

Klasifikasi Emosi Berbasis Teks Menggunakan Metode Artificial Neural Network (ANN)

Mutiara Tyas Putri Andaryanto¹, Ledy Carina Kristie², Desi Natalia Pratiwi³, Pramudya Riandanu⁴, Michael Bima Kurniandi⁵.

¹Jurusan Matematika, Fakultas Sains dan Matematika, Universitas Kristen Satya Wacana, Salatiga, Jawa Tengah 50711

662022011@student.uksw.edu

ABSTRAK

Penelitian ini bertujuan untuk mengembangkan dan mengevaluasi model klasifikasi emosi berbasis teks menggunakan pendekatan *Artificial Neural Network* (ANN). Data yang digunakan bersumber dari Kaggle dengan total sebanyak 388.809 data teks berlabel enam kategori emosi, yaitu *joy*, *sadness*, *anger*, *fear*, *love*, dan *surprise*. Proses penelitian meliputi tahapan pra-pemrosesan data berupa tokenisasi, padding, dan encoding label, kemudian dilanjutkan dengan pembangunan model ANN yang terdiri dari layer embedding, global average pooling, dense layer, dan softmax sebagai output. Model dilatih menggunakan fungsi loss *sparse_categorical_crossentropy* dan optimizer adam, serta dikendalikan dengan *early stopping* untuk mencegah overfitting. Hasil pelatihan menunjukkan bahwa model mampu mencapai akurasi sebesar 90,29% pada data uji. Selain itu, metrik evaluasi seperti precision, recall, dan f1-score juga menunjukkan performa yang baik, terutama pada label emosi *joy* dan *sadness*. Penelitian ini menunjukkan bahwa ANN dapat menjadi solusi efektif dalam pengembangan sistem klasifikasi emosi berbasis teks, yang berpotensi diterapkan dalam berbagai aplikasi seperti chatbot, sistem monitoring opini publik, hingga layanan konseling daring.

Kata kunci: Emosi, Teks, *Artificial Neural Network*, Klasifikasi, Tokenisasi, *Deep Learning*

PENDAHULUAN

Artificial Neural Network (ANN) merupakan salah satu pendekatan yang berkembang pesat dalam bidang *artificial intelligence* dan banyak diterapkan untuk menyelesaikan masalah klasifikasi, prediksi, dan pengenalan pola. ANN bekerja dengan meniru cara kerja jaringan saraf manusia melalui sistem lapisan-lapisan neuron yang saling terhubung dan dapat belajar dari data. Dalam beberapa tahun terakhir, ANN menjadi teknologi yang sangat potensial untuk menangani data tidak terstruktur, seperti teks dan suara, khususnya dalam tugas pengenalan emosi. Kanger & Bathla. (2017) menunjukkan bahwa kombinasi ANN dengan *fuzzy logic* mampu mengklasifikasikan emosi dalam teks menjadi tiga kelas utama (senang, sedih, dan marah) dengan akurasi mencapai 90%. Keberhasilan ini menunjukkan kemampuan ANN untuk beradaptasi dengan konteks emosional yang kompleks dalam teks.

Seiring berkembangnya media sosial dan *platform* digital, volume data berbasis teks yang mengandung muatan emosional terus meningkat. Berbagai metode telah dikembangkan untuk mendeteksi emosi dari teks, mulai dari pendekatan berbasis kata kunci hingga pendekatan *machine learning*. Bharti et al. (2022) menjelaskan bahwa pendekatan berbasis *deep learning* seperti *Convolutional Neural Network* (CNN) dan *Bidirectional Gated Recurrent Unit* (Bi-GRU) menunjukkan performa unggul dalam mengklasifikasikan emosi dari teks informal seperti *tweet* dan dialog, dengan akurasi mencapai 80,11%. ANN memungkinkan proses ekstraksi fitur emosional secara otomatis tanpa perlu aturan linguistik yang eksplisit. Hal ini menjadikan ANN sangat cocok untuk menangani variasi bahasa dan bentuk ekspresi emosional yang luas.

Penelitian oleh Martinez-Navarro et al. (2023) menunjukkan bahwa arsitektur ANN seperti *Spiking Neural Networks* (SNN), CNN, dan *Multi-Layer Perceptron* (MLP) memiliki keunggulan masing-masing dalam mendeteksi emosi dari *audio*. Dalam studi tersebut, akurasi terbaik yang dicapai oleh model CNN dalam klasifikasi emosi berbasis *audio* adalah sebesar 85,4%, menunjukkan kemampuan CNN dalam mengekstraksi fitur spasial yang kompleks dari sinyal suara. Penggunaan berbagai arsitektur ANN memberikan fleksibilitas dalam menyesuaikan model dengan jenis data dan tujuan pengklasifikasian emosi yang spesifik. Perbandingan performa antar model ini menjadi acuan dalam pengembangan sistem pengenalan emosi yang lebih akurat dan efisien. ANN juga berperan penting dalam mengintegrasikan berbagai jenis data, termasuk *audio*, teks, dan *video* dalam satu kerangka klasifikasi.

