LAPORAN FINAL PROJECT TEXT MINING

PENGEMBANGAN CHATBOT PERTOLONGAN PERTAMA KESEHATAN MENTAL BERBASIS KLASIFIKASI INTENT MENGGUNAKAN FINE-TUNING INDOBERT

TAMARA JOVANKA

NRP 5025221213

ATHAYA ROHADATUL YAQUTAH

NRP 5025221235

ALMIRA RAISA IZZATINA

NRP 5025221250

Dosen Pembimbing

Dini Adni Navastara, S.Kom, M.Sc.

NIP 198510172015042001

Ratih Nur Esti Anggraini, S.Kom., M.Sc., Ph.D.

NIP 198412102014042003

Dr. Diana Purwitasari, S.Kom., M.Sc.

NIP 197804102003122001

Program Studi Teknik Informatika

Departemen Teknik Informatika

Fakultas Teknologi Elektro dan Informatika Cerdas

Institut Teknologi Sepuluh Nopember

Surabaya

Tahun 2025

LAPORAN FINAL PROJECT TEXT MINING

PENGEMBANGAN CHATBOT PERTOLONGAN PERTAMA KESEHATAN MENTAL BERBASIS KLASIFIKASI INTENT MENGGUNAKAN FINE-TUNING INDOBERT

TAMARA JOVANKA

NRP 5025221213

ATHAYA ROHADATUL YAQUTAH

NRP 5025221235

ALMIRA RAISA IZZATINA

NRP 5025221250

Dosen Pembimbing

Dini Adni Navastara, S.Kom, M.Sc.

NIP 198510172015042001

Ratih Nur Esti Anggraini, S.Kom., M.Sc., Ph.D.

NIP 198412102014042003

Dr. Diana Purwitasari, S.Kom., M.Sc.

NIP 197804102003122001

Program Studi Teknik Informatika

Departemen Teknik Informatika

Fakultas Teknologi Elektro dan Informatika Cerdas

Institut Teknologi Sepuluh Nopember

Surabaya

Tahun 2025

TEXT MINING FINAL PROJECT REPORT

DEVELOPMENT OF A MENTAL HEALTH FIRST AID CHATBOT BASED ON INTENT CLASSIFICATION USING FINE-TUNING INDOBERT

TAMARA JOVANKA

NRP 5025221213

ATHAYA ROHADATUL YAQUTAH

NRP 5025221235

ALMIRA RAISA IZZATINA

NRP 5025221250

Advisors

Dini Adni Navastara, S.Kom, M.Sc.

NIP 198510172015042001

Ratih Nur Esti Anggraini, S.Kom., M.Sc., Ph.D.

NIP 198412102014042003

Dr. Diana Purwitasari, S.Kom., M.Sc.

NIP 197804102003122001

Study Program Informatics Engineering

Department of Informatics Engineering

Faculty of Intelligent Electrical and Informatics Technology

Institut Teknologi Sepuluh Nopember

Surabaya

Year 2025

DAFTAR ISI

ABSTRAK i
ABSTRACT ii
DAFTAR ISI iii
DAFTAR GAMBAR iv
DAFTAR TABEL v
DAFTAR SIMBOL vi
BAB 1 PENDAHULUAN 7
1.1 Latar Belakang 7
1.2 Rumusan Masalah 7
1.3 Tujuan dan Manfaat 7
BAB 2 TINJAUAN PUSTAKA 8
2.1 Dasar Teori 8
BAB 3 METODOLOGI 9
3.1 Metode yang digunakan 9
3.2 Urutan pelaksanaan penelitian 9
BAB 4 Hasil dan Pembahasan 10
4.1 Hasil penelitian 10
4.2 Pembahasan 10
DAFTAR PUSTAKA 11
LAMPIRAN 12

ABSTRAK

PENGEMBANGAN CHATBOT PERTOLONGAN PERTAMA KESEHATAN MENTAL BERBASIS KLASIFIKASI INTENT MENGGUNAKAN FINE-TUNING INDOBERT

Nama Mahasiswa / NRP	**:**	**Tamara Jovanka / 5025221213** **Athaya Rohadatul / 5025221235** **Almira Raisa Izzatina / 5025221250**
Departemen	**:**	**Teknik Informatika**
Dosen Pembimbing	**:**	**Dini Adni Navastara, S.Kom, M.Sc.** **Ratih Nur Esti Anggraini, S.Kom., M.Sc.** **Prof. Dr. Diana Purwitasari, S.Kom., M.Sc.**

Abstrak

Chatbot telah menjadi solusi inovatif dalam mendukung layanan berbasis percakapan, termasuk di bidang kesehatan mental. Penelitian ini bertujuan untuk merancang dan membangun chatbot berbahasa Indonesia yang mampu mengklasifikasikan intent pengguna secara akurat menggunakan pendekatan text mining berbasis model LLM, khususnya IndoBERT. Dataset yang digunakan merupakan data percakapan kesehatan mental yang diterjemahkan ke dalam bahasa Indonesia dan diproses melalui tahapan text preprocessing seperti stemming, stopword removal, dan lowercasing. Dua model dibandingkan: IndoBERT dan CENDOL. Hasil evaluasi menunjukkan bahwa IndoBERT unggul dengan peningkatan akurasi validasi dari 65,20% menjadi 85,35% setelah fine-tuning, dengan F1-score mencapai 84,59%. Sistem chatbot TanyaRasa kemudian diimplementasikan dalam aplikasi web interaktif menggunakan Streamlit dan di-deploy melalui Hugging Face. Hasil ini menunjukkan bahwa penggunaan IndoBERT dengan strategi fine-tuning yang tepat dapat secara efektif meningkatkan pemahaman chatbot terhadap intent dalam konteks bahasa Indonesia, serta mendukung layanan pertolongan pertama psikologis yang lebih responsif.

Kata kunci: ***Chatbot, IndoBERT, Klasifikasi Intent, Kesehatan Mental, Fine-tuning, Text Mining***

ABSTRACT

DEVELOPMENT OF A MENTAL HEALTH FIRST AID CHATBOT BASED ON INTENT CLASSIFICATION USING FINE-TUNING INDOBERT

Student Name / NRP	**:**	**Tamara Jovanka / 5025221213** **Athaya Rohadatul / 5025221235** **Almira Raisa Izzatina / 5025221250**
Department	**:**	**Teknik Informatika**
Advisor	**:**	**Dini Adni Navastara, S.Kom, M.Sc.** **Ratih Nur Esti Anggraini, S.Kom., M.Sc.** **Prof. Dr. Diana Purwitasari, S.Kom., M.Sc.**

Abstract

Chatbots have become an innovative solution in supporting conversation-based services, including in the field of mental health. This research aims to design and develop an Indonesian-language chatbot capable of accurately classifying user intent using a text mining approach based on LLM models, specifically IndoBERT. The dataset used consists of mental health conversation data translated into Indonesian and processed through text preprocessing stages such as stemming, stopword removal, and lowercasing. Two models were compared: IndoBERT and CENDOL. Evaluation results show that IndoBERT excels with validation accuracy improvement from 65.20% to 85.35% after fine-tuning, achieving an F1-score of 84.59%. The TanyaRasa chatbot system was subsequently implemented in an interactive web application using Streamlit and deployed through Hugging Face. These results demonstrate that the use of IndoBERT with appropriate fine-tuning strategies can effectively enhance chatbot understanding of intent in the Indonesian language context, while supporting more responsive psychological first aid services.

Keywords: ***Chatbot, IndoBERT, Intent Classification, Mental Health, Fine-tuning, Text Mining***

DAFTAR ISI

ABSTRACT 6
DAFTAR ISI 7
DAFTAR GAMBAR 8
DAFTAR TABEL 9
BAB 1 PENDAHULUAN 10
1.1 Latar Belakang 10
1.2 Rumusan Masalah 10
1.3 Tujuan dan Manfaat 11
BAB 2 TINJAUAN PUSTAKA 12
2.1 Dasar Penelitian 12
2.2 Dasar Teori 15
2.2.1 Text Pre-processing 15
2.2.2 Transformer Model 16
2.2.3 IndoBERT 17
2.2.4 Cendol 18
2.2.6 Finetuning 21
2.2.7 BLEU 22
2.2.8 ROUGE 22
2.2.9 Dataset 23
2.2.10 Chatbots dalam Aplikasi Kesehatan Mental 24
BAB 3 METODOLOGI 25
3.1 Metode yang digunakan 25
3.2 Implementasi Sistem 28
3.2.1 Pra-pemrosesan Data dan Analisis Eksplorasi 28
3.2.2 Pra-pemrosesan Teks Bahasa Indonesia 29
3.2.3 Persiapan Model dan Tokenisasi 30
3.2.4 Pelatihan dan Evaluasi Model 30
3.2.5 Implementasi Sistem Prediksi dan Chatbot 31
BAB 4 Hasil dan Pembahasan 33
4.1 Konfigurasi Environment 33
4.2 Hasil Penelitian 33
4.2.1 Perbandingan Performa Model Berdasarkan Jumlah Epoch 33
4.2.2 Analisis Pemilihan Model Optimal 34
4.2.3 Fine-Tuning IndoBERT 35
4.2.4 Evaluasi Kualitas Respons Teks 36
4.3 Hasil Deployment 37
BAB 5 KESIMPULAN 38

DAFTAR GAMBAR

Gambar 3.1 Flowchart Sistem……22

Gambar 4.1 Tampilan Aplikasi……...34

DAFTAR TABEL

Tabel 2.1. Perbandingan Penelitian Terkait……12

Tabel 4.1. Perbandingan Kinerja Model Berdasarkan Epoch 12……30

Tabel 4.2. Perbandingan Kinerja Model Berdasarkan Epoch 30…….31

Tabel 4.3. Perbandingan Kinerja Model Berdasarkan Batch Size dan Learning Rate……32

BAB 1 PENDAHULUAN

1.1 Latar Belakang

Text mining merupakan teknik untuk mengekstraksi informasi dan pola dari kumpulan data teks dalam skala besar. Melalui proses ini, sistem komputer dapat belajar memahami struktur kalimat, mengenali makna kata, serta mengidentifikasi maksud atau intent di balik pernyataan pengguna. Kemampuan ini menjadi semakin penting di era digital saat ini, terutama dalam pengembangan sistem interaktif berbasis bahasa alami.

Salah satu penerapan utama dari text mining adalah pada pengembangan chatbot, yaitu sistem yang dirancang untuk merespons masukan pengguna secara otomatis melalui teks atau suara. Keakuratan respons chatbot sangat bergantung pada sejauh mana sistem tersebut dapat memahami maksud pengguna. Oleh karena itu, metodologi dalam text mining berperan krusial dalam membangun chatbot yang tidak hanya reaktif, tetapi juga adaptif dan kontekstual.

Di Indonesia, kebutuhan akan chatbot berbahasa Indonesia terus meningkat, terutama di sektor layanan pelanggan, edukasi, dan pelayanan publik. Namun, sebagian besar model chatbot yang ada masih bergantung pada model bahasa yang dilatih menggunakan data bahasa Inggris. Hal ini menyebabkan keterbatasan dalam memahami konteks dan gaya bahasa Indonesia yang khas, sehingga berdampak pada akurasi klasifikasi intent dan kualitas respons yang diberikan.

Untuk mengatasi tantangan tersebut, model Large Language Model (LLM) lokal yang dikembangkan secara khusus untuk memahami bahasa Indonesia. Model ini memiliki potensi besar dalam meningkatkan pemahaman chatbot terhadap intent pengguna dalam konteks lokal. Dengan menggabungkan pendekatan text mining dan kekuatan representasi bahasa dari model-model tersebut, chatbot dapat belajar membedakan berbagai jenis intent pengguna secara lebih akurat dan relevan.

