



# Implementasi Metode Long Short-Term Memory (LSTM) untuk Klasifikasi Berita Online Berdasarkan Konten Teks

Indar Kusmanto<sup>1</sup>, Musliadi KH<sup>2,\*</sup>, Hidayat<sup>3</sup>, Kristian<sup>3</sup>

<sup>1</sup>Fakultas Ilmu Komputer, Teknik Informatika, Universitas Tomakaka, Mamuju

Jl. Ir. H. Juanda No. 77/44, Kecamatan Tapalang, Kabupaten Mamuju, Sulawesi Barat, Indonesia

<sup>2</sup>Fakultas Komputer, Teknik Informatika, Universitas Universal, Batam

Kompleks Maha Vihara Duta Maitreya Bukit Beruntung Sungai Panas, Batam, Kota Batam, Kepulauan Riau, Indonesia

<sup>3</sup>Fakultas Ilmu Komputer, Sistem Informasi, Universitas Tomakaka, Mamuju

Jl. Ir. H. Juanda No. 77/44, Kecamatan Tapalang, Kabupaten Mamuju, Sulawesi Barat, Indonesia

Email: <sup>1</sup>indarkusmanto88@gmail.com, <sup>2,\*</sup>musliadikh@gmail.com, <sup>3</sup>hidayatfikom@gmail.com,

<sup>4</sup>kristiansisteminformasi09@gmail.com,

Email Penulis Korespondensi: musliadikh@gmail.com

Submitted: 30/10/2025; Accepted: 03/01/2026; Published: 05/01/2026

**Abstrak**—Penelitian ini bertujuan melakukan klasifikasi berita berbahasa Indonesia menggunakan metode Long Short-Term Memory (LSTM) dan mengevaluasi kinerjanya melalui metrik accuracy, precision, recall, dan F1-score. Dataset terdiri dari 48.634 judul berita yang dikumpulkan dari berbagai portal nasional dan regional, mencakup lima kategori utama: keuangan, perjalanan, kesehatan, makanan, dan olahraga. Proses penelitian meliputi tahapan pra-pemrosesan teks, seperti tokenisasi, stop-word removal, normalisasi, dan stemming, dilanjutkan dengan representasi fitur menggunakan word embedding serta perancangan arsitektur model LSTM. Kinerja model diuji menggunakan confusion matrix dan validasi tambahan melalui cross-validation untuk memastikan konsistensi hasil. Model LSTM menunjukkan performa yang baik, dengan akurasi 90%, precision 89%, recall 88%, dan F1-score 89%, yang menggambarkan kemampuan model dalam mengenali pola semantik dan menangkap ketergantungan konteks pada data teks secara efektif. Selain itu, LSTM terbukti lebih unggul dibandingkan metode baseline dengan peningkatan akurasi sebesar 6%, sehingga menguatkan posisinya sebagai metode yang andal untuk tugas klasifikasi teks berbahasa Indonesia. Secara keseluruhan, hasil penelitian ini menegaskan bahwa kombinasi teknik pra-pemrosesan yang optimal dan rancangan arsitektur LSTM yang tepat mampu meningkatkan performa sistem klasifikasi berita serta memberikan potensi penerapan pada berbagai kebutuhan analisis teks di era informasi digital.

**Kata Kunci:** Klasifikasi Teks; Teks Berbahasa Indonesia; Long Short-Term Memory; LSTM; NLP

**Abstract**—This study aims to classify Indonesian-language news using the Long Short-Term Memory (LSTM) method and to evaluate its performance through accuracy, precision, recall, and F1-score metrics. The dataset consists of 48,634 news titles collected from various national and regional portals, covering five main categories: finance, travel, health, food, and sports. The research process involves several text preprocessing stages—tokenization, stop-word removal, normalization, and stemming—followed by feature representation using word embedding and the design of the LSTM model architecture. The model's performance is assessed using a confusion matrix along with additional validation through cross-validation to ensure result consistency. The LSTM model demonstrates strong performance, achieving 90% accuracy, 89% precision, 88% recall, and 89% F1-score, indicating its capability to capture semantic patterns and contextual dependencies in textual data effectively. In addition, LSTM outperforms the baseline method with a 6% increase in accuracy, reinforcing its reliability for Indonesian text classification tasks. Overall, the findings confirm that the combination of optimal preprocessing techniques and a well-designed LSTM architecture enhances the performance of the news classification system and offers significant potential for various text analysis applications in the digital information era.