Studi sistematis yang dilakukan oleh Chutia & Baruah. (2024) mengulas lebih dari 300 penelitian tentang deteksi emosi berbasis teks menggunakan pendekatan *deep learning*, termasuk *Long Short-Term Memory (LSTM)*, *Bi-LSTM*, dan *BERT*. Beberapa model yang direview dalam studi ini, seperti *Bi-LSTM* dan *BERT*, mampu mencapai akurasi lebih dari 85% dalam klasifikasi enam hingga delapan kategori emosi. Penelitian tersebut menunjukkan bahwa model ANN canggih mampu memberikan hasil yang lebih baik dibandingkan metode tradisional, terutama dalam konteks *big data* yang diperoleh dari media sosial. Penggunaan *pre-trained models* dan *transfer learning* juga memperkuat performa ANN dalam mengatasi keterbatasan data berlabel. Tidak hanya itu, ANN juga mendukung pengembangan sistem *multimodal* yang menggabungkan teks dengan data visual atau suara untuk meningkatkan akurasi pengenalan emosi. Hal ini sangat relevan dalam dunia nyata, seperti layanan pelanggan otomatis, analisis opini publik, hingga deteksi gangguan kesehatan mental.

Meskipun memiliki banyak keunggulan, penerapan ANN dalam deteksi emosi teks menghadapi sejumlah tantangan. Teks mengandung struktur bahasa yang kompleks, ambiguitas makna, serta konteks sosial dan budaya yang beragam. Kumar S. & Geetha. (2024) menekankan pentingnya integrasi ANN dengan teknik *Natural Language Processing (NLP)* agar model dapat memahami konteks emosional dengan lebih baik. Dalam penelitian mereka, ANN dipadukan dengan pendekatan leksikal seperti *NRCLex* untuk memperkuat klasifikasi emosi berbasis teks pendek. Dengan pendekatan tersebut, model mampu mendeteksi delapan jenis emosi dasar seperti marah, takut, sedih, dan bahagia, meskipun akurasi spesifik tidak disebutkan secara eksplisit. Kendati demikian, masih diperlukan optimasi arsitektur dan data pelatihan agar sistem ANN mampu mengenali emosi secara lebih presisi dan kontekstual.

Berangkat dari latar belakang tersebut, maka permasalahan yang mendasari penelitian ini adalah bagaimana peran dan efektivitas *Artificial Neural Network* dalam mendeteksi emosi berbasis teks, serta model jaringan mana yang paling sesuai untuk mengklasifikasikan emosi yang kompleks dan kontekstual. Selain itu, penting untuk dikaji bagaimana integrasi antara ANN dan pendekatan NLP dapat meningkatkan akurasi dan sensitivitas dalam mendeteksi berbagai jenis emosi seperti marah, sedih, takut, atau bahagia. Kompleksitas semantik dalam bahasa alami dan keterbatasan *dataset* yang tersedia menjadi tantangan penting yang perlu ditangani dalam perancangan sistem deteksi emosi berbasis ANN. Penelitian ini juga meninjau kekuatan dan kelemahan dari berbagai arsitektur ANN agar diperoleh pemahaman yang utuh terhadap performa model-model tersebut dalam skenario nyata.

Penelitian ini bertujuan untuk membangun dan mengevaluasi model klasifikasi emosi berbasis teks menggunakan *Artificial Neural Network* dengan arsitektur *embedding*, *ReLU*, dan *softmax* yang efisien dan akurat untuk enam kategori emosi. Penelitian ini terinspirasi dari berbagai studi sebelumnya yang menunjukkan keberhasilan ANN dalam mendeteksi emosi, baik dari teks, *audio*, maupun gabungan data *multimodal* (Bharti et al., 2022; Chutia & Baruah, 2024; Kumar S. & Geetha, 2024). Target akurasi dalam penelitian ini diharapkan dapat menyamai atau melampaui studi sebelumnya yang mencapai 80–90%, dengan struktur jaringan yang lebih sederhana namun tetap andal. Dengan pendekatan ini, diharapkan model yang diusulkan mampu memberikan kontribusi nyata dalam meningkatkan kemampuan sistem untuk memahami dan merespons emosi manusia melalui bahasa tulis.

STUDI LITERATUR

1. Pengertian Dasar

Artificial Neural Network (ANN) atau jaringan syaraf tiruan adalah sistem komputasi yang terinspirasi oleh sistem saraf biologis pada makhluk hidup, khususnya otak manusia. ANN dikembangkan untuk mengatasi keterbatasan sistem komputasi tradisional dalam menangani masalah non-linear dan kompleks. ANN belajar dari data dan dapat melakukan generalisasi dari pola yang dipelajari untuk menghasilkan prediksi atau klasifikasi baru (Thorat et al., 2022).

