

TUGAS MACHINE LEARNING 2

Laporan Tugas Individu : Klasifikasi Teks Menggunakan RNN

Dosen Pengampu : Dr. Oddy Virgantara Putra, S.Kom., M.T



Disusun Oleh:

Nama : Muhammad Rafi Aditya

NIM : 442023611057

**PRODI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS DARUSSALAM GONTOR**

DAFTAR ISI

DAFTAR ISI.....	2
BAB 1 PENDAHULUAN	3
BAB 2 DATASET	4
BAB 3 IMPLEMENTASI MODEL.....	4
3.1 ARSITEKTUR RNN	4
3.2 PREPROCESSING.....	5
3.3 PENGATURAN EKSPERIMEN.....	5
3.4 LOG EKSPERIMEN	6
BAB 4 EVALUASI HASIL.....	7
BAB 5 REFLEKSI PRIBADI.....	8
BAB 6 KESIMPULAN & SARAN	9
REFERENSI	10

BAB 1

PENDAHULUAN

Tugas ini bertujuan untuk membangun model klasifikasi teks berbasis deep learning dengan pendekatan Recurrent Neural Network (RNN), khususnya menggunakan arsitektur Long Short-Term Memory (LSTM). Model ini dirancang untuk mengklasifikasikan opini dalam teks ulasan film menjadi dua kelas, yaitu positif dan negatif. Data yang digunakan diambil dari dataset IMDb dan terdiri dari masing-masing 100 data opini positif dan negatif.

Pendekatan RNN dipilih karena arsitektur ini dirancang untuk memproses data sekuensial seperti teks, sehingga mampu memahami konteks kata berdasarkan urutan kemunculannya. LSTM digunakan karena memiliki kemampuan untuk mengatasi permasalahan vanishing gradient yang sering muncul pada RNN standar saat memproses urutan panjang.

Penggunaan LSTM dalam klasifikasi teks diharapkan dapat memberikan hasil klasifikasi yang lebih akurat, terutama dalam memahami konteks opini dalam kalimat ulasan. Meskipun data yang digunakan terbatas, tugas ini memberikan gambaran nyata bagaimana membangun sistem klasifikasi berbasis RNN secara end-to-end, mulai dari preprocessing teks hingga evaluasi performa model.

BAB 2 DATASET

Dataset yang digunakan berasal dari Kaggle dengan nama "**IMDB Dataset of 50K Movie Reviews**". Saya mengambil 100 data opini positif dan 100 opini negatif untuk digunakan sebagai dataset klasifikasi dua kelas. Data terdiri dari teks ulasan film dengan panjang dan gaya bahasa yang bervariasi.

Alasan pemilihan dataset:

- Data terbuka dan mudah diakses
- Relevan dengan topik klasifikasi opini
- Berkualitas tinggi dan banyak digunakan dalam eksperimen NLP

BAB 3 IMPLEMENTASI MODEL

3.1 Arsitektur RNN

Model yang digunakan adalah LSTM dengan susunan layer sebagai berikut:

- **Embedding Layer:** untuk mengubah indeks kata menjadi vektor berdimensi 128
- **LSTM Layer:** 64 unit, dropout 0.5
- **Dense Layer:** 1 unit dengan aktivasi sigmoid untuk klasifikasi biner

Model Summary:

Layer (type)	Output Shape	Param #
embedding_1 (Embedding)	(None, 200, 128)	974,976
lstm_1 (LSTM)	(None, 64)	49,408
dense_1 (Dense)	(None, 1)	65

Total params: 1,024,449 (3.91 MB)
Trainable params: 1,024,449 (3.91 MB)
Non-trainable params: 0 (0.00 B)

3.2 Preprocessing

- Tokenisasi teks dilakukan menggunakan Tokenizer dari Keras
- Teks diubah menjadi urutan angka (sequence), kemudian dipadatkan (pad_sequences) hingga panjang maksimum 200
- Label dikonversi menjadi angka 0 (negatif) dan 1 (positif)

3.3 Pengaturan Eksperimen

- Optimizer: Adam
- Loss function: Binary Crossentropy
- Epoch: 10
- Batch size: 32
- Validation split: 20%

3.4 Log Eksperimen

Berikut hasil pelatihan model selama 10 epoch:

Epoch 1/10 - accuracy: 0.5756 - val_accuracy: 0.0000 - loss: 0.6913 - val_loss: 0.7211

Epoch 2/10 - accuracy: 0.6532 - val_accuracy: 0.0000 - loss: 0.6776 - val_loss: 0.7692

Epoch 3/10 - accuracy: 0.6155 - val_accuracy: 0.0000 - loss: 0.6594 - val_loss: 0.9014

Epoch 4/10 - accuracy: 0.5933 - val_accuracy: 0.0000 - loss: 0.6311 - val_loss: 1.2741

Epoch 5/10 - accuracy: 0.6793 - val_accuracy: 0.0000 - loss: 0.5353 - val_loss: 1.0344

Epoch 6/10 - accuracy: 0.6391 - val_accuracy: 0.0000 - loss: 0.5317 - val_loss: 1.0795

Epoch 7/10 - accuracy: 0.7110 - val_accuracy: 0.0500 - loss: 0.4661 - val_loss: 0.8464

Epoch 8/10 - accuracy: 0.9741 - val_accuracy: 0.4250 - loss: 0.4415 - val_loss: 0.7313

Epoch 9/10 - accuracy: 0.9850 - val_accuracy: 0.4250 - loss: 0.4309 - val_loss: 0.7281