**Keywords:** Text Classification; Indonesian Text; Long Short-Term Memory; LSTM; NLP

## 1. PENDAHULUAN

Dalam era digital yang berkembang pesat, arus informasi semakin sulit dikendalikan karena jumlah berita yang dipublikasikan secara daring meningkat secara eksponensial setiap harinya [1], [2]. Hal ini menyebabkan munculnya tantangan baru dalam dunia teknologi informasi, khususnya dalam proses klasifikasi dan penyaringan berita yang relevan [3], [4]. Banyaknya berita yang tersebar di berbagai platform digital mengakibatkan pengguna kesulitan membedakan antara berita aktual dan tidak relevan, bahkan terkadang berita palsu [5]. Tantangan lain yang dihadapi adalah kompleksitas bahasa alami yang digunakan dalam berita, seperti ambiguitas makna, variasi kosakata, dan konteks yang luas. Dalam konteks penelitian ini, fokus permasalahan terletak pada klasifikasi topik berita, yaitu proses mengelompokkan berita ke dalam kategori tertentu seperti keuangan, perjalanan, kesehatan, makanan, dan olahraga. Metode konvensional seperti Naïve Bayes dan Support Vector Machine sering kali gagal menangkap ketergantungan jangka panjang dalam teks, sehingga performanya kurang optimal dalam tugas klasifikasi topik tersebut [6], [7], [8]. Oleh karena itu, dibutuhkan pendekatan yang lebih cerdas dan adaptif, yang tidak hanya dapat mengenali struktur sintaksis tetapi juga memahami konteks semantik dalam teks berita. Salah satu solusi yang menjanjikan adalah penggunaan metode Long Short-Term Memory (LSTM), yaitu varian dari jaringan saraf berulang (Recurrent Neural Network) yang mampu menyimpan dan memanfaatkan informasi dari urutan kata dalam teks untuk menghasilkan prediksi yang lebih akurat [9], [10]. Pendekatan ini diharapkan dapat mengatasi permasalahan utama dalam klasifikasi berita, yaitu kehilangan konteks jangka panjang dan ketidakmampuan mengenali pola semantik yang kompleks.

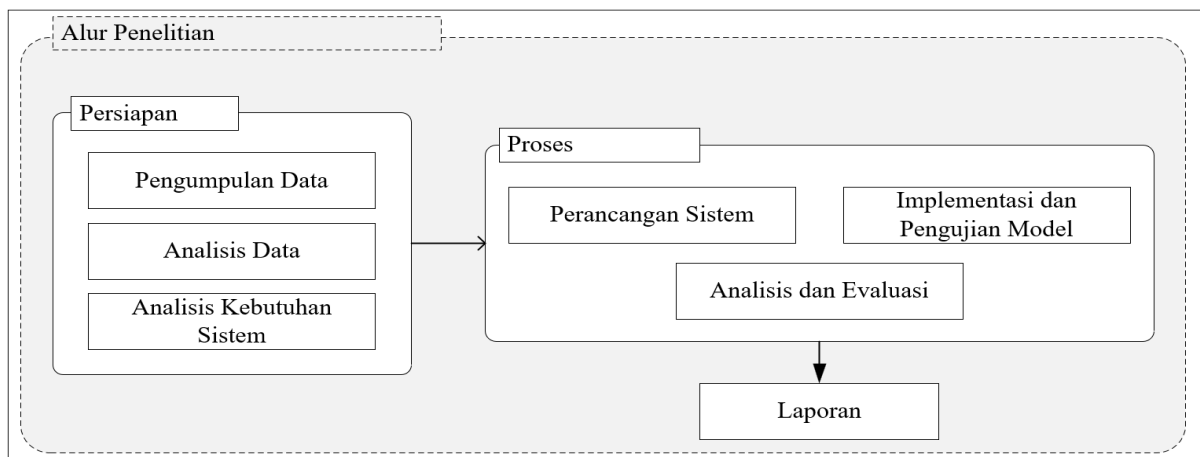
Sejumlah penelitian dalam lima tahun terakhir menunjukkan bahwa metode Long Short-Term Memory (LSTM) memiliki potensi besar dalam bidang klasifikasi teks, termasuk klasifikasi berita. Beberapa studi melaporkan bahwa model Bidirectional LSTM mampu meningkatkan akurasi dalam pengelompokan berita berbahasa Mandarin dengan memanfaatkan konteks urutan kata yang lebih mendalam [11]. Penelitian lain di *Frontiers in Artificial Intelligence* menunjukkan bahwa integrasi mekanisme Attention pada arsitektur Bi-LSTM secara signifikan meningkatkan nilai akurasi dan F1-score dibandingkan dengan LSTM standar dalam mendeteksi berita palsu (fake news detection) [12]. Studi berikutnya yang mengusulkan pendekatan multi-channel LSTM yang menggabungkan beberapa saluran representasi fitur secara paralel untuk mengatasi bias konteks dan memperbaiki performa klasifikasi berita palsu berbahasa Inggris [13]. Pada konteks nasional, penelitian terkini menunjukkan bahwa penerapan LSTM untuk klasifikasi berita lokal sangat dipengaruhi oleh tahapan preprocessing dan pemilihan optimizer yang tepat [10]. Selain itu, tinjauan literatur terbaru menyimpulkan bahwa meskipun model Transformer seperti BERT dan RoBERTa kini banyak digunakan, LSTM tetap relevan karena efisiensi komputasi dan kemampuannya beradaptasi terhadap dataset berskala terbatas [14].

Berdasarkan berbagai penelitian tersebut, terdapat beberapa kesenjangan penelitian (research gap) yang dapat diidentifikasi. Sebagian besar penelitian terdahulu lebih banyak difokuskan pada dataset berbahasa Inggris atau Mandarin, sementara penelitian yang menerapkan LSTM untuk klasifikasi berita berbahasa Indonesia masih sangat terbatas. Selain itu, banyak penelitian yang hanya berfokus pada satu jenis arsitektur LSTM tanpa mengeksplorasi variasi seperti Bidirectional LSTM atau Attention-based LSTM yang berpotensi memberikan hasil lebih optimal. Penelitian-penelitian sebelumnya juga umumnya tidak melakukan analisis mendalam terhadap kontribusi setiap tahap dalam pipeline pemrosesan teks, seperti tokenization, stopword removal, dan penggunaan word embedding. Di sisi lain, walaupun studi-studi terbaru telah menunjukkan keunggulan model Transformer, perbandingan langsung antara LSTM dengan model modern tersebut dalam konteks berita lokal masih jarang dilakukan. Oleh karena itu, diperlukan penelitian lanjutan yang tidak hanya menerapkan model LSTM, tetapi juga melakukan evaluasi sistematis terhadap performa, kompleksitas, dan generalisasi model tersebut pada berbagai jenis berita.