Proyek ini bertujuan untuk mengeksplorasi proses pembelajaran klasifikasi intent pada chatbot berbahasa Indonesia, dengan memanfaatkan model-model LLM Bahasa Indonesia sebagai fondasi dan pendekatan text mining sebagai metode. Fokus utama bukan hanya pada implementasi teknis, tetapi juga pada pemahaman mendalam terhadap bagaimana sistem dapat belajar dari data teks untuk meningkatkan kualitas interaksi berbasis bahasa alami.

1.2 Rumusan Masalah

Berdasarkan latar belakang di atas, rumusan masalah dalam proyek ini adalah:

1. Bagaimana merancang dan membangun chatbot berbahasa Indonesia yang mampu mempelajari dan mengklasifikasikan intent pengguna secara akurat menggunakan pendekatan text mining berbasis model LLM berbahasa Indonesia?
2. Sejauh mana efektivitas proses pembelajaran klasifikasi intent menggunakan model-model tersebut dapat meningkatkan pemahaman chatbot terhadap bahasa alami dalam konteks bahasa Indonesia?

1.3 Tujuan dan Manfaat

Berdasarkan rumusan masalah di atas, tujuan dan manfaat dalam proyek ini adalah:

1. Mengetahui cara merancang dan membangun chatbot berbahasa Indonesia yang mampu mempelajari dan mengklasifikasikan intent pengguna secara akurat menggunakan pendekatan text mining berbasis model LLM berbahasa Indonesia.
2. Mengetahui seberapa jauh efektivitas proses pembelajaran klasifikasi intent yang menggunakan model-model tersebut dapat meningkatkan pemahaman chatbot terhadap bahasa alami dalam konteks bahasa Indonesia.

BAB 2 TINJAUAN PUSTAKA

2.1 Dasar Penelitian

Penelitian yang ada telah menunjukkan berbagai pendekatan dalam membangun chatbot, khususnya dalam penggunaan intent classification. Dwiyono et al. (2024) mengimplementasikan tiga arsitektur transformer state-of-the-art untuk perbandingan performa klasifikasi intent. Arsitektur BERT menggunakan bidirectional encoder dengan 12 layers, 768 hidden units, dan 12 attention heads yang memungkinkan pemahaman konteks dari kedua arah secara simultan. Model RoBERTa merupakan optimisasi dari BERT dengan menghilangkan next sentence prediction task dan menggunakan dynamic masking strategy selama training, serta dilatih dengan dataset yang lebih besar dan batch size yang lebih optimal. IndoBERT mengadopsi arsitektur BERT namun khusus dilatih dengan corpus bahasa Indonesia yang ekstensif untuk memahami struktur linguistik dan konteks budaya Indonesia. Metodologi penelitian dimulai dengan tahap preprocessing data. Proses fine-tuning dilakukan dengan menyesuaikan hyperparameter seperti learning rate, batch size, dan jumlah epoch optimal untuk setiap model. Training process menggunakan cross-entropy loss function dengan Adam optimizer untuk optimasi parameter model. Evaluasi dilakukan menggunakan metrics komprehensif meliputi akurasi, F1-score, precision, dan recall pada dataset test yang terpisah. Hasil menunjukkan IndoBERT mencapai akurasi 94%, BERT 89%, dan RoBERTa 84%, dimana keunggulan IndoBERT disebabkan oleh language-specific training yang lebih relevan dengan konteks bahasa Indonesia.

Ouaddi et al. (2025) melakukan perbandingan multi-model antara GPT2 sebagai generative transformer, BERT sebagai encoder-only model, LLaMA sebagai decoder-only architecture, dan RoBERTa sebagai optimized encoder. Setiap model diadaptasi secara khusus untuk domain pariwisata melalui fine-tuning dengan dataset yang dilabeli berdasarkan kriteria "Six A" yang mencakup Attraction, Accessibility, Amenities, Available packages, Activities, dan Accommodation. Metodologi penelitian dimulai dengan tahap dataset preparation yang melibatkan labeling manual. Model architecture setup dilakukan dengan konfigurasi layer classification head yang optimal untuk setiap arsitektur LLM. Training pipeline menggunakan pendekatan transfer learning dengan fine-tuning pada domain-specific data, dimana pre-trained weights dari general domain diadaptasi untuk memahami konteks pariwisata. Comparative analysis dilakukan dengan evaluasi performa lintas model menggunakan metrics standar seperti akurasi, precision, recall, dan F1-score untuk menentukan model yang paling efektif dalam mengklasifikasikan intent terkait pariwisata. Pada tahap testing F1-score dikeluarkan, BERT mencapai 98%, GPT2 mencapai 99%, RobERTa mencapai 99%, dan LLaMA mencapai 84%.

Kuchlous & Kadaba (2020) mengembangkan arsitektur yang dioptimalkan khusus untuk menangani tantangan teks pendek dalam konteks conversational agents. Arsitektur model dirancang dengan specialized short text processing modules yang dapat mengekstrak informasi maksimal dari input dengan karakter terbatas. Model ini diadaptasi untuk domain mental health melalui fine-tuning dengan data dari aplikasi Wysa, dimana vocabulary dan pattern komunikasi disesuaikan dengan konteks kesehatan mental dan wellness. Metodologi penelitian dimulai dengan tahap short text preprocessing dan feature engineering. Model training mengimplementasikan berbagai algoritma machine learning dengan optimasi khusus untuk handling text brevity, termasuk attention mechanisms yang dapat fokus pada

token-token penting dalam teks pendek. Hasil akhir terbaik dicapai dengan teknik Bigrams, dengan Multinomial Naive Bayes mencapai 89% akurasi, Logistic Regression mencapai 91% akurasi, Support Vector Machine mencapai 91% akurasi, dan Random Forest mencapai 93% akurasi.

Kuligowska & Kowalczuk (2024) mengimplementasikan framework active learning yang menggunakan feedback loop untuk continuous improvement model. Pipeline dirancang dengan sistem iterative yang dapat secara otomatis mengidentifikasi sampel data yang paling informatif untuk proses labeling. Dengan model Sentence-Transformer, metodologi penelitian dimulai dengan initial model training menggunakan dataset baseline yang tersedia untuk membentuk model awal. Uncertainty sampling dilakukan dengan mengidentifikasi sampel-sampel yang memiliki confidence score rendah atau prediction uncertainty tinggi, yang kemudian diprioritaskan untuk proses labeling manual. Human-in-the-loop labeling melibatkan expert annotators untuk memberikan label yang akurat pada sampel-sampel yang tidak pasti. Model retraining dilakukan secara berkala dengan menggabungkan data baru yang telah dilabeli ke dalam training set, menggunakan incremental learning atau full retraining tergantung pada volume data baru. Iterative refinement process dijalankan secara kontinyu hingga mencapai performance threshold yang diinginkan, dengan monitoring metrics seperti akurasi dan F1-score untuk mengevaluasi improvement model. Hasil terbaik memberikan akurasi ~79% dan Macro Average F1-score ~77% pada 26.600 dataset.

Souha et al. (2023) mengimplementasikan framework evaluasi komprehensif untuk membandingkan berbagai arsitektur pre-trained models dalam tugas intent classification. Arsitektur yang dievaluasi mencakup BERT dengan encoder-only structure, GPT dengan decoder-only architecture dan berbagai varian optimisasi seperti RoBERTa, DistilBERT, dan ALBERT. Setiap model diadaptasi menggunakan transfer learning pipeline yang konsisten untuk memastikan fair comparison. Metodologi penelitian dimulai dengan comprehensive model survey yang menganalisis karakteristik arsitektur setiap pre-trained model, termasuk jumlah parameter, training data, dan optimization techniques yang digunakan. Standardized fine-tuning protocol diimplementasikan dengan hyperparameter yang konsisten untuk semua model, termasuk learning rate, batch size, dan training epochs. Comparative evaluation dilakukan menggunakan dataset dan metrics yang identik untuk semua model, dengan fokus pada metrics seperti akurasi, F1-score, training time, dan inference speed. Critical analysis dilakukan untuk mengidentifikasi kelebihan dan keterbatasan setiap model dalam konteks intent classification, termasuk analisis trade-off antara performa dan computational efficiency. Hasil terbaik pada tahap testing diraih oleh BERT dengan F1-score 90.5% dengan learning rate 1e-5, ALBERT dengan F1-score 91.7% dengan learning rate 1e-5, RoBERTa dengan F1-score 91.3% dengan learning rate 1e-6, dan GPT dengan F1-score 90.1% dengan learning rate 1e-5.

Judul (Penulis, Tahun)	**Dataset**	**Metode & Tahapan**	**Model**	**Analisis Gap**
Analisis Perbandingan Klasifikasi Intent Chatbot Menggunakan Deep Learning BERT, RoBERTa, dan IndoBERT (Dwiyono et al., 2024)	Dataset chatbot universitas dalam bahasa Indonesia.	BERT, RoBERTa, dan IndoBERT ditrain dengan dataset untuk klasifikasi intent dan dievaluasi performance-nya.	**89%** (BERT), **84%** (RoBERTa), **93%** (IndoBERT).	Penelitian ini masih terbatas pada domain universitas, belum mengeksplorasi model LLM terbaru, dan perlu dievaluasi menggunakan dataset yang lebih beragam.
Assessing the effectiveness of large language models for intent detection in tourism chatbots: A comparative analysis and performance evaluation (Ouaddi et al., 2025)	Dataset spesifik pariwisata dengan labeling "Six A" criteria (attractions, accessibility, amenities, activities, available packages, and ancillary services).	Fine-tuning GPT, BERT, LLaMA, RoBERTa dengan analisis komparatif performa	**98%** (BERT), **99%** (GPT2), **99%** (RoBERTa), **84%** (LLaMA).	Fokus terbatas pada domain pariwisata dan terdapat kekurangan eksplorasi teknik ensemble.
Short Text Intent Classification for Conversational Agents (Kuchlous & Kadaba, 2020)	Dataset dengan 403 entri dari aplikasi Wysa (mental wellness chatbot)	Preprocessing, feature engineering, dan evaluasi performa.	**89%** (Multinomial Naive Bayes), **91%** (Logistic Regression), **91%** (SVM). **93%** (Random Forest)	Kurang eksplorasi teknik text augmentation

Enhancing Chatbot Intent Classification using Active Learning Pipeline for Optimized Data Preparation (Kuligowska & Kowalczuk, 2024)	Dataset commercial (visitor-bot) yang mengandung 689.832 pesan unik	TF-IDF, Analisis N-grams, Peningkatan akurasi dengan active learning	**~79%** (Sentence Transformer)	Tidak ada perbandingan dengan baseline, kurang detail dalam pemaparan arsitektur model
Pre-Trained Models for Intent Classification in Chatbot: Comparative Study and Critical Analysis (Souha et al., 2023)	Dataset single-domain dengan 77 intent dan 13.083 request customer service.	Fine-tuning model, analisis komparatif, dan critical analysis performa.	**91.7%** (ALBERT), **90.5%** (BERT), **91.3%** (RoBERTa)	Perlu eksplorasi model generasi terbaru

Tabel 2.1 Perbandingan Penelitian Terkait

2.2 Dasar Teori

2.2.1 Text Pre-processing

Preprocessing teks merupakan tahap fundamental dalam pemrosesan bahasa alami yang bertanggung jawab dalam membersihkan dan standardisasi data teks sebelum dimasukkan ke dalam proses pelatihan model pembelajaran mesin. Keputusan yang dibuat selama fase ini memiliki dampak yang signifikan terhadap validitas makna yang diperoleh, kekuatan statistik dari analisis berikutnya, dan kemampuan untuk secara akurat menangkap makna maupun gaya bahasa. Proses ini esensial karena secara langsung memengaruhi kualitas data yang akan dianalisis, dengan tujuan utama untuk mengurangi dimensionalitas data, meningkatkan kekuatan prediktif, dan menstandarisasi teks untuk konsistensi (Hickman et al., 2022).