Perkembangan ANN dimulai sejak tahun 1943 oleh McCulloch dan Pitts yang memodelkan neuron sederhana menggunakan sirkuit elektronik (P, 2014). Sejak saat itu, ANN mengalami perkembangan

signifikan, terutama sejak munculnya *algoritma backpropagation* dan konsep deep learning pada dekade terakhir (Salah Alaloul & Hannan Qureshi, 2020).

2. Konsep Dasar dan Arsitektur ANN

ANN terdiri dari tiga komponen utama: input layer, hidden layer, dan output layer. Tiap lapisan terdiri dari neuron buatan yang menerima input, memproses informasi menggunakan fungsi aktivasi, dan mengirimkan output ke neuron berikutnya.

Matematis, keluaran suatu neuron dinyatakan sebagai:

$$y = \left(\sum_{i=1}^n w_i x_i + b \right)$$

Dimana :

- x_i : input ke-i dari data yang masuk ke neuron
- w_i : weight yang menyatakan kekuatan koneksi terhadap input ke-i
- b : bias, yaitu konstanta yang memungkinkan pergeseran fungsi aktivasi
- $(\sum_{i=1}^n w_i x_i + b)$: sebagai nilai aktivasi sebelum fungsi aktivasi diterapkan
- f : Fungsi aktivasi yang mengubah nilai aktivasi menjadi output akhir dari neuron
- y : hasil keluaran dari neuron setelah fungsi aktivasi diaplikasikan

Fungsi aktivasi yang umum digunakan antara lain:

- Sigmoid : $\frac{1}{1+e^{-z}}$, menghasilkan output antara 0 dan 1.
- ReLU : $f(z) = \max(0, z)$, cepat dan efisien untuk data non-linier
- Tanh : $f(z) = \tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$, output antara -1 dan 1
- Softmax : Digunakan pada output layer klasifikasi multikelas, menghasilkan distribusi probabilistik

ANN dibangun dengan arsitektur berlapis, dan ketika jumlah hidden layer lebih dari satu, jaringan tersebut dikategorikan sebagai deep neural network (DNN) (Salah Alaloul & Hannan Qureshi, 2020).

3. Proses Pembelajaran dan Optimisasi

ANN belajar melalui proses penyesuaian bobot berdasarkan data pelatihan. Proses ini dikenal sebagai *training* dan memanfaatkan algoritma optimisasi seperti *backpropagation* dan *gradient descent*.

Tahapan utama dalam pembelajaran ANN meliputi:

1. Inisialisasi bobot
2. Propagasi input ke output
3. Perhitungan error menggunakan *loss function*
4. *Backpropagation* untuk menghitung gradien
5. Pembaruan bobot menggunakan metode optimisasi

Terdapat tiga jenis pembelajaran dalam ANN:

- *Supervised Learning*
Supervised learning adalah metode pembelajaran di mana model dilatih menggunakan data yang sudah diberi label. Artinya, input dan output yang benar sudah diketahui, sehingga jaringan belajar untuk memetakan input ke output tersebut. Contoh: klasifikasi emosi dari teks berdasarkan data yang telah diberi label seperti "senang", "sedih", atau "marah".
- *Unsupervised Learning*
Unsupervised learning digunakan ketika data tidak memiliki label, dan tujuan utama adalah menemukan pola atau struktur tersembunyi dalam data. ANN dengan pendekatan ini biasanya

digunakan untuk clustering atau reduksi dimensi. Contoh: mengelompokkan dokumen berdasarkan kemiripan emosional tanpa mengetahui jenis emosinya terlebih dahulu.

- **Reinforcement Learning**

Reinforcement learning adalah metode pembelajaran di mana model belajar melalui *trial and error*, dengan mendapatkan umpan balik berupa *reward* atau *punishment*. Jaringan saraf bertindak sebagai agen yang berinteraksi dengan lingkungan dan belajar dari konsekuensi tindakannya. Contoh: agen AI yang belajar bagaimana merespons emosi pengguna dalam chatbot secara optimal berdasarkan *respons* pengguna.

4. Keunggulan dan Keterbatasan ANN Keunggulan ANN meliputi:

- Kemampuan untuk belajar dari data dan menyesuaikan diri (*adaptive learning*)
- Kemampuan untuk melakukan generalisasi terhadap data baru
- Kemampuan mengolah data secara paralel dan *real-time*
- Toleransi terhadap kesalahan atau *noise* dalam data

Namun, ANN juga memiliki beberapa keterbatasan:

- Membutuhkan data pelatihan dalam jumlah besar
- Risiko *overfitting*
- Interpretasi model yang sulit (*black-box*)
- Waktu komputasi yang tinggi untuk jaringan besar (Mr. Dongare Sadanand & Dr. Sachin Bhosale, 2023).