Berdasarkan analisis kesenjangan tersebut, penelitian ini bertujuan untuk mengimplementasikan dan mengevaluasi model klasifikasi berita berbasis LSTM pada dataset berita berbahasa Indonesia dengan mempertimbangkan berbagai konfigurasi arsitektur seperti Bidirectional LSTM dan Attention-based LSTM. Penelitian ini juga akan membandingkan hasilnya dengan metode klasik seperti Support Vector Machine serta model modern berbasis Transformer (BERT dan RoBERTa) untuk melihat sejauh mana LSTM masih relevan dalam konteks bahasa lokal. Selain itu, penelitian ini akan menganalisis pengaruh tahapan preprocessing, ukuran embedding, dan parameter jaringan terhadap kinerja model secara keseluruhan. Harapannya, penelitian ini dapat memberikan rekomendasi arsitektur dan strategi pelatihan yang efisien serta menjadi referensi empiris bagi peneliti dan praktisi dalam pengembangan sistem klasifikasi berita yang akurat, efisien, dan sesuai dengan karakteristik bahasa Indonesia masa kini.

## 2. METODOLOGI PENELITIAN

Penelitian ini menggunakan pendekatan kuantitatif eksperimental dengan fokus penerapan metode Long Short-Term Memory (LSTM) untuk klasifikasi teks berita berbahasa Indonesia [15]. Proses penelitian dirancang secara sistematis mulai dari tahap pengumpulan data hingga Analisis dan evaluasi. Setiap tahapan ditata agar sistem klasifikasi yang dikembangkan mampu memberikan hasil yang akurat dan dapat diandalkan dalam mengelompokkan berita ke dalam kategori relevan [16] [17]. Alur penelitian disusun ke dalam tiga tahap utama, yaitu tahap persiapan, tahap proses, dan tahap laporan. Struktur alur penelitian tersebut ditampilkan pada Gambar 1, sedangkan penjelasan rinci dari setiap tahapan diuraikan pada bagian berikutnya.



Gambar 1. Alur Penelitian



## 2.1 Tahap Persiapan

Tahap pertama adalah pengumpulan, analisis kebutuhan sistem, serta penyusunan dataset berita. Peneliti mengumpulkan dataset teks berita dari berbagai sumber daring dalam rentang waktu tertentu untuk memastikan keragaman topik dan gaya penulisan. Dataset yang digunakan dalam penelitian ini berjumlah 48.634 dokumen berita dan terbagi ke dalam 5 kategori utama, yaitu keuangan, perjalanan, kesehatan, makanan, dan olahraga. Distribusi data per kategori adalah sebagai berikut: keuangan (10.421), perjalanan (9.876), kesehatan (9.552), makanan (8.934), dan olahraga (9.851). Informasi ini penting untuk mengidentifikasi potensi ketidakseimbangan kelas (class imbalance) dalam proses pelatihan model.

Setelah data terkumpul lalu, dilakukan proses data cleaning dan text preprocessing seperti tokenization, stop-word removal, case folding, dan stemming [18]. Selanjutnya dilakukan pembagian data ke dalam set training, dan testing dengan proporsi 80 % training, dan 20% testing) [19]. Proses embedding juga dilakukan karena LSTM memerlukan input numerik berurutan untuk memproses konteks kata. Dalam penelitian ini digunakan embedding layer dari Keras yang diinisialisasi secara acak (random initialization) dan dilatih bersamaan dengan model (trainable), karena pendekatan ini terbukti efektif untuk dataset bahasa Indonesia yang tidak memiliki pretrained embedding yang stabil. Referensi teori menunjukkan bahwa LSTM sangat sesuai untuk urutan data karena kemampuannya menangani ketergantungan jangka panjang (vanishing gradient) dibanding RNN biasa [20].

## 2.2 Tahap Proses

Tahap proses dilakukan setelah tahap persiapan dilaksanakan. Pada tahap ini terbagi atas tiga proses utama, yakni: Perancangan sistem, implementasi dan pengujian model, serta analisis dan evaluasi. Dimana pada proses pertama dilakukan pembuatan arsitektur model LSTM. Model dirancang dengan lapisan embedding untuk representasi kata, lapisan LSTM untuk menangkap hubungan sekuensial antar kata dalam teks, dan lapisan dense dengan fungsi softmax sebagai keluaran klasifikasi. Secara spesifik, arsitektur model menggunakan 1 lapisan Embedding dengan dimensi vektor 128, kemudian 1 lapisan LSTM dengan 128 unit, serta lapisan Dense akhir dengan 3 neuron sesuai jumlah kelas dan fungsi aktivasi softmax. Panjang urutan input (max sequence length) ditetapkan sebesar 100 token berdasarkan distribusi panjang teks pada dataset. Model dilatih menggunakan batch size 32, learning rate 0.001, dan 20 epoch. Struktur model dioptimalkan melalui pengaturan parameter seperti jumlah neuron, batch size, learning rate, dan jumlah epoch [21]. Secara spesifik, pelatihan model menggunakan batch size 32, learning rate 0.001, dan 20 epoch. Optimizer yang digunakan adalah Adam karena kemampuannya menstabilkan proses pembelajaran, sedangkan fungsi loss yang digunakan adalah Categorical Cross-Entropy yang sesuai untuk klasifikasi multikelas. Pendekatan ini sejalan dengan survei teknik klasifikasi teks modern yang menunjukkan bahwa arsitektur LSTM dan embedding layer berpengaruh signifikan terhadap performa klasifikasi teks perlu [22].