Epoch 10/10 - accuracy: 0.9893 - val_accuracy: 0.4250 - loss: 0.4270 - val_loss: 0.7267

BAB 4

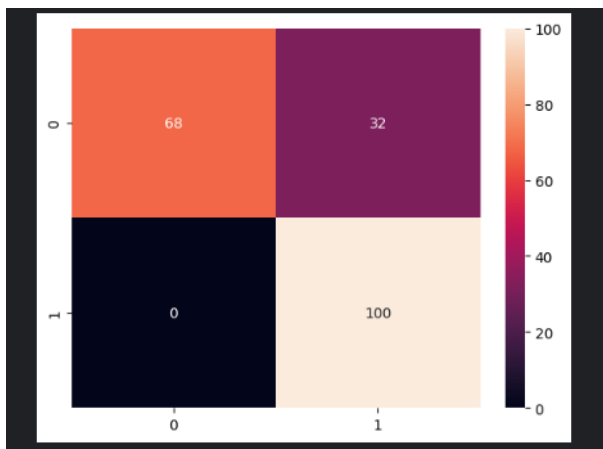
EVALUASI HASIL

Evaluasi dilakukan dengan dua pendekatan utama: confusion matrix dan classification report. Selain itu, ditampilkan juga grafik akurasi dan loss selama proses pelatihan untuk melihat tren performa model.

Hasil evaluasi menunjukkan bahwa model cukup mampu membedakan antara opini positif dan negatif, walaupun terjadi gap antara akurasi pelatihan dan akurasi validasi. Hal ini mengindikasikan adanya overfitting, di mana model belajar terlalu baik pada data latih namun kurang mampu menggeneralisasi pada data validasi.

Berdasarkan hasil evaluasi, model mencapai **akurasi pelatihan sebesar 98.9%**, sedangkan **akurasi validasi tertinggi hanya mencapai 42.5%**. Confusion matrix menunjukkan bahwa model lebih banyak mengklasifikasikan opini ke salah satu kelas tertentu, yang memperkuat indikasi overfitting.

Secara keseluruhan, model LSTM dapat mengenali pola dasar dari opini dalam teks, namun masih memerlukan perbaikan dari sisi generalisasi. Langkah selanjutnya yang direkomendasikan adalah memperluas jumlah data, melakukan eksplorasi regularisasi lanjutan, dan mencoba arsitektur lain seperti BiLSTM atau GRU.



BAB 5

REFLEKSI PRIBADI

Selama mengerjakan tugas klasifikasi teks ini, saya belajar banyak hal baru, terutama tentang bagaimana membangun dan melatih model RNN (khususnya LSTM) dalam proyek klasifikasi opini.

Pada awalnya, saya sempat mengalami kesulitan dalam memahami bagaimana cara kerja tokenisasi teks dan mengatur format input agar sesuai dengan kebutuhan model. Saya juga mengalami error saat membuat model, terutama pada bagian embedding dan input shape. Setelah membaca dokumentasi dan berdiskusi dengan teman, saya menyadari pentingnya memahami alur preprocessing teks secara menyeluruh sebelum melatih model.

Salah satu tantangan lain adalah ketika saya menjalankan training model. Akurasi pelatihan sangat tinggi, namun akurasi validasi rendah, yang membuat saya menyadari bahwa model mengalami overfitting. Dari situ saya belajar tentang pentingnya evaluasi dan eksplorasi parameter seperti dropout, serta validasi performa model secara obyektif.

Saya juga mempelajari bagaimana debugging dengan sabar, dan memahami error tidak selalu berarti kegagalan, tapi bisa menjadi petunjuk untuk memperbaiki arsitektur atau pendekatan kita. Saya dibantu oleh AI (seperti ChatGPT) untuk memahami kesalahan dan menyusun kembali arsitektur model yang benar, terutama saat saya masih kebingungan dengan bentuk input pada layer embedding.

Tugas ini membuat saya lebih percaya diri dalam memahami alur kerja klasifikasi teks, mulai dari pemilihan dataset, preprocessing, pembuatan model, hingga evaluasi hasil secara sistematis.

BAB 6

KESIMPULAN & SARAN

Dengan menggunakan arsitektur LSTM, model dapat dilatih secara efisien untuk mengklasifikasikan opini dalam teks ulasan film menjadi dua kelas, yaitu positif dan negatif, meskipun jumlah data yang digunakan terbatas. Seluruh proses mulai dari pengumpulan dan persiapan data, pembuatan model, pelatihan, hingga evaluasi memberikan pengalaman nyata dalam menerapkan konsep deep learning untuk pemrosesan bahasa alami (NLP).

Model menunjukkan performa pelatihan yang tinggi, namun akurasi validasinya masih rendah karena kemungkinan overfitting. Hal ini memberikan wawasan penting bahwa pengembangan model tidak hanya soal arsitektur, tetapi juga membutuhkan perhatian terhadap kualitas data, proses regularisasi, dan evaluasi menyeluruh.

Untuk pengembangan lebih lanjut, disarankan untuk menggunakan data yang lebih besar dan beragam, menambahkan teknik regularisasi seperti dropout tambahan, serta mengeksplorasi model alternatif seperti Bidirectional LSTM atau GRU. Selain itu, penggunaan pre-trained word embeddings juga dapat menjadi opsi untuk meningkatkan representasi teks dan performa model.

REFERENSI

- Kaggle - IMDB Dataset of 50K Movie Reviews
- TensorFlow Documentation: Text Classification with RNN
- François Chollet. *Deep Learning with Python*. Manning Publications.