5. Penerapan ANN dalam Dunia Nyata ANN telah digunakan secara luas dalam berbagai aplikasi, seperti:

- Pengenalan citra dan suara
- Deteksi dan klasifikasi objek
- Sistem rekomendasi
- Prediksi pasar keuangan
- Analisis data medis dan diagnosis penyakit
- Pemrosesan bahasa alami (NLP)

Penerapan ANN berkembang pesat terutama dalam teknologi deep learning, seperti *convolutional neural networks* (CNN) untuk klasifikasi gambar, dan *recurrent neural networks* (RNN) untuk pemrosesan sekuensial (Neapolitan & Jiang, 2018).

PROSEDUR KERJA

Penelitian ini menggunakan pendekatan klasifikasi teks berbasis *Artificial Neural Network* (ANN) untuk mendeteksi emosi dalam kalimat. Prosedur kerja penelitian terdiri atas beberapa tahapan, yaitu:

1. Pengambilan dan Pemahaman Data

Dataset diunduh dari repositori Kaggle dengan nama *emotions-dataset*. Dataset ini berisi pasangan data berupa teks pendek (kalimat) dan label emosi dalam bentuk angka, yang mewakili enam jenis emosi: *sadness*, *joy*, *love*, *anger*, *fear*, dan *surprise*. Label numerik kemudian dikonversi ke bentuk label teks untuk mempermudah interpretasi.

2. Pra-pemrosesan Teks

Sebelum diproses oleh model, dilakukan beberapa tahapan pra-pemrosesan terhadap data teks:

- Tokenisasi dilakukan menggunakan *Tokenizer* dari pustaka Keras, yang mengubah kalimat menjadi urutan indeks kata. Hanya 10.000 kata terpopuler yang dipertahankan, sementara kata asing diganti dengan token <OOV>.
- Padding dilakukan untuk menyamakan panjang setiap input teks menjadi 100 kata, dengan menambahkan nol di akhir jika panjangnya kurang.
- *Encoding Label* dilakukan menggunakan *LabelEncoder* dari *Scikit-learn*, mengubah label emosi menjadi bentuk numerik 0 hingga 5 sesuai kebutuhan model klasifikasi.

3. Pembagian Dataset

Data dibagi menjadi dua bagian, yaitu data latih sebanyak 80% dan data uji sebanyak 20% menggunakan *train_test_split*. Tujuannya adalah untuk melatih model dan mengevaluasi performa terhadap data yang belum pernah dilihat sebelumnya.

4. Pembangunan Model ANN

Model dibangun menggunakan arsitektur *Sequential* dari Keras, dengan struktur sebagai berikut:

- *Embedding Layer*: Mengonversi indeks kata menjadi vektor berdimensi 64.
- *GlobalAveragePooling1D*: Merangkum representasi kata dalam kalimat menjadi satu vektor tetap.
- *Hidden Layers*: Dua layer fully-connected dengan 64 dan 32 neuron, menggunakan fungsi aktivasi ReLU.
- *Dropout* sebesar 30% diterapkan untuk mencegah overfitting.
- *Output Layer*: Menggunakan fungsi aktivasi softmax untuk menghasilkan probabilitas dari enam kelas emosi.

5. Kompilasi dan Pelatihan Model

Model dikompilasi dengan fungsi *loss sparse_categorical_crossentropy* dan optimizer adam. Pelatihan dilakukan selama maksimal 20 *epoch* dengan *batch size* sebesar 32. *Callback EarlyStopping* digunakan untuk menghentikan pelatihan lebih awal jika validasi loss tidak membaik setelah 3 *epoch*, guna mencegah *overfitting*.

6. Evaluasi Model

Setelah pelatihan, model dievaluasi terhadap data uji menggunakan metrik akurasi dan *classification report (precision, recall, f1-score)* untuk setiap kelas. Visualisasi grafik akurasi dan loss juga dibuat untuk melihat kestabilan model selama pelatihan.

7. Prediksi Emosi dan Pengujian Model

Model diuji dengan beberapa kalimat dari data uji untuk melihat apakah prediksi emosi sesuai dengan label aslinya. Selain itu, dibuat pula fungsi prediksi mandiri yang dapat digunakan untuk mengklasifikasikan emosi dari input teks baru.

Algoritma: Klasifikasi Emosi pada Teks dengan ANN

Input: Dataset emosi yang berisi pasangan data berupa teks (kalimat) dan label emosi dalam bentuk numerik

Output: Model ANN terlatih dan prediksi label emosi untuk teks baru

Langkah-langkah:

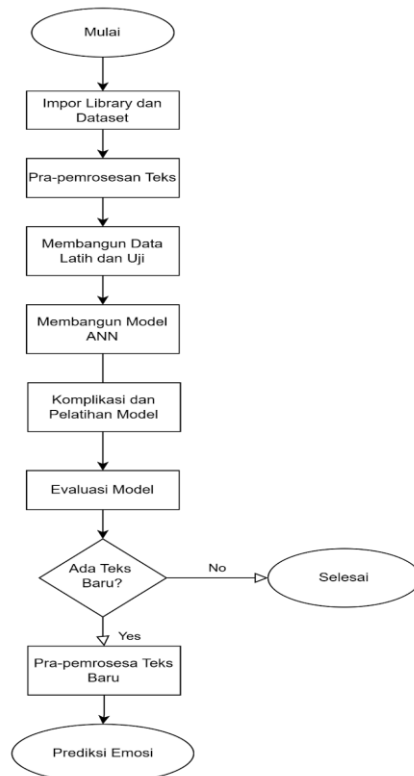
1. *Import Library* dan Dataset
 - 1.1. Unduh dataset dari Kaggle
 - 1.2. Baca file CSV ke dalam DataFrame
2. Pra-pemrosesan Data
 - 2.1. Tokenisasi teks menggunakan Tokenizer (dengan *num_words=10000*, *oov_token='<OOV>'*)
 - 2.2. Ubah teks menjadi urutan angka berdasarkan kamus token
 - 2.3. Lakukan padding untuk menyamakan panjang urutan menjadi 100
 - 2.4. Ubah label emosi ke bentuk numerik menggunakan *LabelEncoder*
3. Split Data
 - 3.1. Bagi data menjadi data latih (80%) dan data uji (20%) menggunakan *train_test_split*
4. Bangun Model ANN
 - 4.1. Inisialisasi model *Sequential*
 - 4.2. Tambahkan layer *Embedding* (*input_dim=10000*, *output_dim=64*, *input_length=100*)
 - 4.3. Tambahkan *GlobalAveragePooling1D*
 - 4.4. Tambahkan *Dense layer* (64 neuron, ReLU)
 - 4.5. Tambahkan *Dropout* (*rate=0.3*)

- 4.6. Tambahkan *Dense layer* (32 neuron, ReLU)
- 4.7. Tambahkan *output layer* (jumlah neuron = jumlah kelas, aktivasi softmax)
5. Kompilasi dan Latih Model
 - 5.1. Kompilasi model dengan:
 - *Loss: sparse_categorical_crossentropy*
 - *Optimizer: adam*
 - *Metrik: accuracy*
 - 5.2. Tambahkan *callback EarlyStopping*
 - 5.3. Latih model selama 20 *epoch*, dengan *batch size* 32 dan data validasi
6. Evaluasi Model
 - 6.1. Hitung akurasi dan loss pada data uji
 - 6.2. Tampilkan *classification report (precision, recall, f1-score)*
 - 6.3. Buat grafik akurasi dan loss selama pelatihan
7. Prediksi Kalimat Baru
 - 7.1. Buat fungsi input teks pengguna
 - 7.2. Tokenisasi dan *padding teks input*
 - 7.3. Gunakan model untuk memprediksi emosi
 - 7.4. Kembalikan label hasil prediksi

Diagram alir (*flowchart*) dari prosedur

Penjelasan Tiap Langkah *Flowchart*

1. Mulai
Proses dimulai.
2. *Impor Library* dan Dataset
Mengimpor pustaka Python seperti *TensorFlow, pandas, dan scikit-learn*, lalu membaca dataset emosi.
3. Pra-pemrosesan Teks
Melakukan *tokenisasi, padding, dan encoding label* agar teks bisa diproses oleh model.
4. Membangun Data Latih dan Uji
Dataset dibagi menjadi data latih dan data uji, biasanya dengan rasio 80:20.
5. Membangun Model ANN
Menyusun arsitektur jaringan saraf tiruan (ANN) yang terdiri dari *embedding layer, pooling, dense layer, dll.*
6. Kompilasi dan Pelatihan Model
Model dikompilasi dengan fungsi *loss dan optimizer*, lalu dilatih menggunakan data latih selama beberapa *epoch*.
7. Evaluasi Model
Model dievaluasi menggunakan data uji untuk mengukur akurasi, *precision, recall, f1-score*, dan visualisasi performa.
8. Apakah Ada Teks Baru?
Digunakan untuk memutuskan apakah sistem akan menerima input teks baru dari pengguna.
 - Jika Tidak: Proses selesai.
 - Jika Ya: Lanjut ke pra-pemrosesan teks baru.
9. Pra-pemrosesan Teks Baru
Teks input dari pengguna diproses seperti pada tahap awal (*tokenisasi dan padding*).
10. Prediksi Emosi
Model digunakan untuk memprediksi emosi dari teks baru berdasarkan pelatihan yang telah dilakukan.



Gambar 1. Flowchart Prosedur Kerja

HASIL DAN DISKUSI

4.1 Hasil Penelitian

Penelitian ini bertujuan untuk mengklasifikasikan emosi berdasarkan teks menggunakan *metode Artificial Neural Network* (ANN). Dataset yang digunakan merupakan dataset emosi dari Kaggle dengan total 388.809 data, yang telah melalui tahap pra-pemrosesan seperti *tokenisasi*, *padding*, dan *encoding label*.

