

LAPORAN PROJEK PEMBELAJARAN MESIN 2

” Klasifikasi Teks Berita Menggunakan Arsitektur *Bidirectional Long Short-Term Memory* (BiLSTM)”



Dosen pengampu :

Al-Ustadz Dr. Oddy Virgantara Putra, S.Kom., M.T.

Disusun oleh :

Nasywa kynda sanina 442023618074

PROGRAM STUDI TEKNIK INFORMATIKA

UNIVERSITAS DARUSSALAM GONTOR MANTINGAN

2025/1447

A. Latar Belakang

Di era digital ini, jumlah data teks yang dihasilkan setiap hari sangatlah besar, mulai dari berita, media sosial, hingga ulasan produk. Mengelola dan memahami informasi ini secara manual adalah hal yang tidak mungkin. Klasifikasi teks adalah proses otomatis untuk mengkategorikan teks ke dalam kelas-kelas yang telah ditentukan. Teknologi ini sangat penting untuk berbagai aplikasi, seperti menyaring email spam, menganalisis sentimen pelanggan, dan mengorganisir artikel berita berdasarkan topiknya.

Tugas ini bertujuan untuk membangun sebuah mode *machine learning* yang mampu mengklasifikasikan teks berita ke dalam kategori yang sesuai (misalnya, olahraga, teknologi, bisnis), dengan ruang lingkup meliputi:

- Mempersiapkan dataset teks berita.
- Melakukan pra-pemrosesan data teks agar siap digunakan oleh model.
- Merancang, melatih, dan mengimplementasikan mode jaringan syaraf tiruan berulang (*Recurrent Neural Network*, RNN), khususnya jenis *bidirectional Long Short-Term Memory* (BiLSTM).
- Mengevaluasi kinerja model menggunakan metrik standar seperti akurasi dan *confussion matrix*.

B. Metodologi

1) Dataset

Dataset yang digunakan dalam tugas ini berasal dari dataset public di Kaggle *BBC-full-text-document-classification*. Di dalamnya berisi dataset berita BBC sejumlah 2225 berita yang telah dikelompokkan ke dalam lima kategori, yaitu *Bussiness*, *Entertainment*, *Tech*, *Politics*, dan *Sport*. Setiap baris data telah berisi artikel dan label kategori yang sesuai. Panjang teks artikel bervariasi, mulai dari paragraf pendek hingga beberapa paragraf panjang. Gaya bahasa yang digunakan juga adalah gaya jurnalistik formal:

"Virus poses Trojan threat to phones A new Trojan virus is targeting mobile phones running the Symbian operating system. The virus, called Skulls, has been created to replace Symbian application icons with images of a skull and crossbones. It also disables the phone so it can only receive and make calls. SMS and MMS messaging do not work..." (berita tech)

Dataset ini dipilih karena merupakan dataset standar untuk tugas klasifikasi teks. Kategori-kategorinya jelas dan telah terdefinisi dengan baik, serta jumlah data per kelasnya cukup seimbang. Hal ini menjadikannya dataset yang ideal untuk melatih dan mengevaluasi model klasifikasi teks secara adil.

2) Arsitektur RNN

Model yang digunakan adalah *Bidirectional Long Short-Term Memory* (BiLSTM). Model ini merupakan pengembangan dari RNN standar yang dirancang untuk mengatasi masalah ‘ingatan jangka pendek’ dan mampu memproses data sekuensial (seperti teks) dari dua arah (depan ke belakang atau sebaliknya). Arsitektur model yang dibangun adalah sebagai berikut:

- **Embedding Layer:** Mengubah setiap kata (yang sudah diubah menjadi angka/token) menjadi vektor numerik dengan dimensi 128. Lapisan ini mempelajari representasi kata berdasarkan konteksnya.
- **BiLSTM layer (1):** Lapisan BiLSTM pertama dengan 64 unit. Lapisan ini memproses urutan kata dari kedua arah untuk menangkap konteks secara lebih baik. *return_sequences=True* berarti lapisan ini akan memberikan output berupa seluruh sekuens ke lapisan berikutnya.
- **Dropout layer (1):** Dropout dengan rate 0 berfungsi untuk mencegah overfitting dengan menonaktifkan sebagian neuron secara acak selama pelatihan.
- **BiLSTM layer (2):** Lapisan ini menerima output dari lapisan sebelumnya dan menghasilkan satu vektor output yang merangkum informasi dari seluruh kalimat.
- **Dropout layer (2):** dengan rate 0.5, dropout kedua ini membantu regularisasi agar model tidak terlalu menghafal data latih.
- **Dense layer:** Lapisan tersembunyi dengan 64 neuron dan fungsi aktivasi ReLU untuk mempelajari pola yang lebih kompleks dari data.
- **Output layer:** Lapisan output dengan 5 neuron (sesuai jumlah kelas) dan fungsi aktivasi Softmax. Softmax menghasilkan probabilitas untuk setiap kelas, dan kelas dengan probabilitas tertinggi akan menjadi prediksi akhir.

3) Pre-processing

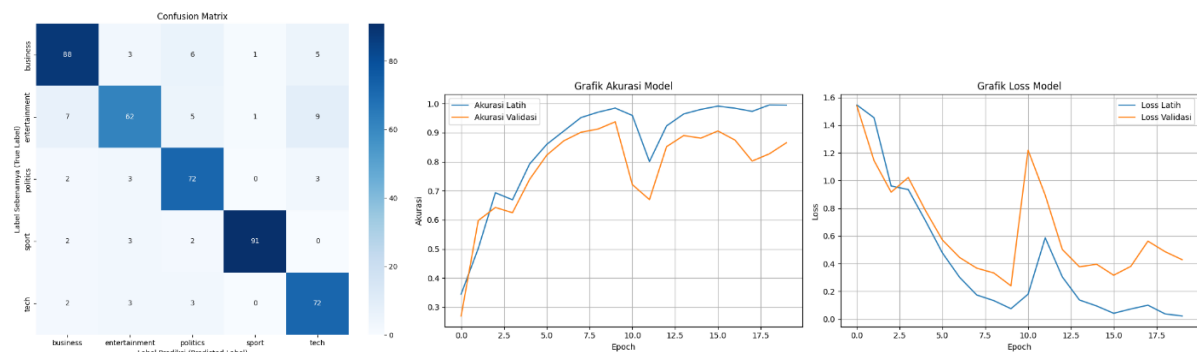
Sebelum data dimasukkan ke model, dataset dilakukan normalisasi terlebih dahulu dengan label encoding, tokenisasi, dan padding guna menyamakan panjang setiap sekuens (kalimat) menjadi 200 token.

4) Pengaturan Eksperimen

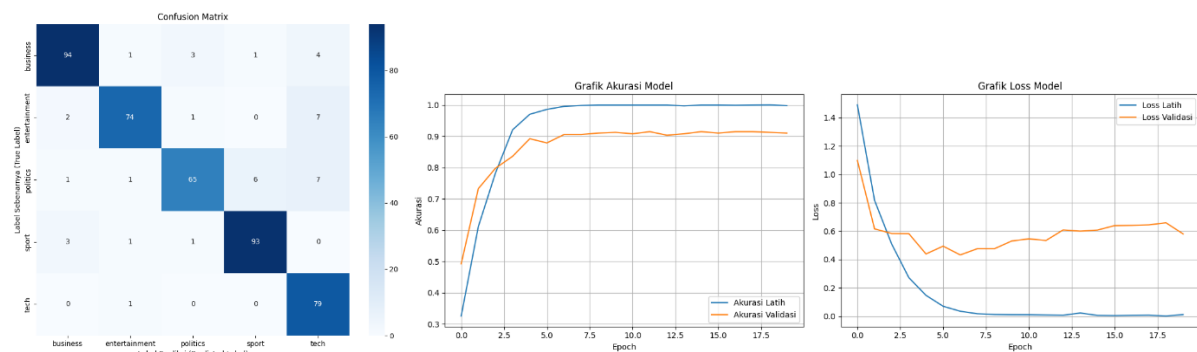
- Jumlah epoch : 20 epoch
- Batch size : 32
- Optimizer : Adam, karena merupakan algoritma optimisasi yang populer dan efektif untuk memperbarui bobot model selama pelatihan.
- *Loss function : Sparse Categorical Crossentropy*