Pada proses kedua, melakukan pelatihan model pada data training set, pengujian pada validation set untuk memilih model terbaik, dan evaluasi akhir pada testing set. Pelatihan dilakukan secara iteratif hingga loss function berkurang dan akurasi meningkat dengan penerapan early stopping untuk mencegah overfitting. Pada tahap ini, penelitian menguji lima model klasifikasi, yaitu LSTM, Bidirectional LSTM, CNN-LSTM, SVM, dan NB. Setiap model dibangun menggunakan konfigurasi hiperparameter yang konsisten agar perbandingan performa menjadi adil. Ringkasan varian model yang diuji disajikan pada Tabel 1. Untuk memastikan hasil yang andal serta mencegah overfitting, proses evaluasi juga dilakukan menggunakan k-fold Cross-validation dengan nilai 5, sehingga performa yang diperoleh benar-benar merepresentasikan kemampuan generalisasi model.

**Tabel 1.** Varian model yang diuji

| Model       | Tipe Model    | Deskripsi                                     | Alasan Diuji             |
|-------------|---------------|---|--------------------------|
| LSTM        | Deep Learning | LSTM satu arah                                | Baseline DL              |
| Bi-LSTM     | Deep Learning | Memproses urutan dua arah                     | Menguji konteks dua arah |
| CNN-LSTM    | Deep Learning | CNN untuk ekstraksi fitur + LSTM untuk urutan | Menguji fitur lokal      |
| SVM         | Metode Klasik | Kernel-based classifier                       | Pembandingan non-DL      |
| Naïve Bayes | Metode Klasik | Probabilistic classifier                      | Baseline tradisional     |

Selanjutnya, pada proses ketiga memastikan bahwa sistem klasifikasi berita berbasis LSTM berfungsi secara optimal dan sesuai dengan tujuan penelitian. Proses pengujian dilakukan terhadap data uji (testing set) yang belum pernah digunakan selama proses pelatihan model untuk menilai kemampuan generalisasi sistem dalam mengenali pola teks baru. Pengujian dilakukan dengan menggunakan confusion matrix, yang berfungsi untuk mengevaluasi hasil prediksi model terhadap label sebenarnya. Dari confusion matrix tersebut, diperoleh empat metrik utama, yaitu accuracy, precision, recall, dan F1-score. Metrik-metrik ini digunakan untuk mengukur tingkat ketepatan, kelengkapan, serta keseimbangan antara keduanya dalam performa klasifikasi [23]. Sedangkan selama tahap evaluasi, model diuji menggunakan beberapa konfigurasi parameter seperti jumlah neuron, batch size, dan dropout rate untuk menentukan kombinasi yang menghasilkan performa terbaik. Selain itu, dilakukan perbandingan hasil model LSTM dengan metode lain, seperti Support Vector Machine (SVM) dan Naïve Bayes,



sebagai baseline [12]. Perbandingan ini bertujuan untuk melihat sejauh mana model LSTM mampu menangkap ketergantungan jangka panjang antar kata dalam teks berita dibandingkan metode konvensional. Selanjutnya, dilakukan analisis hasil pengujian dengan membandingkan nilai metrik evaluasi terhadap temuan pada penelitian-penelitian terdahulu [24]. Langkah ini bertujuan untuk memvalidasi efektivitas model yang diusulkan dan memastikan hasilnya sejalan atau bahkan lebih baik dari penelitian sebelumnya.

### 2.3 Tahap Laporan

Tahap laporan merupakan tahap akhir dari keseluruhan proses penelitian, di mana seluruh hasil kegiatan mulai dari pengumpulan data, analisis, perancangan sistem, hingga implementasi dan pengujian model didokumentasikan secara sistematis. Pada tahap ini, peneliti menyusun laporan penelitian yang memuat penjelasan metodologi, hasil eksperimen, pembahasan temuan, kesimpulan, serta saran untuk pengembangan penelitian selanjutnya. Laporan disusun agar dapat menjadi acuan ilmiah dan memastikan bahwa seluruh proses dan hasil penelitian dapat dipahami, diverifikasi, serta direplikasi oleh pihak lain.

## 3. HASIL DAN PEMBAHASAN

Tahapan penelitian yang telah dijelaskan sebelumnya menghasilkan beberapa keluaran empiris berupa model klasifikasi berita berbasis LSTM dengan performa yang dibandingkan terhadap varian arsitektur lain seperti Bi-LSTM dan CNN-LSTM. Penelitian ini menggunakan dataset berita daring berbahasa Indonesia yang terdiri dari 48.634 judul artikel dengan lima kategori utama: keuangan, perjalanan, kesehatan, makanan, dan olahraga. Setiap kategori memiliki jumlah data yang relatif seimbang agar menghindari bias klasifikasi. Seluruh data disimpan dalam format CSV (Comma-Separated Values) agar mudah dibaca dan diolah menggunakan Python. Dataset ini kemudian dibagi menjadi tiga bagian utama, yaitu data training (data pelatihan) dan data pengujian (testing data), dengan perbandingan 80:20 untuk memastikan model dapat belajar dari data yang cukup besar dan tetap mampu menggeneralisasi terhadap data baru.