4.1.1 Distribusi Label

Setelah dilakukan pemetaan label numerik ke dalam bentuk teks, distribusi data tiap kelas adalah sebagai berikut:

Tabel 1. Distribusi Label

EMOSI	JUMLAH DATA
Joy	141.067
Sadness	121.187
Anger	57.317
Fear	47.712
Love	34.554
Surprise	14.972

Data ini menunjukkan bahwa label "*joy*" dan "*sadness*" mendominasi, sedangkan "*surprise*" memiliki jumlah paling sedikit, yang mungkin berdampak pada performa model dalam mengenali emosi tersebut.

4.1.2 Arsitektur Model ANN

Model ANN dibangun menggunakan layer berikut:

- *Embedding*: 10000 kata unik, *output dimension* 64.
- *GlobalAveragePooling1D*: Untuk merata-ratakan hasil *embedding*.
- *Dense Layer* 1: 64 neuron, ReLU activation.
- *Dropout*: 30% untuk mencegah *overfitting*.
- *Dense Layer* 2: 32 neuron, ReLU activation.

- *Output Layer*: 6 neuron (jumlah kelas emosi), *softmax activation*.

Model menggunakan *loss function sparse_categorical_crossentropy* dan *optimizer adam*.

Total parameter yang dilatih adalah 646.438.

4.1.3 Proses Pelatihan

Model dilatih selama maksimal 20 *epoch* dengan *early stopping (patience=3)*. Berikut ringkasan hasil pelatihan:

Tabel 2. Proses Pelatihan

Epoch	Train Acc	Val Acc	Train Loss	Val Loss
1	76.41%	86.85%	0.6317	0.3935
4	90.55%	90.11%	0.2001	0.2063
7	90.92%	90.29%	0.1810	0.1875
10	91.30%	90.57%	0.1674	0.2074

Model berhenti setelah *epoch* ke-10 dengan akurasi uji sebesar 90,29%, menunjukkan performa yang sangat baik dalam klasifikasi emosi berbasis teks.

4.1.4 Laporan Klasifikasi

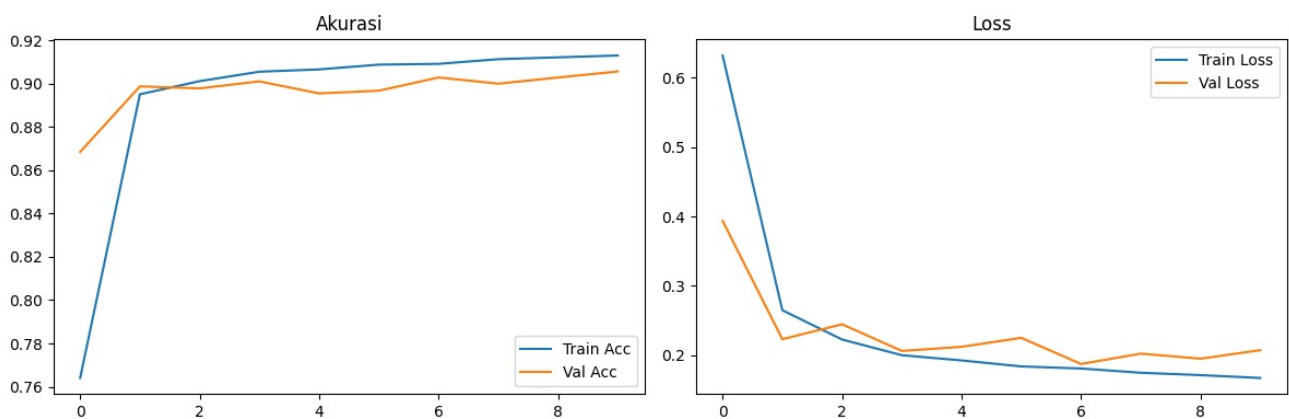
Berikut adalah metrik evaluasi dari data uji sebanyak 83.362 data:

Tabel 3. Laporan Klasifikasi

Emosi	Precision	Recall	F1-Score	Support
Sadness	0.92	0.96	0.94	24.201
Joy	0.92	0.93	0.93	28.164
Love	0.82	0.78	0.80	6.929
Anger	0.94	0.86	0.90	11.441
Fear	0.91	0.80	0.85	9.594
Surprise	0.69	0.95	0.80	3.033
Accuracy				90%
			90%	83.362

Nilai precision dan recall yang tinggi pada kelas "joy" dan "sadness" menunjukkan bahwa model mengenali emosi umum dengan baik. Sementara itu, nilai precision rendah pada kelas "surprise" mengindikasikan kemungkinan ketidakseimbangan data atau kemiripan semantik antar kelas.