5) Log Eksperimen

Berikut adalah hasil proses pengembangan model untuk mendapatkan hasil terbaik:



Gambar 1. Hasil Eksperimen dengan LSTM



Gambar 2. Hasil Eksperimen dengan BiLSTM

| Percobaan | Model/Konfigurasi | Dropout | Optimizer | Akurasi Validasi | Catatan |
|-----------|------------------------------------|---------|-----------|------------------|---|
| #1 | 1 Lapis LSTM (64 unit) | 0 | Adam | ~85% | Model sederhana, cepat dilatih. Mulai menunjukkan tanda overfitting setelah epoch ke-10. Akurasi validasi stabil di sekitar 85%. |
| #2 | 2 Lapis BiLSTM (32 unit per lapis) | 0.5 | Adam | ~91% | Model lebih kompleks dengan BiLSTM bertingkat, akurasi validasi meningkat hingga ~91%. Namun, grafik menunjukkan adanya gap besar dengan akurasi latih → indikasi overfitting ringan . |

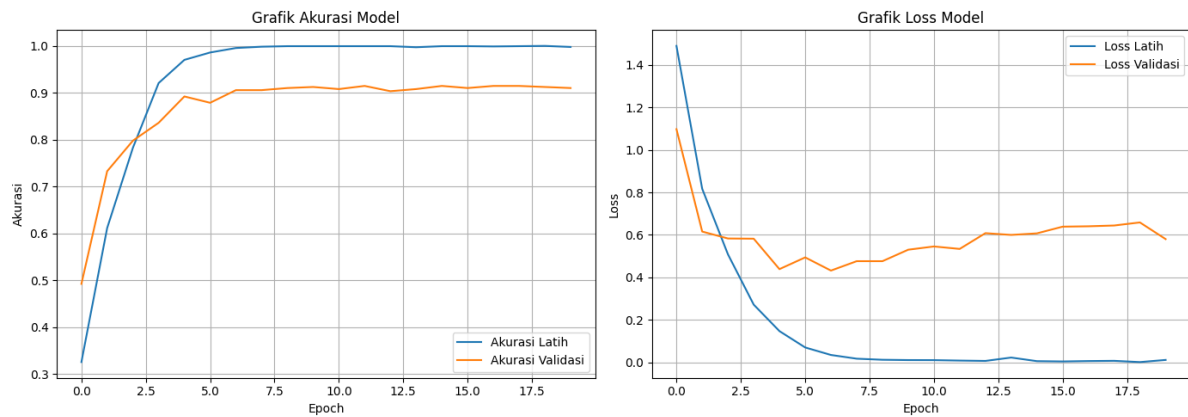
Proses pengembangan model dimulai dengan eksperimen awal menggunakan arsitektur sederhana, yaitu 1 lapis LSTM (64 unit) sebagai *baseline*. Model ini mencapai akurasi validasi sekitar ~85%, namun menunjukkan tanda-tanda *overfitting* setelah epoch ke-10 (Percobaan #1).

Untuk meningkatkan performa dan mengatasi *overfitting*, model diubah secara signifikan pada percobaan kedua. Arsitektur diganti menjadi 2 lapis BiLSTM (32 unit per lapis) dan ditambahkan Dropout 0.5. Alasan utamanya adalah agar pemahaman konteks tidak hanya bergantung pada informasi sebelumnya (masa lalu) tetapi juga informasi sesudahnya (masa depan) dalam sebuah urutan teks. BiLSTM mampu memproses data dari kedua arah, sehingga diharapkan dapat menangkap representasi makna yang lebih kaya dan komprehensif. Penambahan Dropout juga diperuntukkan untuk mencegah model menjadi terlalu bergantung pada neuron tertentu selama pelatihan, sehingga meningkatkan kemampuannya untuk generalisasi pada data yang belum pernah dilihat. Perubahan ini bertujuan agar model dapat memahami konteks teks dari dua arah (maju dan mundur) serta lebih tahan terhadap *overfitting*.