### 3.1 Hasil Perancangan

Hasil penelitian menunjukkan bahwa seluruh tahapan pra-pemrosesan teks berhasil dilakukan sesuai kebutuhan model klasifikasi berbasis LSTM. Proses pra-pemrosesan meliputi pembersihan teks, case folding, tokenisasi, stop-word removal, dan stemming. Tahapan tersebut menghasilkan representasi teks yang lebih bersih dan konsisten sehingga siap digunakan sebagai input model.

Setelah pra-pemrosesan, setiap kata dikonversi ke bentuk representasi numerik melalui proses embedding. Penelitian ini menggunakan embedding layer yang dilatih bersamaan dengan model untuk menangkap hubungan semantik antar kata dalam dataset berita. Hasil embedding menunjukkan bahwa kata-kata dengan konteks serupa memiliki vektor representasi yang saling berdekatan, sehingga mempermudah proses pembelajaran pada lapisan LSTM dan model pembanding lainnya (Bi-LSTM, CNN-LSTM, SVM, dan Naïve Bayes).

Tahap pelatihan model kemudian dilakukan menggunakan dataset yang telah melalui proses embedding tersebut. Evaluasi awal menunjukkan bahwa model mampu mempelajari pola sekuensial dalam teks, dan hasil lengkapnya dibahas pada sub-bagian hasil performa model.

### 3.2 Proses Implementasi

Pada tahap ini dilakukan proses pelatihan model yang diawali dengan serangkaian langkah text preprocessing untuk mengubah data teks mentah menjadi bentuk yang lebih terstruktur sehingga dapat diproses oleh algoritma pembelajaran mesin. Setiap tahapan preprocessing dilakukan menggunakan tool dan pustaka spesifik yang umum digunakan dalam pemrosesan bahasa alami.

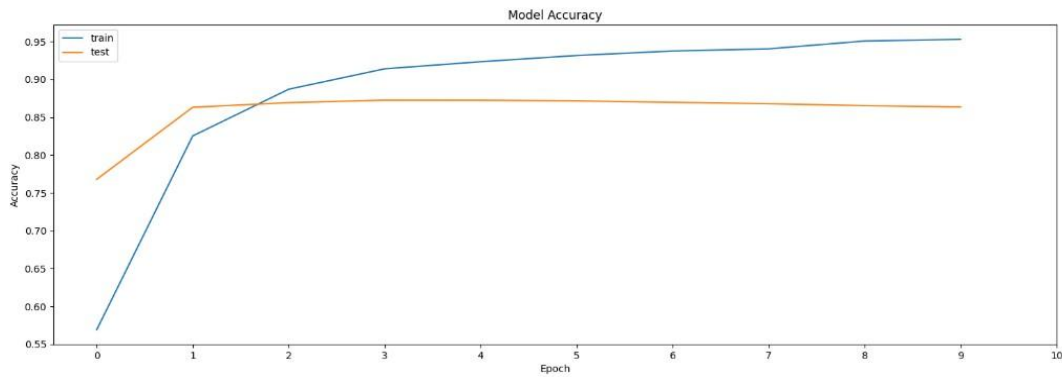
Langkah pertama adalah noise removal, yaitu proses menghilangkan karakter atau elemen yang tidak diperlukan seperti tanda baca, angka, simbol, maupun tautan. Pada tahap ini digunakan pustaka Regular Expression (Regex) Python untuk melakukan pembersihan teks secara otomatis.

Langkah berikutnya adalah case folding, yaitu mengubah seluruh huruf menjadi huruf kecil (.lower()) menggunakan fungsi bawaan Python agar model tidak membedakan kata berdasarkan kapitalisasi. Setelah itu dilakukan tokenization menggunakan modul word\_tokenize dari NLTK, yang memisahkan teks menjadi unit-unit kata berdasarkan spasi atau tanda baca. Hasil tokenisasi ini menjadi dasar untuk tahapan analisis selanjutnya.

Tahap berikutnya adalah normalisasi kata, yaitu mengganti kata tidak baku menjadi bentuk standar merujuk pada Kamus Besar Bahasa Indonesia (KBBI). Pada proses ini digunakan library kbbl (pykbbl) untuk penyesuaian kata baku. Setelah normalisasi selesai, dilakukan stopword removal untuk menghapus kata-kata umum seperti “yang”, “dan”, “atau” menggunakan daftar stopwords Bahasa Indonesia dari NLTK serta tambahan stopwords lokal yang relevan.

Tahap terakhir pada proses pra-pemrosesan adalah stemming, yaitu mengembalikan setiap kata ke bentuk dasarnya. Proses stemming dilakukan menggunakan Sastrawi Stemmer, yang merupakan pustaka paling umum digunakan untuk Bahasa Indonesia. Hasil dari seluruh tahapan ini menunjukkan bahwa dataset teks telah dibersihkan, dinormalisasi, dan siap untuk digunakan pada tahap embedding.

Setelah pra-pemrosesan selesai, seluruh data dikonversi menjadi representasi numerik menggunakan teknik word embedding. Representasi ini memungkinkan model LSTM memahami hubungan semantik antar kata. Selanjutnya dilakukan proses pelatihan model LSTM dengan arsitektur yang terdiri dari lapisan embedding, lapisan LSTM dengan 128 neuron, dan lapisan dense dengan fungsi aktivasi Softmax. Parameter pelatihan ditetapkan menggunakan batch size 32, jumlah epoch 20, dan learning rate sebesar 0.001. Untuk mencegah overfitting diterapkan teknik dropout dengan nilai 0.5 serta early stopping untuk menghentikan pelatihan otomatis ketika performa model tidak menunjukkan peningkatan signifikan.