2.2.1.1 Stemming

Stemming adalah sebuah teknik preprocessing teks yang bertujuan untuk mengurangi dimensionalitas data dengan menggunakan algoritma heuristik untuk "menghapus imbuhan morfologis dari kata, menyisakan hanya batang kata (word stem)". Proses ini tidak selalu menghasilkan bentuk dasar kata yang valid secara leksikal, sehingga batang kata yang dihasilkan seringkali sulit untuk diinterpretasikan. Sebagai contoh, stemmer Porter yang populer akan mengubah variasi kata seperti organ, organs, organic, organism, organize, dan organization menjadi satu batang kata yang sama, yaitu "organ".

Tujuan utama dari stemming adalah untuk meningkatkan kekuatan statistik dengan menggabungkan kata-kata yang secara semantik terkait menjadi satu unit tunggal, yang sangat bermanfaat ketika bekerja dengan korpus data berukuran kecil yang memiliki kekuatan statistik terbatas untuk membedakan variasi kata yang halus. Meskipun demikian, perlu kehati-hatian dalam penerapannya, karena proses ini berisiko menyatukan kata-kata yang memiliki makna berbeda, yang pada akhirnya dapat mengurangi validitas analisis (Vajjala et al., 2020).

2.2.1.2 Stopword Removal

Stopword removal adalah proses menghilangkan kata-kata yang sangat umum muncul dalam sebuah korpus sehingga dianggap tidak memberikan nilai informatif untuk analisis atau pencarian dokumen. Kata-kata ini, seperti "sebuah", "di", dan "dan", dapat dihapus menggunakan daftar stop words generik maupun daftar yang dibuat secara spesifik untuk domain tertentu. Awalnya, teknik ini banyak digunakan dalam sistem pencarian informasi (information retrieval) untuk mengurangi waktu komputasi (Jurafsky & Martin, 2023).

2.2.1.3 Lowercasing

Lowercasing atau konversi huruf kecil adalah teknik preprocessing teks yang mengubah semua huruf dalam sebuah korpus menjadi huruf kecil. Alasan utama di balik proses ini adalah karena sistem komputer merepresentasikan huruf kapital dan huruf kecil secara berbeda; misalnya, kata "Kerja" dan "kerja" akan dihitung sebagai dua unit yang terpisah jika tidak diseragamkan. Dengan melakukan lowercasing, semua variasi kapitalisasi dari sebuah kata akan dianggap setara, yang secara efektif mengurangi dimensionalitas data (Vajjala et al., 2020).

2.2.2 Transformer Model

Model Transformer adalah arsitektur deep learning yang diperkenalkan oleh Vaswani et al. (2017) dan telah menjadi standar de facto dalam berbagai tugas NLP, menggantikan arsitektur sekuensial seperti Recurrent Neural Networks (RNNs). Inovasi utamanya adalah mekanisme self-attention, yang memungkinkan model untuk menimbang pentingnya kata-kata yang berbeda dalam sebuah urutan secara paralel, tanpa bergantung pada jarak antar kata.

Arsitektur Transformer terdiri dari encoder dan decoder. Encoder memetakan urutan input $(x_1, \ldots, x_n)$ ke representasi kontinu $z = (z_1, \ldots, z_n)$. Decoder kemudian menghasilkan urutan output $(y_1, \ldots, y_m)$ dari z. Komponen kunci dari Transformer adalah Scaled Dot-Product Attention, yang dihitung menggunakan rumus berikut.

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V$$

dimana Q adalah matriks Query, K adalah matriks Key, V adalah matriks Value, dan d_k adalah dimensi dari Key. $\sqrt{d_k}$ berfungsi sebagai faktor penskalaan untuk mencegah argumen fungsi softmax menjadi terlalu besar, yang dapat menyebabkan gradien yang sangat kecil saat proses pelatihan. Transformer juga mengimplementasikan Multi-Head Attention, yang

memungkinkan model untuk bersama-sama memperhatikan informasi dari subspace representasi yang berbeda pada posisi yang berbeda (Vaswani et al., 2017; Wang et al., 2022).

2.2.3 IndoBERT

IndoBERT adalah model bahasa representasi dua arah (*bidirectional*) yang didasarkan pada arsitektur BERT (*Bidirectional Encoder Representations from Transformers*) . BERT sendiri hanya menggunakan tumpukan encoder dari arsitektur Transformer. Model ini dilatih menggunakan lebih dari 220 juta kata yang dikumpulkan dari tiga sumber utama, yaitu Wikipedia Indonesia, news articles, dan web crawl data. Proses pelatihan dilakukan selama 2.4 juta steps dengan 180 epochs, menghasilkan perplexity 3.97 pada development set yang sebanding dengan BERT-base English.

IndoBERT secara spesifik dilatih (pre-trained) menggunakan korpus bahasa Indonesia yang besar dan beragam, sehingga mampu memahami konteks dan nuansa linguistik bahasa Indonesia secara mendalam.Tujuan dari proses pre-training adalah untuk meminimalkan fungsi kerugian gabungan (combined loss function) dari kedua tugas tersebut dan mengajarkan pemahaman bahasa Indonesia yang mendalam kepada model sebelum disesuaikan (fine-tuned) untuk tugas-tugas spesifik (seperti analisis sentimen atau NER).

Proses pre-training IndoBERT melibatkan dua tugas utama untuk membangun pemahaman konteks bahasa Indonesia. Tugas pertama adalah Masked Language Model (MLM), di mana sekitar 15% dari token input disembunyikan secara acak, dan model ditugaskan untuk memprediksi token asli berdasarkan konteks kata-kata di sekitarnya. Tugas ini ditujukan untuk meminimalkan loss function, yang biasanya cross-entropy loss dari prediksi. Rumus untuk fungsi kerugian MLM dapat direpresentasikan sebagai berikut.

$$L_{MLM}(\theta) = -\sum_{i \in M} logP(t_i | T_{masked}; \theta)$$

dimana θ adalah parameter dari model Transformer, T_{masked} adalah urutan token input dengan beberapa token yang telah disembunyikan, Madalah himpunan indeks dari token-token yang disembunyikan, t_i adalah token asli pada indeks ke-i, dan $P(t_i | T_{masked}; \theta)$ adalah probabilitas yang diprediksi oleh model untuk token asli t_i pada posisi yang disembunyikan, berdasarkan konteks dari token-token yang tidak disembunyikan.

Tugas kedua adalah Next Sentence Prediction (NSP), yang memberikan model dua kalimat (A dan B) dan mengharuskan model untuk menentukan apakah kalimat B merupakan kelanjutan yang logis dari kalimat A dalam korpus data aslinya (Wilie et al., 2020).

IndoBERT mengadopsi arsitektur dasar dari BERT-Base, yang hanya menggunakan tumpukan encoder dari arsitektur Transformer asli. Setiap lapisan encoder terdiri dari dua sub-lapisan utama, yaitu Multi-Head Self-Attention dan Position-wise Feed-Forward Network. Mekanisme Self-Attention memungkinkan model untuk menilai pentingnya setiap kata dalam kalimat saat merepresentasikan kata tersebut, sehingga model dapat memahami konteks secara lebih baik. "Multi-head" di sini merujuk pada pelaksanaan mekanisme attention secara paralel sebanyak 12 kali (dalam kasus ini), di mana setiap head secara independen belajar untuk memfokuskan perhatian pada hubungan yang berbeda antar kata dalam suatu kalimat. Dengan demikian, penggunaan multi-head attention memungkinkan

model untuk secara simultan menangkap berbagai jenis hubungan sintaksis dan semantik, sehingga memperkaya representasi konteks kata dalam kalimat. Setelah proses attention, representasi token diproses melalui feed-forward network, yaitu jaringan saraf sederhana yang diterapkan secara independen pada setiap token untuk mentransformasi representasi yang telah diperkaya dengan konteks. Kedua sub-lapisan ini dilengkapi dengan koneksi residual dan layer normalization untuk menjaga stabilitas dan efektivitas pelatihan model yang dalam Koto et al. (2020).

2.2.4 Cendol

Model Cendol merupakan Large Language Model (LLM) yang dirancang khusus untuk memahami dan menghasilkan teks dalam bahasa Indonesia serta sejumlah bahasa daerah di Nusantara. Secara arsitektural, model ini mengintegrasikan dua arsitektur transformer utama yaitu decoder-only dan encoder-decoder dengan rentang ukuran parameter yang bervariasi dari 300 juta hingga 13 miliar parameter, dimana arsitektur decoder-only menggunakan backbone LLaMA-2 untuk tugas generasi teks unidireksional, sementara arsitektur encoder-decoder memanfaatkan backbone mT5 untuk tugas yang memerlukan pemahaman dan generasi teks secara simultan. Secara matematis, fungsi objektif decoder-only pada Cendol dapat direpresentasikan sebagai berikut.

$$L(\theta) = -\sum_{i=1}^{N} logP(y_i|x_{1:i-1}, \theta)$$

dimana θ adalah parameter model (bobot yang dipelajari saat pelatihan), x adalah input, y adalah input, N adalah panjang dari output (jumlah kata yang diprediksi), $logP$ adalah logaritma dari probabilitas prediksi model, dan Σ adalah penjumlahan total kesalahan tiap kata. Pada fungsi berikut, model menghitung kesalahan dalam memprediksi kata-kata target y_1, ..., y_n satu persatu, berdasarkan kata-kata sebelumnya x_1 sampai x_{i-1}. Sementara untuk encoder-decoder model, encoder pertama akan membaca seluruh input, kemudian decoder memproduksi jawaban satu per satu, menggunakan informasi dari encoder dan kata-kata yang telah diprediksi sebelumnya. Fungsi objektif encoder-decoder dapat ditulis sebagai berikut.

$$L(\theta) = -\sum_{i=1}^{N} logP(y_i|x_{1:m}, y_{1:i-1}, \theta)$$

dimana $x_{1:m}$ adalah input lengkap, dan $y_{1:i-1}$ adalah kata-kata yang sudah diprediksi sebeleumnya.

Mekanisme kerja Cendol didasarkan pada proses instruction tuning yang melibatkan adaptasi kosakata (vocabulary adaptation) sebagai alternatif dari parameter-efficient tuning, yang menurut Cahyawijaya et al. (2024) memiliki keterbatasan dalam adaptasi bahasa. Proses ini dimulai dengan pre-training pada corpus bahasa Indonesia yang luas, dilanjutkan dengan fine-tuning menggunakan dataset instruksi yang telah dikurasi khusus untuk konteks linguistik dan kultural Indonesia, kemudian dioptimalkan melalui supervised fine-tuning (SFT) dengan formulasi loss function sebagai berikut.

$$L_{SFT} = \sum_{i=1}^{N} - log\ P(y_o|x, y_{1:i-1}, \theta_{SFT})$$

dimana θ_{SFT} adalah parameter yang telah diatur.

Evaluasi komprehensif menunjukkan bahwa Cendol mengungguli open-source multilingual dan region-specific LLMs pada sebagian besar benchmark dengan margin yang signifikan, mencapai peningkatan performa hingga 20% dan mendemonstrasikan kemampuan generalisasi yang baik pada tugas-tugas yang belum pernah dilihat sebelumnya serta bahasa-bahasa indigenous Indonesia (Cahyawijaya et al., 2024).

2.2.5 Metric Evaluation

Dalam ekosistem pembelajaran mesin modern, penilaian kinerja model tidak hanya terbatas pada tahap eksperimental tetapi juga mencakup pemantauan berkelanjutan dalam tahap produksi. Kemampuan untuk mengukur kinerja secara teruji adalah dasar dalam membangun sistem yang terpercaya dan bertanggung jawab. Metrik evaluasi berfungsi sebagai jembatan antara kapabilitas teoretis model dan dampak praktisnya. Pemilihan metrik yang tidak tepat dapat mengarah pada kesimpulan yang keliru tentang nilai sebuah model, terutama dalam skenario dunia nyata yang sering kali melibatkan data tidak seimbang dan biaya kesalahan yang asimetris.