4.1.5 Grafik Pelatihan



Gambar 2. Plot Akurasi dan Loss

1. Grafik Kiri: Akurasi

- Sumbu X: *Epoch* (jumlah iterasi pelatihan model)
- Sumbu Y: Nilai akurasi (skala 0–1)
- Garis Biru ("*Train Acc*"): Akurasi data pelatihan
- Garis Oranye ("*Val Acc*"): Akurasi data validasi

Interpretasi:

- Akurasi pelatihan meningkat dengan cukup konsisten dari sekitar 0.76 ke 0.915.
- Akurasi validasi juga meningkat dari sekitar 0.87 ke 0.905, tetapi lebih fluktuatif.
- Tidak ada tanda *overfitting* yang signifikan: akurasi validasi tidak menurun drastis, dan cukup dekat dengan akurasi pelatihan.

2. Grafik Kanan: Loss

- Sumbu X: *Epoch*
- Sumbu Y: Nilai loss (kesalahan prediksi)
- Garis Biru ("*Train Loss*") : Nilai loss pada data pelatihan
- Garis Oranye ("*Val Loss*") : Nilai loss pada data validasi

Interpretasi:

- *Train loss* menurun tajam dari awal (sekitar 0.63 ke 0.16), menunjukkan model belajar dengan baik.
- *Val loss* juga menurun, meskipun *fluktuatif* di tengah *epoch* (sekitar 0.39 ke 0.20).
- Fluktuasi kecil pada *val loss* bisa disebabkan oleh *noise* atau kompleksitas data validasi.
- Tidak ada tanda *overfitting* parah: loss validasi tetap rendah, tidak naik drastis

Kesimpulan Umum:

- Model mengalami proses pembelajaran yang baik.
- Tidak terlihat *overfitting* yang signifikan.
- Model sudah cukup stabil dan siap diuji lebih lanjut atau digunakan untuk prediksi jika diperlukan.

4.1.6 Contoh Prediksi Model

Berikut adalah 5 dari 10 contoh hasil prediksi teks:

Tabel 4. Representasi Prediksi Modal terhadap Kalimat Uji pada Klasifikasi Emosi

Teks	Label Asli	Prediksi
"i hate that i can still feel if any nerve is damaged..."	Sadness	Sadness
"i feel respected for what i'm doing..."	joy	joy
"i left without a real explanation..."	love	love
"i must admit i didn't feel like hugging him..."	anger	anger
"i have a really nice feeling about this one..."	surprise	surprise

Model berhasil memprediksi dengan akurat pada sebagian besar data uji. Kalimat ambigu atau penuh OOV (*out of vocabulary*) tetap bisa diklasifikasi dengan cukup tepat berkat embedding dan pooling.

4.1.7 Contoh Prediksi Model dengan Input Baru

Tabel 5. Representasi Prediksi Modal dengan Input Baru

Teks	Prediksi
"I cried when I got the championship"	joy
"I'm scared of the dark"	fear
"Im feeling a bit amazed and grateful about having landed amid such a congregation"	surprise
"I haven't done my eye make up I hate anyone seeing me and I feel nekkid that sounds really vain but I just think I look horrendous without it"	sadness
"You don't pay attention to me"	anger
"I loved you more than I ever said out loud"	love

4.2 Diskusi

Model ANN berhasil mencapai akurasi uji sebesar 90,29%, yang merupakan hasil sangat baik untuk klasifikasi multi-label berbasis teks.

Beberapa poin penting dalam diskusi:

- Kelas dominan seperti "*joy*" dan "*sadness*" memberikan kontribusi besar terhadap akurasi keseluruhan.

- Model dapat mengenali emosi intens seperti *anger* dan *fear* dengan baik, berkat distribusi data dan arsitektur yang efisien.
- Kelas underrepresented seperti *surprise* menunjukkan tantangan tersendiri dengan precision rendah meski recall tinggi. Hal ini dapat ditingkatkan dengan teknik augmentasi data atau metode penyeimbangan data seperti SMOTE.
- Fungsi prediksi individual juga telah berhasil dikembangkan, memungkinkan klasifikasi emosi untuk teks baru secara real-time.
- Kehadiran token <OOV> dalam teks menunjukkan adanya kata di luar kamus tokenisasi, namun tidak terlalu memengaruhi prediksi model secara signifikan karena rata-rata akurasi tetap tinggi.

KESIMPULAN

5.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa:

1. *Artificial Neural Network* (ANN) terbukti efektif dalam mengklasifikasikan emosi berdasarkan data teks. Dengan arsitektur sederhana namun optimal, ANN mampu mengenali berbagai ekspresi emosional dalam kalimat.
2. Model ANN yang dibangun dalam penelitian ini berhasil mencapai akurasi sebesar 90,29% pada data uji, yang menunjukkan performa yang sangat baik dalam tugas klasifikasi multi-label.
3. Emosi yang umum seperti *joy* dan *sadness* berhasil diklasifikasikan dengan akurasi tinggi. Namun, model mengalami tantangan dalam mengenali emosi dengan frekuensi rendah seperti *surprise*, yang menunjukkan perlunya penanganan ketidakseimbangan data.
4. Fungsi prediksi teks baru berhasil dikembangkan dan menunjukkan kemampuan model dalam mengenali emosi dari input baru secara real-time, yang menandakan bahwa sistem ini dapat diimplementasikan dalam aplikasi dunia nyata.
5. Meskipun performa model sangat baik, tantangan seperti token <OOV> (kata asing) dan variasi konteks tetap menjadi perhatian dalam pengembangan sistem klasifikasi emosi berbasis teks ke depannya.