Gambar 2. Akurasi Pelatihan Model

Hasil pelatihan model menunjukkan peningkatan akurasi yang signifikan pada epoch awal dan mencapai kestabilan setelah beberapa iterasi. Berdasarkan Gambar 2, kurva akurasi data pelatihan (train accuracy) meningkat tajam dari sekitar 0,60 pada epoch pertama hingga mendekati 0,95 pada epoch kesepuluh. Sementara itu, akurasi data pengujian (test accuracy) juga meningkat hingga sekitar 0,88 pada epoch ketiga, kemudian stabil dengan sedikit fluktuasi pada epoch berikutnya.

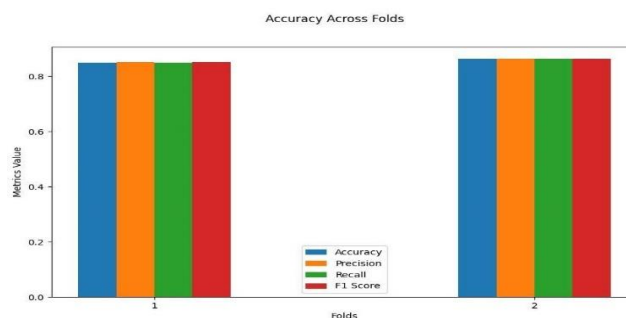
Pola ini menunjukkan bahwa model mampu belajar dengan baik dari data pelatihan tanpa mengalami overfitting. Perbedaan kecil antara akurasi pelatihan dan pengujian menunjukkan kemampuan generalisasi yang baik. Stabilitas akurasi pada test set setelah epoch keempat juga mengindikasikan bahwa penerapan early stopping efektif untuk mengoptimalkan proses pelatihan.

Secara keseluruhan, hasil pelatihan menunjukkan bahwa konfigurasi arsitektur LSTM beserta parameter-parameter pelatihan telah ditetapkan dengan tepat sehingga menghasilkan keseimbangan antara akurasi yang tinggi dan kemampuan generalisasi yang baik. Kinerja ini membuktikan bahwa model klasifikasi berita yang dikembangkan mampu beradaptasi terhadap variasi data teks tanpa penurunan akurasi yang signifikan.

### 3.3 Proses Pengujian

Pengujian sistem untuk menilai kinerja model setelah proses pelatihan selesai. Pengujian dilakukan menggunakan data uji yang belum pernah dilihat oleh model selama pelatihan untuk mengukur kemampuan generalisasi sistem. Total data yang digunakan untuk pengujian sebanyak 20% dari keseluruhan dataset, atau sekitar 9.727 judul berita. Evaluasi kinerja model dilakukan melalui dua pendekatan utama, yaitu Cross-validation dan confusion matrix.

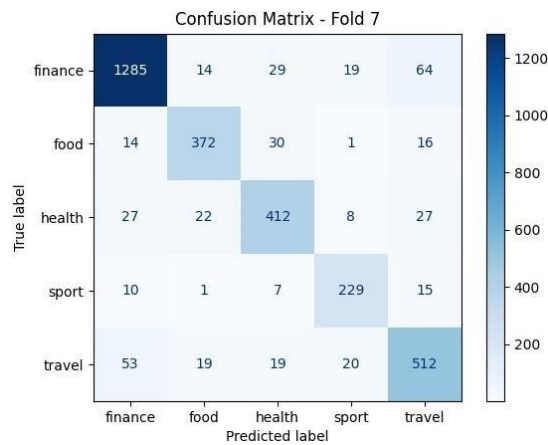
Pada tahap pertama, dilakukan pengujian menggunakan k-fold Cross-validation untuk mengukur stabilitas model pada berbagai pembagian data. Teknik ini membagi dataset ke dalam beberapa fold, lalu melatih dan menguji model secara bergantian pada setiap fold. Hasil rata-rata akurasi Cross-validation adalah 0.80, yang menunjukkan performa model pada kondisi pelatihan yang lebih variatif dan acak. Nilai ini umumnya lebih rendah karena pada setiap fold model hanya dilatih menggunakan sebagian data (80%), sehingga kapasitas generalisasi belum optimal. Selain itu, nilai 0.80 juga merepresentasikan rata-rata performa di seluruh fold, termasuk fold yang memiliki distribusi kelas yang lebih sulit atau tidak seimbang. Grafik tren akurasi dan loss pada proses ini ditunjukkan pada Gambar 3.



Gambar 3. Pengujian Cross Validation

Setelah proses Cross-validation selesai, model kemudian dilatih kembali menggunakan seluruh data pelatihan (train set 80%) dengan konfigurasi dan parameter terbaik yang diperoleh dari hasil tuning. Model final yang telah ditingkatkan performanya ini kemudian diuji menggunakan test set 20% pada evaluasi confusion matrix.