2.2.5.1 Accuracy

Akurasi adalah metrik yang paling intuitif, mengukur proporsi total prediksi yang benar dari keseluruhan data. Metrik ini mengukur kemampuan model secara keseluruhan dalam membedakan semua kelas.

$$Accuracy = \frac{True\ Positive + True\ Negative}{True\ Positive + True\ Negative + False\ Positive + False\ Negative}$$

Rumus 1. Rumus Akurasi

Meskipun sederhana, akurasi dapat memberikan gambaran yang menyesatkan pada dataset dengan distribusi kelas yang tidak seimbang (imbalanced dataset). Sebagai contoh, pada dataset deteksi penipuan dengan 99% transaksi non-penipuan, model yang selalu memprediksi "non-penipuan" akan mencapai akurasi 99%, meskipun gagal total dalam mengidentifikasi kasus penipuan (Géron, 2022).

2.2.5.2 Presisi (Precision)

Presisi merupakan metrik evaluasi yang mengukur proporsi instance yang diprediksi sebagai positif dan benar-benar termasuk dalam kelas positif. Metrik ini sangat penting dalam situasi di mana kesalahan prediksi positif (False Positive) memiliki dampak yang lebih besar dibandingkan kesalahan prediksi negatif (False Negative).

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

Sebagai contoh, dalam sistem rekomendasi konten yang aman untuk anak, salah melabeli konten dewasa sebagai aman (FP) memiliki konsekuensi serius, sehingga presisi menjadi metrik utama (Géron, 2022).

2.2.5.3 Recall

Recall merupakan metrik evaluasi yang mengukur proporsi instance yang benar-benar termasuk dalam kelas positif dan berhasil dikenali dengan benar oleh model. Metrik ini

menjadi sangat krusial dalam situasi di mana kesalahan klasifikasi negatif (False Negative) membawa konsekuensi serius.

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

Contoh klasik penerapan recall terdapat pada sistem diagnosis medis, di mana kegagalan mendeteksi penyakit pada pasien yang sebenarnya sakit (False Negative) dapat berakibat fatal. Sebaliknya, kesalahan dalam mendiagnosis pasien sehat sebagai sakit (False Positive) umumnya memiliki dampak yang lebih ringan. Oleh karena itu, dalam konteks seperti ini, recall menjadi metrik utama yang digunakan untuk mengevaluasi kinerja model secara lebih tepat (Géron, 2022).

2.2.5.4 F1-Score

F1-Score adalah rata-rata harmonik (harmonic mean), yaitu adalah jenis rata-rata yang dihitung dengan membagi jumlah data dengan jumlah kebalikan (invers) dari masing-masing nilai data, dari Presisi dan Recall. Metrik ini berupaya menyeimbangkan kontribusi kedua metrik tersebut. F1-Score menjadi metrik pilihan ketika kepentingan antara Presisi dan Recall seimbang. Penggunaan rata-rata harmonik memberikan bobot lebih pada nilai yang lebih rendah, sehingga F1-Score akan tinggi hanya jika kedua metrik (Presisi dan Recall) juga tinggi (Géron, 2022).

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} = \frac{2 \times True\ Positive}{(2 \times True\ Positive) + False\ Positive + False\ Negative}$$

2.2.5.5 Perplexity

Perplexity merupakan metrik standar yang digunakan untuk mengevaluasi performa model bahasa. Secara intuitif, perplexity mengukur tingkat ketidakpastian atau "kebingungan" model ketika memprediksi kata-kata dalam data uji. Semakin rendah nilai perplexity, semakin baik kemampuan model dalam memprediksi urutan kata, karena menunjukkan bahwa model memberikan probabilitas yang tinggi terhadap kata-kata yang benar.

Perplexity didefinisikan sebagai eksponensial dari entropi silang (*cross-entropy*) rata-rata. Untuk sebuah sekuens data uji $W = (W_1, W_2, \ldots, W_n)$, perplexity dihitung sebagai berikut.

$$Perplexity\ (W) = \left(\prod_{i=1}^{N} \frac{1}{P(w_i, \ldots, w_{i-1})}\right)^{\frac{1}{n}}$$

atau dapat ditulis sebagai berikut.

$$Perplexity\ (W) = exp\left(-\frac{1}{N}\sum_{i=1}^{N} log\ p(w_i | w_1, \ldots, w_{i-1})\right)$$

di mana N adalah jumlah total token dalam data uji, dan $p(w_i | w_1, \ldots, w_{i-1})$ adalah probabilitas kondisional yang diberikan model untuk token w_i setelah melihat histori $w_1, \ldots, w_{i-1}$ (Jurafsky & Martin, 2023).

2.2.6 Finetuning

Finetuning adalah metode transfer learning yang umum digunakan dalam penerapan model bahasa yang telah melalui tahap pre-training. Proses ini dimulai dengan memanfaatkan model yang sudah dilatih sebelumnya pada korpus data berskala besar dan bersifat umum. Selanjutnya, model tersebut dilatih ulang menggunakan data berlabel yang lebih kecil dan spesifik sesuai dengan tugas yang ingin diselesaikan (downstream task). Tujuannya adalah untuk menyesuaikan kembali parameter model, θ, yang berasal dari model awal $\theta_{pre-trained}$, agar dapat meminimalkan fungsi kerugian pada tugas tersebut L_{task}, dengan data tugas D_{task}. Secara matematis, proses ini dirumuskan sebagai berikut (Hu et al., 2021)

$$\theta^{*} = arg\ min\ L_{task}(D_{task}; \theta)$$

Penyesuaian hyperparameter seperti learning rate, batch size, dan jumlah epoch merupakan fase krusial dan standar dalam metodologi fine-tuning model bahasa. Praktik ini berakar dari rekomendasi fundamental yang disajikan dalam studi oleh Baharudin et al. (2023) dan Devlin et al. (2018), di mana mereka menyarankan rentang nilai spesifik yang terbukti efektif secara empiris, seperti batch size 32, learning rate pada orde 2e-5, dan jumlah epoch yang relatif singkat, untuk mencegah catastrophic forgetting.

Sementara itu, studi oleh Kannan et al. (2021) menunjukkan keunggulan utama dari fine-tuning pada model BERT, yaitu kemampuannya untuk mengadaptasi model yang telah pre-trained pada korpus umum ke dalam tugas klasifikasi yang lebih spesifik, seperti analisis sentimen pada data Twitter yang dilakukan pada studi tersebut. Fine-tuning memungkinkan model BERT untuk mencapai performa tinggi tanpa perlu melakukan perubahan signifikan pada arsitektur dasarnya—cukup dengan menambahkan satu lapisan output tambahan. Dalam penelitian ini, model hasil fine-tuning (disebut M-BERT) mampu menghasilkan F1-score rata-rata sebesar 97,63% dan akurasi sebesar 98,91%, secara signifikan mengungguli empat pendekatan machine learning lainnya: Multinomial Naive Bayes, Random Forest, Logistic Regression, dan Multi-SVM. Selain itu, fine-tuning terbukti efektif dalam menangani dataset yang tidak seimbang (unbalanced dataset), dengan peningkatan performa yang signifikan terutama pada kelas emosi dengan jumlah sampel yang lebih sedikit, seperti fear dan anger.

Studi oleh Baharudin et al. (2023) menunjukkan bahwa fine-tuning atau penyesuaian pada model BERT merupakan tahap krusial yang secara signifikan meningkatkan performa model klasifikasi untuk tugas-tugas spesifik. Proses ini terbukti efisien dalam meningkatkan kinerja, terutama ketika dilakukan penyesuaian terhadap hyperparameter yang digunakan oleh optimizer. Penelitian tersebut menekankan bahwa pemilihan hyperparameter, khususnya learning rate, memiliki pengaruh besar terhadap hasil akhir model. Fine-tuning juga memungkinkan model untuk memahami batas-batas konteks dengan lebih baik, yang pada akhirnya berdampak pada peningkatan akurasi klasifikasi. Hal ini dibuktikan dalam penelitian mereka, di mana model IndoBERT yang telah di-fine-tuning dengan learning rate 2E-5 berhasil mencapai akurasi validasi 97%, F1-Score 97%, Recall 97%, dan Presisi 98%. Penggunaan fine-tuning pada arsitektur BERT sendiri merupakan praktik yang umum dalam berbagai penelitian untuk meningkatkan kinerja model pada berbagai kasus, seperti klasifikasi sentimen, deteksi berita palsu, dan analisis ulasan pelanggan.

2.2.7 BLEU

BLEU (Bilingual Evaluation Understudy) adalah metrik evaluasi yang secara fundamental dikembangkan oleh Papineni et al. (2002) untuk mengukur kualitas teks yang dihasilkan oleh mesin, khususnya dalam tugas terjemahan mesin (machine translation). Prinsip dasar BLEU adalah membandingkan kedekatan teks kandidat (hasil terjemahan mesin) dengan satu atau lebih teks referensi berkualitas tinggi (hasil terjemahan manusia) melalui perhitungan tumpang tindih n-gram. Metrik ini berorientasi pada presisi (precision), yang menghitung seberapa banyak n-gram pada teks kandidat juga muncul pada teks referensi. Meskipun metrik yang lebih baru telah dikembangkan, kajian-kajian modern menunjukkan bahwa BLEU masih sangat relevan karena efisiensinya dan korelasinya yang cukup baik dengan penilaian manusia, terutama pada domain teknis (Lee et al., 2023).

Untuk mencegah skor presisi tinggi yang tidak semestinya pada kalimat pendek, BLEU mengintegrasikan dua mekanisme utama yang diperkenalkan dalam karya aslinya: modified precision dan Brevity Penalty (BP). Modified precision membatasi jumlah hitungan setiap n-gram kandidat sesuai dengan jumlah maksimum kemunculannya di salah satu referensi, sementara BP memberikan penalti jika panjang teks kandidat secara substansial lebih pendek daripada panjang teks referensi. Kombinasi keduanya menghasilkan skor akhir yang didefinisikan dengan rumus berikut.

$$BLEU = BP \times exp(\sum_{n=1}^{N} w_n log p_n)$$

Di mana BP adalah *Brevity Penalty*, dihitung sebagai $BP = \begin{cases} 1, if\ c>r \\ e^{1-r/c}, if\ c \leq r \end{cases}$, dengan c sebagai panjang kandidat dan r sebagai panjang efektif referensi. Simbol p_n merepresentasikan *modified n-gram precision* dan w_n adalah bobot yang umumnya diatur seragam.

2.2.8 ROUGE

ROUGE (Recall-Oriented Understudy for Gisting Evaluation), yang diperkenalkan oleh Lin (2004), adalah serangkaian metrik yang dirancang untuk mengevaluasi kualitas ringkasan teks otomatis. Berbeda dengan BLEU, ROUGE berorientasi pada recall, yang mengukur sejauh mana unit linguistik (seperti n-gram) dari ringkasan referensi (dibuat manusia) berhasil dicakup oleh dihasilkan mesin. Logika dasarnya adalah bahwa ringkasan yang efektif harus mampu menangkap informasi esensial yang terkandung pada referensi. Dalam praktik evaluasi di bidang Natural Language Generation (NLG), ROUGE masih menjadi standar yang banyak digunakan, meskipun berbagai studi, seperti oleh Celikyilmaz et al. (2020), juga menyoroti keterbatasannya dan mendorong penggunaan metrik pelengkap yang lebih memperhatikan aspek semantik.

Varian ROUGE yang paling umum digunakan adalah ROUGE-N, yang menghitung tumpang tindih n-gram (misalnya, ROUGE-1 untuk unigram). Menurut definisi Lin (2004), ROUGE-N dihitung sebagai rasio jumlah *n*-gram yang cocok (*overlap*) dengan jumlah total *n*-gram dalam teks referensi. Rumus ROUGE-N dapat dituliskan sebagai berikut.

$$ROUGE - N = \frac{\sum S \epsilon \{RefSummaries\} \sum gram_{n \epsilon S} Count_{match}(gram_n)}{\sum S \epsilon \{RefSummaries\} \sum gram_{n \epsilon S} Count(gram_n)}$$

Pembilang (numerator) menghitung jumlah n-gram dalam referensi yang juga muncul dalam teks kandidat, sedangkan penyebut (denominator) adalah total semua n-gram dalam referensi.