Hasilnya, performa model meningkat tajam dengan akurasi validasi mencapai ~91% (Percobaan #2). Peningkatan signifikan ini membuktikan bahwa arsitektur BiLSTM yang lebih kompleks lebih efektif untuk tugas ini, sehingga dipilih sebagai model final.

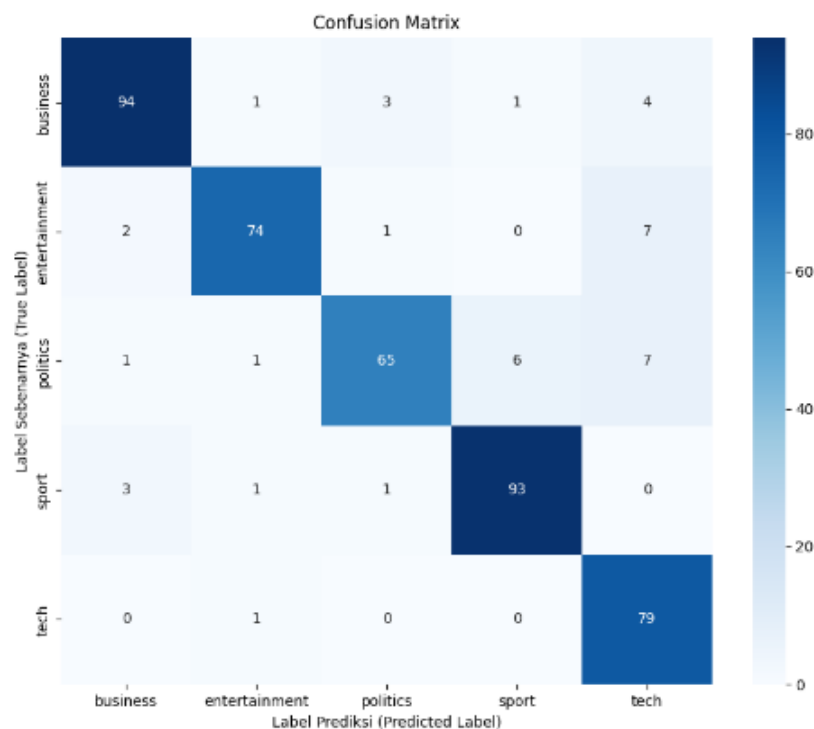
C. Evaluasi Hasil

Evaluasi model dilakukan untuk mengukur performa arsitektur BiLSTM bertingkat yang telah dilatih. Metrik utama yang digunakan adalah akurasi, loss, dan confusion matrix, didukung oleh visualisasi learning curve untuk menganalisis proses pelatihan. Model BiLSTM yang diimplementasikan (Percobaan #2) berhasil mencapai akurasi validasi puncak sekitar 91%. Performa ini menunjukkan peningkatan yang signifikan dibandingkan dengan model dasar LSTM (Percobaan #1) yang hanya mencapai akurasi ~85%.



Dari kurva pembelajaran di atas, dapat dianalisis bahwa model belajar dengan sangat cepat, di mana akurasi validasi meningkat tajam pada epoch-epoch awal dan kemudian stabil di angka 90%-91%. Perilaku *loss* pada grafik memberikan indikasi yang jelas masih *overfitting*. Kurva *loss* latih (biru) terus menurun secara konsisten hingga mendekati nol, yang berarti model berhasil meminimalkan kesalahan pada data yang dilihatnya berulang kali. Namun, kurva *loss* validasi (oranye) berhenti menurun setelah sekitar epoch ke-7 dan mulai bergerak stagnan bahkan sedikit naik. Celah (gap) yang semakin melebar antara kedua kurva ini adalah tanda klasik di mana model kehilangan kemampuannya untuk menggeneralisasi pola pada data baru yang belum pernah dilihat sebelumnya.