Evaluasi pada confusion matrix (Gambar 4) menunjukkan akurasi yang lebih tinggi, yaitu 90%, dengan presisi 89%, recall 88%, dan F1-score 89%. Perbedaan antara akurasi test set yang lebih tinggi dibandingkan hasil Cross-validation dapat dijelaskan oleh beberapa alasan metodologis. Model final dilatih menggunakan data pelatihan yang lebih banyak, sedangkan pada proses Cross-validation setiap fold hanya memanfaatkan sebagian data sehingga kapasitas pembelajaran model lebih terbatas. Selain itu, model final menggunakan konfigurasi hiperparameter terbaik hasil tuning, sedangkan performa pada setiap fold Cross-validation tidak selalu mencapai kondisi optimal. Variasi distribusi kelas pada masing-masing fold juga berpengaruh, karena beberapa fold mungkin memiliki komposisi data yang lebih sulit sehingga menurunkan rata-rata akurasi. Sementara itu, test set cenderung lebih representatif karena diambil secara proporsional dari keseluruhan dataset, sehingga tidak terpengaruh oleh ketidakseimbangan atau noise yang mungkin terjadi pada fold tertentu. Dengan demikian, perbedaan antara akurasi Cross-validation sebesar 0.80 dan akurasi test set sebesar 0.90 merupakan kondisi yang wajar dan mencerminkan bahwa model final mengalami peningkatan performa setelah melalui proses tuning dan pelatihan penuh.



**Gambar 4.** Pengujian Confusion matrix

Secara keseluruhan, hasil evaluasi menunjukkan bahwa model LSTM memiliki performa klasifikasi yang kuat dan stabil. Cross-validation menunjukkan kemampuan generalisasi model pada berbagai skenario pelatihan, sedangkan hasil confusion matrix menunjukkan kemampuan model final dalam mengklasifikasikan data baru dengan akurasi tinggi.

## 4. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa penerapan metode Long Short-Term Memory (LSTM) pada sistem klasifikasi berita berbahasa Indonesia mampu memberikan kinerja yang tinggi dan stabil dalam mengelompokkan teks ke dalam kategori yang relevan. Model yang dikembangkan berhasil mencapai akurasi sebesar 90%, dengan presisi 89%, recall 88%, dan F1-score 89%. Temuan ini menunjukkan bahwa LSTM efektif dalam menangkap ketergantungan jangka panjang dan memahami konteks semantik dalam teks berita, sekaligus mengisi kesenjangan penelitian yang sebelumnya dijelaskan pada bagian Pendahuluan mengenai kurangnya model klasifikasi berbasis deep learning yang dioptimalkan untuk bahasa Indonesia. Hasil cross-validation dengan akurasi rata-rata 0,80 semakin mengonfirmasi kemampuan generalisasi model tanpa indikasi overfitting. Keberhasilan tersebut menunjukkan bahwa pendekatan LSTM dapat menjadi solusi yang kuat untuk permasalahan klasifikasi berita berbasis teks. Namun, penelitian selanjutnya disarankan untuk memulai dengan melakukan perbandingan terhadap model berbasis Transformer seperti BERT atau IndoBERT, mengingat arsitektur tersebut telah terbukti unggul dalam memahami hubungan kontekstual yang kompleks dalam teks. Selain itu, integrasi varian arsitektur lain seperti Bidirectional LSTM atau LSTM dengan mekanisme Attention juga berpotensi meningkatkan akurasi dan pemahaman konteks dua arah. Pengembangan dataset yang lebih besar, lebih beragam, dan berasal dari berbagai sumber berita juga diperlukan agar model dapat menghasilkan prediksi yang lebih adaptif, representatif, dan relevan untuk implementasi dunia nyata.

## REFERENCES

- [1] D. R. Tobergte and S. Curtis, “Reuters Institute Digital News Report,” *J. Chem. Inf. Model.*, vol. 53, no. 9, pp. 1689–1699, 2013.