Selain ROUGE-N, varian lain yang juga populer adalah ROUGE-L, yang menggunakan Longest Common Subsequence (LCS). ROUGE-L menilai kemiripan berdasarkan urutan kata terpanjang yang muncul di kedua teks secara berurutan, meskipun tidak harus berdampingan. Hal ini membuat ROUGE-L lebih fleksibel dalam mengakomodasi perbedaan susunan kata dibanding ROUGE-N.

2.2.9 Dataset

Dalam pengembangan proyek ini, digunakan dua dataset yang bersumber dari platform Kaggle. Kedua dataset ini menyediakan data percakapan dalam bahasa Inggris yang relevan dengan konteks kesehatan mental. Dataset tersebut kemudian diterjemahkan ke dalam bahasa Indonesia untuk melatih model agar dapat memahami dan merespons pengguna dalam konteks lokal.

Dataset pertama adalah "Mental Health Conversational Data" yang tersedia di Kaggle. Dataset ini berisi koleksi percakapan yang dirancang untuk melatih chatbot yang mampu memberikan dukungan emosional. Struktur data ini terdiri dari berbagai macam jenis percakapan, termasuk dialog sehari-hari, pertanyaan yang sering diajukan (FAQ) mengenai kesehatan mental, diskusi terapi klasik, serta nasihat umum bagi individu yang menghadapi kecemasan dan depresi. Setiap bagian percakapan diklasifikasikan ke dalam intent tertentu, yang menjadi dasar bagi chatbot untuk memahami tujuan dari pesan pengguna dan memberikan respons yang relevan dan empatik. Berikut tautan dataset pertama.

https://www.kaggle.com/datasets/elvis23/mental-health-conversational-data

Dataset kedua yang digunakan adalah "Therapist-Patient Conversation Dataset". Dataset ini menyajikan dialog simulasi antara terapis dan pasien. Pola percakapan merepresentasikan sisi pasien, sementara respons merepresentasikan balasan dari terapis. Penggunaan dataset ini bertujuan untuk melatih model agar dapat mengenali pola-pola bahasa yang digunakan oleh individu ketika mendiskusikan masalah kesehatan mental dan memberikan balasan yang meniru gaya percakapan seorang terapis. Berikut tautan dataset kedua.

https://www.kaggle.com/datasets/neelghoshal/therapist-patient-conversation-dataset/

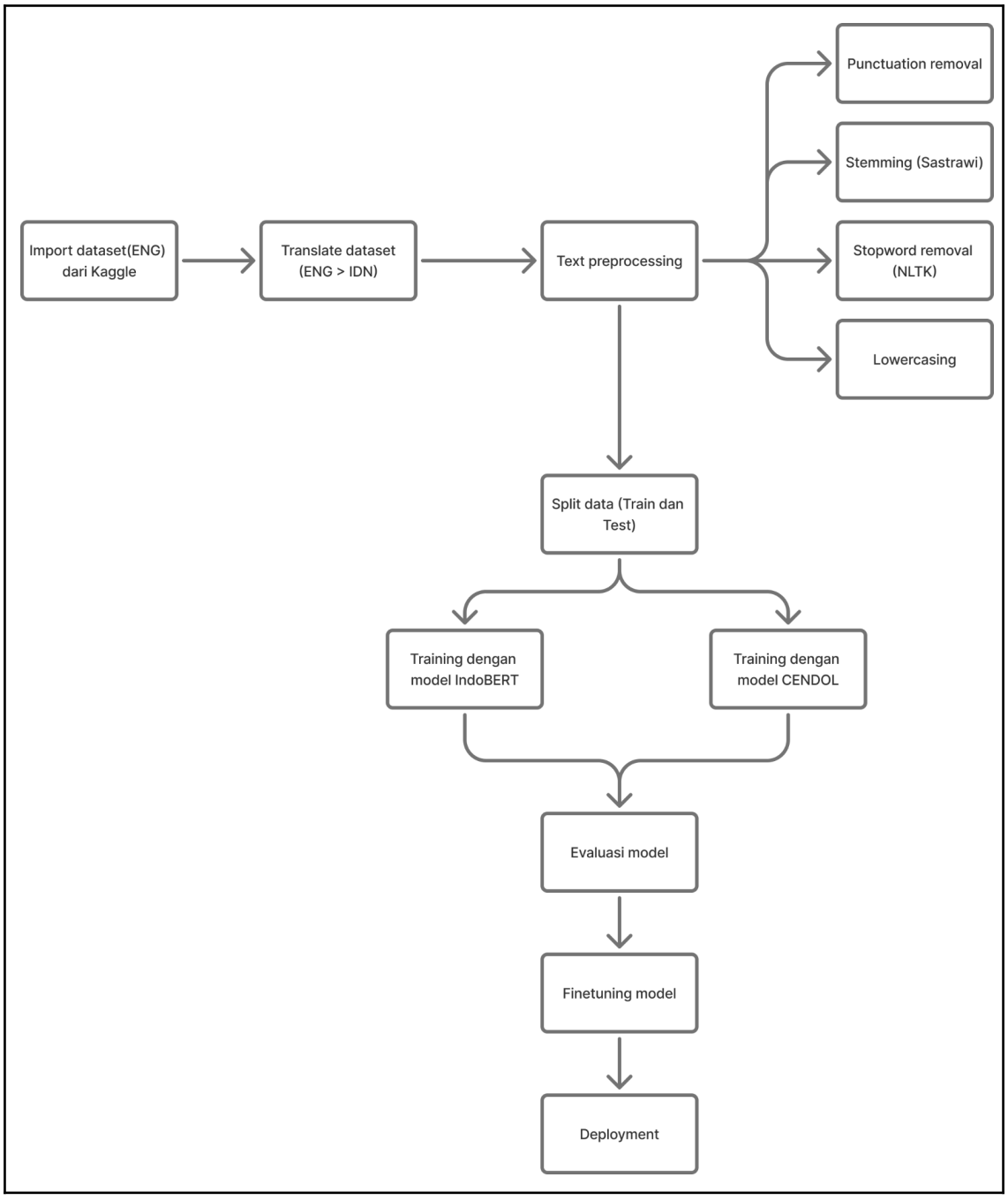
Kedua dataset tersebut kemudian digabungkan dan melalui tahap pra-pemrosesan yang meliputi penerjemahan, pembersihan data seperti penghapusan tanda baca (punctuation removal), stemming, penghapusan stopword, dan lowercasing sebelum digunakan untuk melatih model IndoBERT dan CENDOL.

2.2.10 Chatbots dalam Aplikasi Kesehatan Mental

Penerapan chatbot dalam ranah kesehatan mental telah menunjukkan potensi yang signifikan sebagai alat bantu pertolongan pertama psikologis dan dukungan emosional yang mudah diakses. Chatbot ini dirancang untuk mensimulasikan percakapan dengan seorang terapis. Fungsi utamanya adalah untuk membantu pengguna mengidentifikasi pola pikir negatif, mengelola stres, dan mengembangkan strategi coping yang sehat. Dengan memanfaatkan teknik NLP, chatbot ini mampu mengenali intent pengguna—apakah mereka mengekspresikan kecemasan, depresi, atau stres—dan memberikan respons yang relevan dan empatik. Keunggulan utama dari chatbot kesehatan mental adalah ketersediaannya yang 24/7, anonimitas yang ditawarkan kepada pengguna, dan kemampuannya untuk mengatasi hambatan stigma yang seringkali menghalangi individu untuk mencari bantuan profesional (Abd-Alrazaq et al., 2021).

BAB 3 METODOLOGI

3.1 Metode yang digunakan



Gambar 3.1 Flowchart Sistem.

Berikut penjelasan lebih lanjut untuk tiap tahapan:

a. Import dataset(ENG) dari Kaggle

   Tahap awal menggunakan dataset berbahasa Inggris yang berisi percakapan atau data terkait kesehatan mental. Dataset ini berisi intent-intent umum kesehatan mental seperti anxiety, depression, stress, dll.

b. *Translate* dataset (ENG > IDN)

   Dalam tahap ini kami melakukan proses penerjemahan dataset dari bahasa Inggris ke bahasa Indonesia menggunakan *manual translate* dengan Google Translate. Hal ini dilakukan agar chatbot dapat memahami dan merespons dalam bahasa Indonesia sesuai konteks lokal.

c. Text Pre-processing

   Sebelum data dimasukkan ke dalam *train* model, dilakukan tahap *pre-processing* terlebih dahulu. Tahap ini mencakup beberapa sub-proses:

   1. *Punctuation removal*

      Tahap ini menghilangkan semua tanda baca seperti titik (.), koma (,) tanda tanya (?), tanda seru (!), dan simbol lainnya dari teks. Dalam konteks kesehatan mental, penghapusan tanda baca penting karena model perlu fokus pada kata-kata yang mengandung makna emosional daripada struktur gramatikal.

   2. *Stemming*

      Stemming adalah proses mengubah kata-kata berimbuhan menjadi bentuk dasarnya. Untuk dataset kesehatan mental, stemming sangat penting karena pengguna mungkin mengekspresikan emosi yang sama dengan variasi kata yang berbeda. Proses ini menggunakan algoritma seperti Sastrawi untuk bahasa Indonesia, yang memahami aturan morfologi bahasa Indonesia untuk memotong imbuhan dengan tepat.

   3. *Stopwords removal*

      Stopword adalah kata-kata yang sering muncul tetapi tidak memberikan informasi penting untuk klasifikasi, seperti "dan", "atau", "yang", "adalah", "dengan", dll. Dalam konteks kesehatan mental, penghapusan stopword membantu model fokus pada kata-kata yang mengandung indikator emosional atau psikologis.

   4. *Lowercasing*

      Semua huruf dalam teks diubah menjadi huruf kecil untuk memastikan konsistensi. Hal ini penting karena model machine learning bersifat case-sensitive, sehingga "SEDIH", "Sedih", dan "sedih" akan dianggap sebagai token yang berbeda jika tidak di-lowercase.

d. Split data (Train dan Test)

Dataset yang telah diproses kemudian dibagi menjadi dua bagian: data training dan data testing. Pembagian data ini adalah 80:20, dimana sebagian besar data digunakan untuk melatih model dan sisanya untuk mengevaluasi performa model dalam mengklasifikasikan intent yang belum pernah dilihat sebelumnya.

e. Train Model

- Training dengan model IndoBERT

  Model IndoBERT menggunakan arsitektur indolem/indobert-base-uncased yang dikonfigurasi untuk klasifikasi sequence dengan dropout 0.0 dan maksimum token length 128. Data dipreprocessing menggunakan Sastrawi untuk stemming dan penghapusan stopwords, kemudian dibagi dengan rasio 80:20 untuk training-validation. Training dilakukan dengan batch size 64, optimizer AdamW (learning rate 5e-5), maksimal 30 epoch dengan early stopping patience 3 epoch berdasarkan validation loss terendah. Model melakukan klasifikasi langsung menggunakan AutoModelForSequenceClassification dan dievaluasi dengan accuracy, classification report, confusion matrix, dan perplexity untuk mengukur performa klasifikasi intent kesehatan mental.

