *Recall*, dan *F1-Score*. Metode evaluasi tersebut akan menggunakan *confusion matrix* yang berisi tabel informasi terkait prediksi hasil klasifikasi terhadap hasil aktual untuk menghitung kinerja model. Tabel ini terdiri dari empat atribut, yaitu *True Positive* (TP), *False Positive* (FP), *True Negative* (TN), dan *False Negative* (FN) (Jude Chukwura Obi, 2023).

Accuracy. Evaluasi ini akan menghitung jumlah klasifikasi yang benar pada keseluruhan set pengujian (Jude Chukwura Obi, 2023). Persamaan untuk menghitung *Accuracy* (12) yaitu:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (12)$$

Precision. *Precision* digunakan untuk menghitung *True Positive* dengan semua data yang diprediksi *Positive*, termasuk beberapa nilai *True Negative* yang diprediksi secara keliru oleh model sebagai *Positive* (*False Positive*) (Jude Chukwura Obi, 2023). Persamaan untuk menghitung *Precision* (13) yaitu:

(13)



Recall diketahui sebagai evaluasi untuk mengukur kemampuan mengidentifikasi nilai *True Positive* dengan benar terhadap yang aktualnya bernilai *Positive* (Jude Chukwura Obi, 2023). Persamaan untuk menghitung *Recall* (14) yaitu:

$$Recall = \frac{TP}{TP+FN} \quad (14)$$

F1-Score. *F1-Score* akan menghitung *harmonic mean* dari *Precision* dan *Recall*. Evaluasi ini melakukan perhitungan terhadap *False Positive* dan *False Negative*. Hasil evaluasi ini lebih sering digunakan jika melibatkan *cost* atau *class* yang tidak sama (Jude Chukwura Obi, 2023). Persamaan untuk menghitung *F1-Score* (15) yaitu:

$$F1 - Score = \frac{2 \times Precision \times Recall}{Precision+Recall} = \frac{2TP}{2TP+FP+FN} \quad (15)$$

1.3 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, penelitian ini merumuskan masalah bagaimana cara mengimplementasikan dan tingkat performa model IndoBERT-LSTM dalam mengklasifikasikan berita *hoax* dan *non-hoax*?

1.4 Tujuan dan Manfaat

1.4.1 Tujuan

Berdasarkan rumusan masalah yang ada, maka tujuan dari penelitian ini adalah untuk mengetahui cara mengimplementasikan dan tingkat performa model IndoBERT-LSTM dalam mengklasifikasikan berita *hoax* dan *non-hoax*.

1.4.2 Manfaat

Manfaat yang diharapkan dari penelitian ini yaitu peneliti dapat menjadikan hasil penelitian yang telah dilakukan sebagai referensi dalam mengembangkan model untuk tugas *text classification*, masyarakat dapat lebih mudah mengetahui suatu berita itu termasuk dalam kategori *hoax* atau bukan dengan bantuan model klasifikasi ini, dan meningkatkan kesadaran publik terkait pentingnya melakukan verifikasi informasi yang diterima sebelum menyebarkan atau mempercayai suatu informasi.

1.5 Ruang Lingkup

1. Berfokus pada berita berbasis teks dan berbahasa Indonesia
2. Berfokus pada 2 topik utama, yaitu Politik & Pemerintahan, Kesehatan
3. Kategori klasifikasi teks hanya terbagi menjadi dua kategori (*hoax* dan *non-hoax*)
4. *Dataset* berita *hoax* bersumber dari organisasi MAFINDO dan *dataset* berita *non-hoax* bersumber dari situs berita detik.com.
5. *Dataset* berita *hoax* berfokus pada 4 jenis klasifikasi, yaitu *Misleading Content*,



inipulated Content, *Fabricated Content*

tidak akan hanya berfokus pada isi teks berita, tidak menyertakan

akan algoritma IndoBERT-LSTM untuk proses pemodelan *oax*.