5.2 Saran

Untuk pengembangan selanjutnya, beberapa saran yang dapat diberikan antara lain:

- Menggunakan pendekatan penyeimbangan data seperti SMOTE untuk meningkatkan performa model terhadap kelas minoritas.
- Mengintegrasikan ANN dengan teknik Natural Language Processing (NLP) lanjutan seperti word embedding berbasis BERT atau LSTM untuk menangkap konteks lebih dalam.
- Melakukan augmentasi data untuk memperluas representasi emosi dan memperbaiki klasifikasi pada teks-teks ambigu.
- Menerapkan model dalam sistem interaktif seperti chatbot, layanan konseling, atau analisis opini publik sebagai aplikasi dunia nyata.

5.3 Ucapan Terima Kasih

Penulis mengucapkan terima kasih kepada:

- Bapak Lenox Larwuy, M.Si. sebagai dosen pengampu mata kuliah Komputasi Matematika, yang telah memberikan arahan dan bimbingan selama proses penyusunan tugas ini.
- Seluruh rekan mahasiswa yang turut berdiskusi dan berbagi ide sehingga penulis mendapatkan berbagai wawasan baru dalam pengolahan data berbasis pembelajaran mesin.
- Tim pengembang Kaggle yang telah menyediakan dataset *emotions-dataset* yang sangat berguna dalam eksperimen ini.
- Dan semua pihak yang tidak dapat disebutkan satu per satu yang telah memberikan dukungan secara langsung maupun tidak langsung selama proses penelitian ini berlangsung.

Semoga hasil dari penelitian ini dapat memberikan manfaat dalam pengembangan sistem cerdas yang lebih manusiawi dan responsif terhadap emosi manusia.

DAFTAR PUSTAKA

- Bharti, S. K., Varadhaganapathy, S., Gupta, R. K., Shukla, P. K., Bouye, M., Hingaa, S. K., & Mahmoud, A. (2022). Text-Based Emotion Recognition Using Deep Learning Approach. *Computational Intelligence and Neuroscience*, 2022. <https://doi.org/10.1155/2022/2645381>
- Chutia, T., & Baruah, N. (2024). A review on emotion detection by using deep learning techniques. In *Artificial Intelligence Review* (Vol. 57, Issue 8). Springer Netherlands. <https://doi.org/10.1007/s10462-024-10831-1>
- Kanger, N., & Bathla, G. (2017). Recognizing Emotion in Text using Neural Network and Fuzzy Logic. *Indian Journal of Science and Technology*, 10(12), 1–6. <https://doi.org/10.17485/ijst/2017/v10i12/100526>
- Kumar S., S. A., & Geetha, A. (2024). Emotion Detection from Text using Natural Language Processing and Neural Networks. *International Journal of Intelligent Systems and Applications in Engineering*, 12(14s), 609–615.
- Martinez-Navarro, J. A., Rubio-Espino, E., Sossa-Azuela, J. H., Ponce-Ponce, V. H., Molina-Lozano, H., & Garcia-Sebastian, L. M. (2023). Comparison of Neural Networks for Emotion Detection. *Computacion y Sistemas*, 27(3), 653–665. <https://doi.org/10.13053/CyS-27-3-4515>
- Mr. Dongare Sadanand, & Dr. Sachin Bhosale. (2023). Basic of Artificial Neural Network. *International Journal of Advanced Research in Science, Communication and Technology*, 3(3), 299–303. <https://doi.org/10.48175/ijarsct-8159>
- Neapolitan, R. E., & Jiang, X. (2018). Neural Networks and Deep Learning. *Artificial Intelligence*, 389–411. <https://doi.org/10.1201/b22400-15>
- P, M. S. and W. (2014). Research Paper on Basic of Artificial Neural Network. *International Journal on Recent and Innovation Trends in Computing and Communication*, 2(1), 96–100.
- Salah Alaloul, W., & Hannan Qureshi, A. (2020). Data Processing Using Artificial Neural Networks. *Dynamic Data Assimilation - Beating the Uncertainties, DI*, 1–26. <https://doi.org/10.5772/intechopen.91935>
- Thorat, M., Pandit, S., & Balote, S. (2022). Artificial Neural Network: A brief study. *Asian Journal of Convergence in Technology*, 8(3), 12–16. <https://doi.org/10.33130/ajct.2022v08i03.003>