- Model CENDOL menggunakan arsitektur indonlp/cendol-mt5-small-inst yang merupakan model generative T5 dengan maksimum input length 128 dan target length 16 token. Data diformat sebagai instruction-following dengan template "Tentukan intent dari kalimat berikut: [text]" dan target berupa nama intent, menggunakan custom dataset class untuk menangani format sequence-to-sequence dengan padding token -100. Training menggunakan batch size 64, optimizer AdamW (learning rate 5e-5), maksimal 30 epoch dengan early stopping patience 3 epoch. Berbeda dengan IndoBERT, CENDOL melakukan text generation menggunakan model.generate() untuk menghasilkan label intent, kemudian dievaluasi dengan accuracy, classification report, confusion matrix, dan perplexity untuk mengukur kualitas generasi intent.

f. Evaluasi model

Tahap evaluasi model dilakukan untuk membandingkan performa IndoBERT dan CENDOL dalam mengklasifikasikan intent kesehatan mental menggunakan validation dataset (10% dari total data). IndoBERT dievaluasi menggunakan klasifikasi langsung dengan menghitung accuracy, precision, recall, F1-score, dan confusion matrix, sedangkan CENDOL dievaluasi melalui text generation dengan exact string matching antara generated text dan reference labels. Kedua model juga dievaluasi menggunakan perplexity

(np.exp(avg_val_loss)) untuk mengukur confidence level prediksi. Hasil evaluasi komprehensif ini dibandingkan secara objektif meliputi accuracy score, F1-score, dan perplexity untuk menentukan model mana yang lebih efektif dalam mengklasifikasikan intent kesehatan mental.

g. Finetuning model

Tahap finalisasi model merupakan proses fine-tuning terhadap model IndoBERT terpilih dengan memuat model pre-trained dari direktori sebelumnya sebagai starting point. Fine-tuning dilakukan dengan hyperparameter yang dioptimalkan: learning rate lebih rendah (2e-5) untuk mencegah catastrophic forgetting, batch size 32 untuk training yang lebih stabil, maksimal 9 epoch dengan early stopping patience 3, dan data split 80:20 untuk evaluasi yang lebih robust. Model di-fine-tune kemudian dievaluasi menggunakan accuracy, weighted F1-score, precision, recall, confusion matrix, dan perplexity untuk memastikan peningkatan performa across all intent categories. Model final dilengkapi dengan confidence threshold mechanism untuk menangani input ambiguous dan disimpan ke direktori baru, sehingga chatbot dapat memberikan respons yang lebih akurat dan reliable untuk pertolongan pertama kesehatan mental.

h. Deployment

Tahap deployment mengimplementasikan TanyaRasa Mental Health Chatbot menggunakan Streamlit sebagai web interface dengan fitur word cloud visualization untuk menampilkan kata kunci kesehatan mental dari dataset training. Model IndoBERT yang telah di-fine-tune di-upload ke Hugging Face Hub (athayary/indobert) untuk akses mudah dan reproducible deployment. Aplikasi bekerja dengan flow: user input → preprocessing (Sastrawi) → model inference → intent classification → response selection, dilengkapi fitur real-time chat, analytics dashboard, debug mode, dan fallback similarity matching menggunakan teknologi stack Streamlit + PyTorch + Transformers untuk memberikan respons akurat pertolongan pertama kesehatan mental.

3.2 Implementasi Sistem

3.2.1 Pra-pemrosesan Data dan Analisis Eksplorasi

Pada tahap ini, data intent JSON diubah menjadi format yang dapat diproses oleh model. Pertama, data diratakan dari struktur hierarkis menjadi format tabular di mana setiap baris berisi satu pattern dan satu response yang terkait dengan intent tertentu. Selanjutnya, dilakukan analisis eksplorasi untuk memahami distribusi data, termasuk visualisasi word cloud untuk melihat kata-kata yang paling sering muncul, analisis panjang pattern, dan distribusi intent.

```
MULAI Pra-pemrosesan Data
    BACA file intents.json
    BUAT dataframe kosong dengan kolom [tag, patterns, responses]

    UNTUK setiap intent dalam data:
        AMBIL tag, patterns, responses
        UNTUK setiap pattern:
            TAMBAHKAN ke dataframe (tag, pattern, responses)

    BUAT word cloud dari semua patterns
    HITUNG distribusi panjang patterns
    ANALISIS distribusi intent
    TAMPILKAN visualisasi
SELESAI
```

Pseudocode 3.2.1 Implementasi Pra-pemrosesan Data dan Analisis Eksplorasi

Proses ini menghasilkan dataset yang terstruktur dengan total pattern yang dapat digunakan untuk pelatihan model, serta memberikan wawasan tentang karakteristik data melalui berbagai visualisasi statistik.

3.2.2 Pra-pemrosesan Teks Bahasa Indonesia

Tahap ini melakukan pembersihan dan normalisasi teks menggunakan teknik khusus untuk bahasa Indonesia. Setiap teks diubah menjadi huruf kecil, tanda baca dihilangkan, stopwords bahasa Indonesia dibuang, dan dilakukan stemming menggunakan Sastrawi stemmer untuk mengembalikan kata ke bentuk dasarnya.

```
FUNGSI preprocess_text(text):
    text = ubah_ke_huruf_kecil(text)
    text = hapus_tanda_baca(text)
    words = pisah_kata(text)
    words = hapus_stopwords(words, stopwords_indonesia)
    text = gabung_kata(words)
    text_bersih = stemming_sastrawi(text)
    KEMBALIKAN text_bersih

MULAI Pra-pemrosesan Teks
    INISIALISASI Sastrawi stemmer
    MUAT stopwords bahasa Indonesia

    UNTUK setiap pattern dan response:
        pattern_bersih = preprocess_text(pattern)
        response_bersih = preprocess_text(response)
        SIMPAN hasil ke dataframe
SELESAI
```

Pseudocode 3.2.2 Implementasi Pra-pemrosesan Teks Bahasa Indonesia

Hasil dari tahap ini adalah teks yang telah dinormalisasi dan siap untuk diproses oleh model IndoBERT, dengan menghilangkan noise dan mempertahankan informasi semantik yang penting.

3.2.3 Persiapan Model dan Tokenisasi

Pada tahap ini, model IndoBERT yang telah di-pretrain dimuat dan dikonfigurasi untuk tugas klasifikasi intent. Label intent dikodekan menggunakan LabelEncoder, dan tokenizer BERT digunakan untuk mengubah teks menjadi representasi numerik yang dapat dipahami model.

```
MULAI Persiapan Model
    INISIALISASI LabelEncoder
    y_encoded = encode_labels(intent_labels)
    num_labels = hitung_jumlah_label_unik(y_encoded)

    MUAT tokenizer dari MODEL_ID
    BUAT konfigurasi model dengan num_labels
    MUAT model IndoBERT dengan konfigurasi

    FUNGSI encode_texts(texts, max_len):
        UNTUK setiap text:
            encoded = tokenizer.encode_plus(
                text, max_length=max_len,
                padding='max_length', truncation=True
            )
            SIMPAN input_ids dan attention_mask
        KEMBALIKAN tensor input_ids dan attention_masks

    input_ids, attention_masks = encode_texts(patterns_bersih)
    labels = convert_to_tensor(y_encoded)
SELESAI

```

Pseudocode 3.2.3 Implementasi Persiapan Model dan Tokenisasi

Tahap ini menghasilkan representasi numerik dari teks yang kompatibel dengan arsitektur BERT, serta menyiapkan model dengan jumlah kelas yang sesuai dengan intent yang ada dalam dataset.

3.2.4 Pelatihan dan Evaluasi Model

Tahap ini merupakan inti dari pemodelan, di mana model IndoBERT di-fine-tune pada data intent yang telah disiapkan. Data dibagi menjadi set pelatihan dan validasi dengan rasio 80:20. Model dilatih menggunakan optimizer AdamW dengan learning rate 2e-5 dan dilengkapi dengan mekanisme early stopping untuk mencegah overfitting.

```
MULAI Pelatihan Model
    BUAT dataset dari input_ids, attention_masks, labels
    BAGI dataset menjadi train (80%) dan validation (20%)
    BUAT dataloader untuk batch processing

    PINDAHKAN model ke device (GPU/CPU)
    INISIALISASI optimizer AdamW dengan lr=2e-5
    SET epochs = 9, patience = 3

    UNTUK setiap epoch:
        SET model ke mode training
        total_loss = 0
        UNTUK setiap batch dalam train_dataloader:
            HITUNG forward pass
            HITUNG loss
            LAKUKAN backward pass
            UPDATE parameters
            total_loss += loss

        SET model ke mode evaluation
        val_loss = 0, predictions = [], true_labels = []
        UNTUK setiap batch dalam validation_dataloader:
            HITUNG predictions tanpa gradient
            KUMPULKAN hasil prediksi dan label asli

        JIKA val_loss < best_val_loss:
            SIMPAN model terbaik
            RESET patience counter
        SELAIN ITU:
            INCREMENT patience counter
            JIKA patience counter >= patience:
                HENTIKAN training
SELESAI
```

Kode Semu 3.2.4 Implementasi Pelatihan dan Evaluasi Model

Proses pelatihan dilakukan dengan monitoring performa pada set validasi menggunakan metrik akurasi, F1-score, precision, recall, dan perplexity untuk memastikan model tidak mengalami overfitting dan mencapai performa optimal.

3.2.5 Implementasi Sistem Prediksi dan Chatbot

Setelah model dilatih, tahap terakhir adalah implementasi sistem prediksi yang dapat menerima input teks dari pengguna dan memberikan respons yang sesuai. Sistem ini dilengkapi dengan threshold confidence untuk menangani input yang tidak dikenali.

```
FUNGSI predict_intent(text, threshold=0.3):
    SET model ke mode evaluation
    text_bersih = preprocess_text(text)
    input_ids, attention_mask = encode_texts([text_bersih])
    PINDAHKAN tensor ke device

    TANPA gradient:
        outputs = model(input_ids, attention_mask)
        logits = outputs.logits
        probabilities = softmax(logits)
        confidence, predicted_class = max(probabilities)

    JIKA confidence < threshold:
        KEMBALIKAN None, "Pesan tidak dipahami"
    SELAIN ITU:
        intent = decode_label(predicted_class)
        responses = ambil_responses_untuk_intent(intent)
        response = pilih_random(responses)
        KEMBALIKAN intent, response

MULAI Chatbot Loop
    TAMPILKAN "TanyaRasa Chatbot"
    SELAMA True:
        user_input = input("You: ")
        JIKA user_input == "quit":
            KELUAR dari loop
        intent, response = predict_intent(user_input)
        TAMPILKAN respons chatbot
SELESAI

```

Pseudocode 3.2.5 Implementasi Sistem Prediksi dan Chatbot

Sistem chatbot yang diimplementasikan mampu memproses input bahasa Indonesia, melakukan klasifikasi intent dengan tingkat confidence tertentu, dan memberikan respons yang relevan berdasarkan intent yang terdeteksi. Jika confidence di bawah threshold, sistem akan memberikan pesan bahwa input tidak dipahami, memastikan pengalaman pengguna yang lebih baik.

BAB 4 Hasil dan Pembahasan

4.1 Konfigurasi *Environment*

Seluruh rangkaian percobaan dalam penelitian ini dilakukan menggunakan platform *Google Colaboratory*. Platform ini dilengkapi dengan akselerator GPU NVIDIA A100, yang memberikan kemampuan komputasi tinggi terutama pada proses pelatihan model berbasis arsitektur Transformer.

Selain itu, pengembangan sistem dilakukan menggunakan bahasa pemrograman Python versi 3.12, dengan bantuan berbagai pustaka pendukung. Beberapa pustaka utama yang digunakan antara lain merupakan: pandas untuk pengelolaan data, nltk untuk proses pembersihan teks dan penghapusan stopword, serta Sastrawi untuk stemming Bahasa Indonesia, dan scikit-learn & numpy digunakan dalam perhitungan metrik evaluasi seperti akurasi, precision, recall, dan F1-score. Model yang digunakan adalah IndoBERT dan CENDOL, yang diakses melalui pustaka transformers dari Hugging Face.

4.2 Hasil Penelitian

Disini bagian ini akan diuraikan temuan komprehensif dari berbagai eksperimen yang telah dijalankan untuk menganalisis dan meningkatkan performa model chatbot kesehatan mental. Presentasi hasil penelitian diorganisir secara bertahap, mencakup evaluasi komparatif antar arsitektur model, investigasi dampak variasi epoch terhadap pembelajaran, serta analisis mendalam mengenai pengaruh tipe dataset dan metodologi fine-tuning. Semua conclusions yang disampaikan berlandaskan pada pengukuran metrik evaluasi yang terstandarisasi guna menjamin validitas dan reliabilitas dalam penilaian efektivitas setiap konfigurasi eksperimental.