Meskipun menunjukkan gejala *overfitting*, penerapan *dropout* 0.5 telah membantu menahan agar *overfitting* tidak terlalu parah, namun tidak menghilangkannya sepenuhnya. Untuk melihat performa klasifikasi pada tiap-tiap kelas, digunakan *confusion matrix*.



Confusion matrix di atas menunjukkan bahwa model memiliki performa yang sangat baik di hampir semua kategori. Prediksi di kategori *business* (94 benar dari 103) dan *sport* (93 benar dari 98) memiliki tingkat prediksi yang sangat akurat. Sedangkan kesalahan prediksi paling banyak terjadi pada kategori *politics*, di mana beberapa artikel salah diklasifikasikan sebagai *business* (1 artikel) dan *tech* (6 artikel). Hal ini kemungkinan disebabkan oleh adanya tumpang tindih kosakata atau topik antara berita politik dengan berita bisnis dan teknologi.

Secara keseluruhan, meskipun ada indikasi *overfitting* ringan yang perlu diwaspadai, arsitektur BiLSTM bertingkat tetap menjadi pilihan model terbaik karena menghasilkan akurasi tertinggi.

D. Refleksi Pribadi

Tantangan utama dalam proyek ini adalah mengatasi *overfitting*. Model pertama (LSTM) sudah menunjukkan gejala *overfitting* setelah epoch ke-10. Ketika arsitektur diubah menjadi BiLSTM dengan tujuan agar model dapat menangkap dependensi kontekstual dari kedua arah teks, hasilnya sangat positif, akurasi validasi meningkat secara signifikan dari ~85% menjadi ~91%.

Tetapi risiko *overfitting* menjadi lebih besar karena model memiliki kapasitas lebih tinggi untuk menghafal data latih. Untuk mengatasi masalah *overfitting*, solusi yang diterapkan adalah menambahkan lapisan *Dropout* dengan laju 0.5 di antara lapisan BiLSTM. Hasilnya cukup efektif. Meskipun gejala *overfitting* ringan masih terlihat, *dropout* berhasil mencegah penurunan performa drastis pada data validasi dan memungkinkan model mencapai akurasi yang lebih tinggi.

Dalam pengerjaan tugas ini, AI generatif digunakan untuk membantu beberapa hal, seperti debugging kode dan penjelasan tambahan mengenal logika model. Pelajaran penting dari eksperimen ini adalah pentingnya eksperimen yang terstruktur. Memulai dengan model *baseline* yang sederhana (LSTM) memberikan tolok ukur yang jelas untuk mengevaluasi model yang lebih kompleks (BiLSTM). Hal ini membuktikan bahwa penambahan kompleksitas harus dijustifikasi dengan peningkatan performa yang terukur. Selain itu, tugas ini juga mengajarkan bahwa *overfitting* adalah masalah nyata yang harus selalu diwaspadai dan ditangani dengan teknik regularisasi yang sesuai seperti salah satunya dengan *dropout*.

E. Kesimpulan dan Saran

Berdasarkan serangkaian eksperimen yang telah dilakukan, model terbaik untuk tugas klasifikasi teks berita adalah arsitektur BiLSTM bertingkat dengan 2 lapis dan dropout rate 0.5.

Model ini berhasil mencapai akurasi validasi sekitar 91%, sebuah peningkatan performa yang signifikan dibandingkan model baseline LSTM. Model ini menunjukkan kemampuan yang kuat dalam membedakan lima kategori berita, dengan tingkat kesalahan klasifikasi yang tergolong rendah.

Meskipun hasil yang dicapai sudah sangat baik, ada beberapa area yang dapat dieksplorasi untuk pengembangan selanjutnya, seperti penggunaan *Pre-trained Embeddings* atau *Hyperparameter Tuning*.

F. Referensi Dataset

<https://www.kaggle.com/datasets/alfatherry/bbc-full-text-document-classification/data>