- [2] K. Musliadi, Z. Hazriani, and W. Yuyun, “Twitter Social Media Conversion Topic Trending Analysis Using Latent Dirichlet Allocation Algorithm,” *J. Appl. Eng. Technol. Sci.*, vol. 4, no. September 2020, pp. 390–399, 2022, doi: <https://doi.org/10.37385/jaets.v4i1.1143>.
- [3] P. Malik, R. Pandit, A. Chourasia, L. Singh, P. Rane, and P. Chouhan, “Automated Fake News Detection: Approaches, Challenges, and Future Directions,” *Int. J. Intell. Syst. Appl. Eng.*, vol. 11, no. 4, pp. 682–692, 2023.
- [4] K. I. Roumeliotis, N. D. Tselikas, and D. K. Nasiopoulos, “Fake News Detection and Classification: A Comparative Study of Convolutional Neural Networks, Large Language Models, and Natural Language Processing Models,” *Futur. Internet*, vol. 17, no. 1, pp. 1–29, 2025, doi: 10.3390/fi17010028.
- [5] İ. R. Karaş et al., “Fake news and misinformation: A systematic review of detection and impact studies,” *J. Contemp. Soc. Sci. Educ. Stud.*, vol. 5, no. 2, pp. 77–87, 2025, doi: 10.5281/zenodo.16749332.
- [6] Y. Song, X. Liu, and Z. Zhou, “A Comprehensive Review of Text Classification Algorithms,” *J. Electron. Inf. Sci.*, vol. 9, no. 2, pp. 34–42, 2024, doi: 10.23977/jeis.2024.090205.
- [7] D. Ogaga and A. Olalere, “Evaluation and Comparison of SVM, Deep Learning, and Naïve Bayes Performances for Natural Language Processing Text Classification Task,” <https://www.researchgate.net/publication/375773207>, no. November, 2023, doi: 10.20944/preprints202311.1462.v1.
- [8] M. KH, Kaharuddin, and I. Syafrinal, “Diagnosing Android-Based Virus Infections in Children Using Naive Bayes,” *JURTEKSI (Jurnal Teknol. dan Sist. Informasi)*, vol. XI, no. 2, pp. 273–280, 2025, doi: <https://doi.org/10.33330/jurteksi.v11i2.3685>.
- [9] L. J. Tutika, “A Comprehensive Review of Sentiment Analysis Techniques: From Naive Bayes to LSTM-A Review,” *Int. J. Res. Publ. Rev.*, vol. 5, no. 11, pp. 4239–4246, 2024, [Online]. Available: [www.ijrpr.com](http://www.ijrpr.com)
- [10] K. Taha, P. D. Yoo, C. Yeun, D. Homouz, and A. Taha, “A comprehensive survey of text classification techniques and their research applications: Observational and experimental insights,” *Comput. Sci. Rev.*, vol. 54, no. August, p. 100664, 2024, doi: 10.1016/j.cosrev.2024.100664.
- [11] C. Liu, “Long Short-Term Memory (LSTM)-based news classification model,” *PLoS One*, vol. 19, no. 5 May, pp. 1–23, 2024, doi: 10.1371/journal.pone.0301835.
- [12] T. C. Praha, Widodo, and M. Nugraheni, “Indonesian Fake News Classification Using Transfer Learning in CNN and LSTM,” *Int. J. Informatics Vis.*, vol. 8, no. 3, pp. 1213–1221, 2024, doi: 10.62527/joiv.8.2.2126.
- [13] E. P. Widhi, D. H. Fudholi, and S. Hidayat, “Implementation Of Deep Learning For Fake News Classification In Bahasa Indonesia,” *J. Res. Soc. Sci. Econ. Manag.*, vol. 3, no. 02, pp. 370–381, 2023, doi: 10.59141/jrssem.v3i02.546.
- [14] I. Triyadi, B. Prasetyo, and T. L. Nikmah, “News text classification using Long-Term Short Memory (LSTM) algorithm,” *J. Soft Comput. Explor.*, vol. 4, no. 2, pp. 79–86, 2023, doi: 10.52465/josce.v4i2.136.
- [15] A. F. Hanifah and R. Kusumaningrum, “Non-Factoid Answer Selection in Indonesian Science Question Answering System using Long Short-Term Memory (LSTM),” *Procedia Comput. Sci.*, vol. 179, no. 2020, pp. 736–746, 2021, doi: 10.1016/j.procs.2021.01.062.
- [16] S. Li, Z. Wang, S. Yang, X. Luo, D. He, and S. Chan, “Internet of Things intrusion detection : Research and practice of NSENet and LSTM fusion models,” *Egypt. Informatics J.*, vol. 26, no. October 2023, p. 100476, 2024, doi: 10.1016/j.eij.2024.100476.
- [17] A. Glenn, P. LaCasse, and B. Cox, “Emotion classification of Indonesian Tweets using Bidirectional LSTM,” *Neural Comput. Appl.*, vol. 35, no. 13, pp. 9567–9578, 2023, doi: 10.1007/s00521-022-08186-1.
- [18] M. Liang and T. Niu, “Research on Text Classification Techniques Based on Improved TF-IDF Algorithm and LSTM Inputs,” *Procedia Comput. Sci.*, vol. 208, pp. 460–470, 2022, doi: 10.1016/j.procs.2022.10.064.
- [19] B. Zhang, C. Song, Y. Li, and X. Jiang, “Spatiotemporal prediction of O3 concentration based on the KNN-Prophet-LSTM model,” *Heliyon*, vol. 8, no. 11, p. e11670, 2022, doi: 10.1016/j.heliyon.2022.e11670.
- [20] W. K. Sari, D. P. Rini, and R. F. Malik, “Text Classification Using Long Short-Term Memory With GloVe Features,” *J. Ilm. Tek. Elektro Komput. dan Inform.*, vol. 5, no. 2, p. 85, 2020, doi: 10.26555/jiteki.v5i2.15021.
- [21] J. Alghamdi, Y. Lin, and S. Luo, “Fake news detection in low-resource languages: A novel hybrid summarization approach,” *Knowledge-Based Syst.*, vol. 296, no. May, p. 111884, 2024, doi: 10.1016/j.knosys.2024.111884.
- [22] A. Ba Alawi and F. Bozkurt, “Performance Analysis of Embedding Methods for Deep Learning-Based Turkish Sentiment Analysis Models,” *Arab. J. Sci. Eng.*, vol. 50, no. 10, pp. 7299–7321, 2025, doi: 10.1007/s13369-024-09360-4.
- [23] B. Liu, J. Chen, R. Wang, J. Huang, Y. Luo, and J. Wei, “Optimizing News Text Classification with Bi-LSTM and Attention Mechanism for Efficient Data Processing,” *2024 5th Int. Conf. Intell. Comput. Human-Computer Interact. ICHCI 2024*, pp. 281–285, 2024, doi: 10.1109/ICHCI63580.2024.10808002.
- [24] Maspiati, Nasruddin, and K. Musliadi, “Sistem Informasi Pemesanan Baju Olahraga Berbasis Web Pada Konveksi ‘Adher’ Tammerodo Sendana Majene,” *Sinov. - Sci. Lit. Innov. Technol. J.*, vol. 01, no. 01, pp. 7–13, 2024.