4.2.1 Perbandingan Performa Model Berdasarkan Jumlah Epoch

Jumlah epoch merupakan salah satu parameter penting dalam proses pelatihan model bahasa. Semakin tinggi jumlah epoch, maka semakin banyak model "melihat" data pelatihan, yang berpotensi meningkatkan akurasi atau menurunkan nilai loss dan perplexity. Namun, peningkatan epoch juga berisiko menyebabkan overfitting. Oleh karena itu, pada bagian ini dilakukan perbandingan kinerja model CENDOL dan IndoBERT pada dua konfigurasi epoch: **12 epoch** dan **30 epoch**, dengan parameter lainnya tetap sama (batch size = 64, learning rate = 5e-5, dataset = intent classification).

Model	Validation Perplexity	Validation Accuracy	F1 Score	Train Loss	Validation Loss	Precision	Recall
CENDOL	**3.8953**	0.2344	0.2192	1.4715	**1.3598**	0.2319	0.2344
IndoBERT	14.9250	**0.6520**	**0.6284**	**0.9721**	2.7030	**0.6488**	**0.6520**

Tabel 4.1. Perbandingan Kinerja Model Berdasarkan Epoch 12

Berdasarkan hasil evaluasi pada epoch ke-12, IndoBERT menunjukkan superioritas yang jelas dalam metrik-metrik klasifikasi yang relevan untuk tugas intent classification. Model

IndoBERT mencapai validation accuracy sebesar 65.20%, hampir tiga kali lipat lebih tinggi dibandingkan CENDOL yang hanya mencapai 23.44%. Demikian pula dengan F1 score, IndoBERT meraih skor 62.84% sementara CENDOL hanya 21.92%. Perbedaan performa ini sangat krusial karena ketepatan klasifikasi intent secara langsung mempengaruhi kemampuan chatbot dalam memberikan respons yang sesuai dan relevan dengan kebutuhan pengguna.

Meskipun CENDOL menunjukkan validation perplexity yang lebih rendah (3.8953 vs 14.9250), hal ini tidak dapat dijadikan indikator utama keberhasilan model dalam tugas klasifikasi intent. Perplexity yang rendah pada CENDOL justru mengindikasikan bahwa model tersebut sangat percaya diri terhadap prediksi yang dihasilkan, meskipun prediksi tersebut seringkali salah.

Model	Validation Perplexity	Validation Accuracy	F1 Score	Train Loss	Validation Loss	Precision	Recall
CENDOL	**2.2481**	0.4505	0.4378	0.7668	**0.8101**	0.4565	0.4505
IndoBERT	6.8942	**0.6374**	**0.6143**	**0.4288**	1.9307	**0.6420**	**0.6374**

Tabel 4.2. Perbandingan Kinerja Model Berdasarkan Epoch 30

Pada epoch ke-30, pola performa yang sama terlihat konsisten. IndoBERT mempertahankan keunggulannya dengan validation accuracy 63.74% dan F1 score 61.43%, sementara CENDOL mengalami peningkatan menjadi 45.05% untuk accuracy dan 43.78% untuk F1 score. Meskipun CENDOL menunjukkan improvement yang signifikan dibandingkan epoch 12, performanya masih jauh tertinggal dari IndoBERT.

Analisis lebih lanjut menunjukkan bahwa IndoBERT mengalami sedikit penurunan performa dari epoch 12 ke epoch 30, yang mengindikasikan potensi terjadinya slight overfitting. Train loss IndoBERT turun drastis dari 0.9721 menjadi 0.4288, sementara validation accuracy sedikit menurun dari 65.20% menjadi 63.74%. Hal ini menunjukkan bahwa model mulai terlalu "menghafal" training data daripada belajar generalisasi yang baik.

4.2.2 Analisis Pemilihan Model Optimal

Berdasarkan evaluasi komprehensif yang telah dilakukan, IndoBERT dipilih sebagai model optimal untuk fine-tuning chatbot kesehatan mental. Pemilihan ini didasarkan pada superioritas IndoBERT dalam metrik klasifikasi yang relevan untuk tugas intent classification, yaitu accuracy, F1 score, precision, dan recall. IndoBERT secara konsisten mengungguli CENDOL dengan margin yang signifikan - mencapai accuracy 2.78 kali lebih tinggi (65.20% vs 23.44%) dan F1 score 2.87 kali lebih baik (62.84% vs 21.92%) pada epoch 12.

Keunggulan IndoBERT juga terlihat dari keseimbangan precision-recall yang optimal (64.88% vs 65.20%), menunjukkan kemampuan model dalam mengidentifikasi intent dengan akurat tanpa menghasilkan terlalu banyak kesalahan klasifikasi. Hal ini sangat crucial untuk aplikasi chatbot kesehatan mental dimana misklasifikasi intent dapat berdampak pada kualitas respons dan bahkan keselamatan pengguna. Meskipun CENDOL menunjukkan perplexity yang lebih rendah, ini justru mengindikasikan overconfidence dalam prediksi yang salah, sementara IndoBERT dengan train loss yang lebih rendah menunjukkan proses pembelajaran yang lebih efektif.

Dari perspektif praktis, IndoBERT dengan accuracy ~65% memungkinkan chatbot memahami intent pengguna dengan benar pada 2 dari 3 interaksi, sementara CENDOL hanya berhasil pada 1 dari 4-5 interaksi. Berdasarkan analisis ini, IndoBERT pada epoch 12 dipilih sebagai baseline untuk fine-tuning selanjutnya karena menunjukkan performa optimal sebelum terjadinya overfitting pada epoch 30.

4.2.3 Fine-Tuning IndoBERT

Setelah memilih IndoBERT sebagai model dasar, langkah selanjutnya adalah menentukan konfigurasi hyperparameter yang optimal untuk tugas klasifikasi intent kesehatan mental. Eksperimen dilakukan dengan memvariasikan dua hyperparameter kunci, yaitu adalah learning rate dan batch size. Learning rate yang diuji adalah 2e-5, 3e-5, 4e-5, dan 5e-5, sementara batch size yang diuji adalah 16, 32, dan 64. Setiap kombinasi dilatih dengan jumlah epoch maksimum 9 dan mekanisme early stopping dengan patience 3 untuk mencegah overfitting.

Batch Size	Learning Rate	Validation Perplexity	Validation Accuracy	F1 Score (Weighted)	Train Loss	Validation Loss	Precision (Weighted)	Recall (Weighted)
16	2e-5	2.5180	0.8333	0.8195	0.5370	0.9235	0.8504	0.8333
	3e-5	3.0575	0.8077	0.7959	0.4889	1.1176	0.8281	0.8077
	4e-5	2.2566	0.8626	0.8538	0.5002	0.8139	0.8777	0.8626
	5e-5	2.7484	0.7985	0.7813	0.5114	1.0110	0.8089	0.7985
32	2e-5	2.9305	0.8462	0.8401	0.5846	1.0752	0.8693	0.8462
	3e-5	2.3584	0.8388	0.8255	0.5359	0.8580	0.8482	0.8388
	4e-5	2.4434	0.8498	0.8434	0.5198	0.8934	0.8732	0.8498
	5e-5	2.3088	0.8407	0.8256	0.5052	0.8367	0.8434	0.8407
64	2e-5	2.3793	0.8608	0.8497	0.6396	0.8668	0.8653	0.8608
	3e-5	2.8057	0.8059	0.7921	0.5838	1.0316	0.8223	0.8059
	4e-5	2.5744	0.8260	0.8227	0.5760	0.9456	0.8593	0.8260
	5e-5	2.4767	0.8462	0.8289	0.5445	0.9069	0.8500	0.8462

Tabel 4.3. Perbandingan Kinerja Model Berdasarkan Batch Size dan Learning Rate

Hasil analisis menunjukkan bahwa ukuran batch dan learning rate memiliki interaksi yang saling memengaruhi terhadap performa model. Ukuran batch 16 menghasilkan performa terbaik pada learning rate 4e-5 dengan Validation Accuracy sebesar 0.8626 dan F1 Score tertinggi sebesar 0.8538. Namun, konfigurasi ini menunjukkan sensitivitas tinggi terhadap variasi learning rate lainnya. Ukuran batch 32 memberikan performa yang lebih stabil di berbagai learning rate, dengan akurasi yang konsisten di atas 0.83, menjadikannya pilihan yang relatif aman. Sementara itu, ukuran batch 64 menunjukkan performa optimal pada learning rate 2e-5, namun mengalami penurunan performa seiring peningkatan learning rate.

Dari sisi laju pembelajaran, learning rate rendah (2e-5) cenderung menghasilkan performa yang stabil, terutama pada ukuran batch yang lebih besar. Learning rate menengah (3e-5 dan 4e-5) menunjukkan variabilitas yang lebih tinggi, namun mampu menghasilkan konfigurasi terbaik secara keseluruhan. Sebaliknya, learning rate tinggi (5e-5) umumnya menurunkan performa model, mengindikasikan potensi ketidakstabilan dalam proses pelatihan.

Dengan mempertimbangkan seluruh metrik evaluasi, konfigurasi dengan **batch size 16 dan learning rate 4e-5** dipilih sebagai yang paling optimal dan akan digunakan untuk melatih model akhir yang diimplementasikan dalam sistem chatbot TanyaRasa. Pemilihan ini didasarkan pada bukti kuantitatif yang menunjukkan bahwa konfigurasi tersebut mampu mencapai keseimbangan terbaik antara akurasi, presisi, dan recall dalam menjalankan tugas klasifikasi intent. Model dengan konfigurasi tersebut menunjukkan peningkatan performa yang dramatis dibandingkan baseline. Validation accuracy meningkat dari 65.20% (epoch 12 baseline) menjadi **86.26%**, mencatat improvement sebesar 21.06 poin persentase. F1 score mengalami peningkatan dari 62.84% menjadi **85.38%**, menunjukkan perbaikan kemampuan klasifikasi yang seimbang di seluruh kategori intent.

Keseimbangan precision-recall juga menunjukkan optimisasi yang sangat baik, dengan precision weighted mencapai **87.77%** dan recall weighted **86.26%**. Tingginya precision mengindikasikan minimnya false positive predictions, sementara recall yang tinggi menunjukkan kemampuan model mendeteksi berbagai kategori intent dengan komprehensif. Validation perplexity sebesar 2.2566 menunjukkan confidence level yang proporsional terhadap akurasi prediksi.

4.2.4 Evaluasi Kualitas Respons Teks

Selain mengukur akurasi model dalam mengklasifikasikan intent pengguna, evaluasi lebih lanjut dilakukan untuk mengukur kualitas kalimat balasan (respons) yang diberikan oleh chatbot. Setelah model memprediksi sebuah intent, sistem akan memberikan salah satu respons yang telah didefinisikan untuk intent tersebut. Untuk mengevaluasi seberapa relevan dan berkualitas respons ini dibandingkan dengan respons referensi, digunakan metrik evaluasi generasi teks standar, yaitu BLEU dan ROUGE.

BLEU (Bilingual Evaluation Understudy) digunakan untuk mengukur tingkat presisi n-gram, dengan membandingkan sejauh mana kata atau frasa dalam respons yang dihasilkan sesuai dengan respons referensi. Di sisi lain, ROUGE (Recall-Oriented Understudy for Gisting Evaluation) berfokus pada aspek recall, dengan mengukur seberapa besar proporsi kata atau frasa dari respons referensi yang berhasil muncul dalam respons yang dihasilkan oleh chatbot.

Evaluasi ini dilakukan menggunakan model dengan konfigurasi terbaik berdasarkan hasil klasifikasi intent sebelumnya, yaitu model dengan batch size 16 dan learning rate 4e-5. Berdasarkan hasil evaluasi, diperoleh skor BLEU sebesar **0.6237**, ROUGE-1 F1 sebesar **0.8267**, ROUGE-2 F1 sebesar **0.7273**, dan ROUGE-L F1 sebesar **0.8262.**

Hasil tersebut menunjukkan bahwa chatbot mampu menghasilkan respons yang relevan dan sesuai dengan ekspektasi. Skor ROUGE-1 dan ROUGE-L yang tinggi menunjukkan bahwa respons yang diberikan mengandung kata-kata kunci penting serta

urutan kata yang menyerupai respons ideal, mencerminkan relevansi dan struktur yang tepat. Selain itu, skor ROUGE-2 yang cukup tinggi menunjukkan bahwa pasangan kata yang digunakan dalam respons cukup sering cocok dengan respons referensi, mengindikasikan tingkat kelancaran dan koherensi yang baik. Skor BLEU yang diperoleh juga mencerminkan bahwa respons yang dihasilkan memiliki presisi yang baik terhadap referensi, menunjukkan kesesuaian dalam pilihan kata dan struktur kalimat.

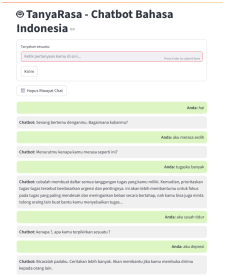
Secara keseluruhan, hasil evaluasi ini melengkapi metrik klasifikasi yang telah dibahas sebelumnya. Dengan tingkat akurasi klasifikasi sebesar 86.26% dan kualitas respons yang tinggi berdasarkan metrik BLEU dan ROUGE, dapat disimpulkan bahwa sistem chatbot TanyaRasa tidak hanya efektif dalam memahami maksud pengguna, tetapi juga mampu memberikan balasan yang relevan, natural, dan mendekati kualitas respons ideal.

4.3 Hasil Deployment

Sistem diimplementasikan menjadi sebuah aplikasi web interaktif menggunakan framework streamlit. Aplikasi ini menyediakan antarmuka sederhana bagi pengguna menguji kemampuan model secara langsung dan dapat diakses melalui tautan https://tanyarasa.streamlit.app/.

Berikut adalah langkah-langkah penggunaan chatbot TanyaRasa sebagai user:

a. **Menginput Pertanyaan**
   Pengguna memasukkan pertanyaan atau pernyataan ke dalam kolom input teks yang tersedia. Pertanyaan ini berkaitan dengan topik yang didukung oleh chatbot, seperti gizi, makanan sehat, dan lain-lain.
b. **Mengirim Pertanyaan**
   Setelah mengetik pertanyaan, pengguna menekan tombol **"Kirim"** untuk mengirimkan input ke sistem chatbot.
c. **Menerima Balasan dari Chatbot**
   Chatbot akan memproses pertanyaan yang dikirim dan menampilkan balasan secara otomatis. Balasan ditampilkan dalam tampilan percakapan berbentuk balon chat, yang memisahkan pesan dari pengguna dan respons dari chatbot.
d. **Melanjutkan Percakapan**
   Pengguna dapat melanjutkan interaksi dengan chatbot secara berkelanjutan, cukup dengan mengirim pertanyaan-pertanyaan tambahan sesuai kebutuhan.
e. **Menghapus Riwayat Percakapan**
   Apabila pengguna ingin menghapus riwayat chat yang telah terjadi, tersedia tombol **"🗑️ Hapus Riwayat Chat"**. Dengan menekan tombol ini, seluruh percakapan akan dihapus, dan tampilan akan disegarkan untuk memulai interaksi baru.



Gambar 4.1 Tampilan Aplikasi

BAB 5 KESIMPULAN

Berdasarkan hasil penelitian, eksperimen, dan analisis yang telah dilakukan dalam proyek ini, diperoleh beberapa kesimpulan sebagai berikut:

1. Terkait perancangan dan pembangunan chatbot berbahasa Indonesia yang mampu mempelajari dan mengklasifikasikan intent pengguna secara akurat, dapat dikatakan bahwa chatbot *TanyaRasa* berhasil dirancang dan dibangun dengan pendekatan text mining berbasis model LLM berbahasa Indonesia, khususnya IndoBERT. Proses pengembangan diawali dari pengumpulan dan penerjemahan dataset kesehatan mental, dilanjutkan dengan preprocessing, pelatihan model, dan implementasi sistem chatbot berbasis web.

2. Terkait efektivitas proses pembelajaran klasifikasi intent menggunakan model LLM dalam meningkatkan pemahaman chatbot terhadap bahasa alami dalam konteks bahasa Indonesia, Model IndoBERT menunjukkan performa klasifikasi intent yang unggul dibandingkan CENDOL, dengan peningkatan akurasi validasi dari 65,20% menjadi 85,35% setelah fine-tuning. F1-score dan precision-recall yang tinggi juga menunjukkan bahwa IndoBERT mampu mengenali berbagai intent secara komprehensif dan akurat. Hal ini membuktikan bahwa proses fine-tuning pada model LLM berbahasa Indonesia secara signifikan meningkatkan kemampuan chatbot dalam memahami konteks lokal dan memberikan respons yang relevan.

Secara keseluruhan, penelitian ini menunjukkan bahwa pemanfaatan model bahasa besar lokal seperti IndoBERT, jika dikombinasikan dengan preprocessing dan tuning yang tepat, mampu menghasilkan sistem chatbot yang efektif untuk mendukung layanan pertolongan pertama kesehatan mental dalam bahasa Indonesia.

DAFTAR PUSTAKA

Abd-Alrazaq, Alaa A., et al. “Perceptions and Opinions of Patients About Mental Health Chatbots: Scoping Review.” *Journal of Medical Internet Research*, vol. 23, no. 1, Jan. 2021, p. e17828. *DOI.org (Crossref)*, https://doi.org/10.2196/17828.

Ahmadian, Hendri, et al. “Hybrid Models for Emotion Classification and Sentiment Analysis in Indonesian Language.” *Applied Computational Intelligence and Soft Computing*, edited by Anandakumar Haldorai, vol. 2024, no. 1, Jan. 2024, p. 2826773. *DOI.org (Crossref)*, https://doi.org/10.1155/2024/2826773.

Baharuddin, Fikri, and Mohammad Farid Naufal. “Fine-Tuning IndoBERT for Indonesian Exam Question Classification Based on Bloom’s Taxonomy.” *Journal of Information Systems Engineering and Business Intelligence*, vol. 9, no. 2, Nov. 2023, pp. 253–63. *DOI.org (Crossref)*, https://doi.org/10.20473/jisebi.9.2.253-263.

Cahyawijaya, Samuel, et al. *Cendol: Open Instruction-Tuned Generative Large Language Models for Indonesian Languages*. arXiv:2404.06138, arXiv, 8 July 2024. *arXiv.org*, https://doi.org/10.48550/arXiv.2404.06138.

Celikyilmaz, Asli, et al. *Evaluation of Text Generation: A Survey*. arXiv, 2020. *DOI.org (Datacite)*, https://doi.org/10.48550/ARXIV.2006.14799.

Devlin, Jacob, et al. *BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding*. arXiv, 2018. *DOI.org (Datacite)*, https://doi.org/10.48550/ARXIV.1810.04805.

Dwiyono, Aswin, et al. “Analisis Perbandingan Klasifikasi Intent Chatbot Menggunakan Deep Learning BERT, RoBERTa, Dan IndoBERT.” *Journal of Information System Research (JOSH)*, vol. 6, no. 1, Oct. 2024, pp. 595–606. *DOI.org (Crossref)*, https://doi.org/10.47065/josh.v6i1.6051.

Géron, Aurélien. *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. Third edition, O’Reilly, 2023.

Hickman, Louis, et al. “Text Preprocessing for Text Mining in Organizational Research: Review and Recommendations.” *Organizational Research Methods*, vol. 25, no. 1, Jan. 2022, pp. 114–46. *DOI.org (Crossref)*, https://doi.org/10.1177/1094428120971683.

Hu, Edward J., et al. *LoRA: Low-Rank Adaptation of Large Language Models*. arXiv:2106.09685, arXiv, 16 Oct. 2021. *arXiv.org*, https://doi.org/10.48550/arXiv.2106.09685.

Kannan, Eswariah, and Lakshmi Anusha Kothamasu. “Fine-Tuning BERT Based Approach for Multi-Class Sentiment Analysis on Twitter Emotion Data.” *Ingénierie Des Systèmes d Information*, vol. 27, no. 1, Feb. 2022, pp. 93–100. *DOI.org (Crossref)*, https://doi.org/10.18280/isi.270111.

Koto, Fajri, et al. *IndoLEM and IndoBERT: A Benchmark Dataset and Pre-Trained Language Model for Indonesian NLP*. arXiv:2011.00677, arXiv, 2 Nov. 2020. *arXiv.org*, https://doi.org/10.48550/arXiv.2011.00677.

Kowalczuk, Bartłomiej, and Karolina Kuligowska. “Enhancing Chatbot Intent Classification Using Active Learning Pipeline for Optimized Data Preparation.” *Journal of Applied Economic Sciences (JAES)*, vol. 19, no. 16, Sept. 2024, p. 317. *DOI.org (Crossref)*, https://doi.org/10.57017/jaes.v19.3(85).07.

Kuchlous, Sahil, and Madhura Kadaba. “Short Text Intent Classification for Conversational Agents.” *2020 IEEE 17th India Council International Conference (INDICON)*, IEEE, 2020, pp. 1–4. *DOI.org (Crossref)*, https://doi.org/10.1109/INDICON49873.2020.9342516.

Lee, Seungjun, et al. “A Survey on Evaluation Metrics for Machine Translation.” *Mathematics*, vol. 11, no. 4, Feb. 2023, p. 1006. *DOI.org (Crossref)*, https://doi.org/10.3390/math11041006.

Lin, Chin-Yew. “ROUGE: A Package for Automatic Evaluation of Summaries.” *Text Summarization Branches Out*, Association for Computational Linguistics, 2004, pp. 74–81. *ACLWeb*, https://aclanthology.org/W04-1013/.

Lin, Tianyang, et al. “A Survey of Transformers.” *AI Open*, vol. 3, 2022, pp. 111–32. *DOI.org (Crossref)*, https://doi.org/10.1016/j.aiopen.2022.10.001.

Ouaddi, Charaf, et al. “Assessing the Effectiveness of Large Language Models for Intent Detection in Tourism Chatbots: A Comparative Analysis and Performance Evaluation.” *Scientific African*, vol. 28, June 2025, p. e02649. *DOI.org (Crossref)*, https://doi.org/10.1016/j.sciaf.2025.e02649.

Papineni, Kishore, et al. “BLEU: A Method for Automatic Evaluation of Machine Translation.” *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics - ACL ’02*, Association for Computational Linguistics, 2001, p. 311. *DOI.org (Crossref)*, https://doi.org/10.3115/1073083.1073135.

Souha, Adnane, et al. “Pre-Trained Models for Intent Classification in Chatbot: Comparative Study and Critical Analysis.” *2023 6th International Conference on Advanced Communication Technologies and Networking (CommNet)*, IEEE, 2023, pp. 1–6. *DOI.org (Crossref)*, https://doi.org/10.1109/CommNet60167.2023.10365312.

Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. Second edition, Fifteenth impression, 2022., Pearson, 2022.

Vajjala, Sowmya, et al. *Practical Natural Language Processing: A Comprehensive Guide to Building Real-World NLP Systems*. First edition, O’Reilly, 2020.

Vaswani, Ashish, et al. *Attention Is All You Need*. arXiv:1706.03762, arXiv, 2 Aug. 2023. *arXiv.org*, https://doi.org/10.48550/arXiv.1706.03762.

Wilie, Bryan, et al. “IndoNLU: Benchmark and Resources for Evaluating Indonesian Natural Language Understanding.” *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, Association for Computational Linguistics, 2020, pp. 843–57. *DOI.org (Crossref)*, https://doi.org/10.18653/v1/2020.aacl-main.85.

LAMPIRAN