

# **Pembangunan *Chatbot* Untuk Layanan Akademik Menggunakan *Transformer***

**Tugas Akhir**

**diajukan untuk memenuhi salah satu syarat  
memperoleh gelar sarjana  
dari Program Studi S1 Informatika  
Fakultas Informatika  
Universitas Telkom**

**1301202287**

**Faqihuddin Habibi**



**Program Studi Sarjana Informatika  
Fakultas Informatika  
Universitas Telkom**

**Bandung**

**2024**

## LEMBAR PENGESAHAN

### PEMBANGUNAN *CHATBOT* UNTUK LAYANAN AKADEMIK MENGUNAKAN *TRANSFORMER*

*Chatbot Development for Academic Services Using Transformer*

NIM : 1301202287

Faqihuddin Habibi

Tugas akhir ini telah diterima dan disahkan untuk memenuhi sebagian syarat memperoleh gelar pada Program Studi Sarjana Informatika

Fakultas Informatika

Universitas Telkom

Bandung, 6 Agustus 2024

Menyetujui

Pembimbing I,

Dr. Arie Ardiyanti Suryani, S.T., M.T.

NIP: 02770027

Ketua Program Studi  
Sarjana Informatika,

Dr. Erwin Budi Setiawan, S.Si, M.T.

NIP: 00760045

**LEMBAR PERNYATAAN**

Dengan ini saya Faqihuddin Habibi, menyatakan sesungguhnya bahwa Tugas Akhir saya dengan judul Pembangunan *Chatbot* Untuk Layanan Akademik Menggunakan *Transformer* beserta dengan seluruh isinya adalah merupakan hasil karya sendiri, dan saya tidak melakukan penjiplakan yang tidak sesuai dengan etika keilmuan yang berlaku dalam masyarakat keilmuan. Saya siap menanggung resiko/sanksi yang diberikan jika di kemudian hari ditemukan pelanggaran terhadap etika keilmuan dalam buku TA atau jika ada klaim dari pihak lain terhadap keaslian karya.

Bandung, 6 Agustus 2024

Yang Menyatakan



Faqihuddin Habibi

## Pembangunan *Chatbot* Untuk Layanan Akademik Menggunakan *Transformer*

Faqihuddin Habibi<sup>1</sup>, Dr. Arie Ardiyanti Suryani, S.T., M.T.<sup>2</sup>

<sup>1,2</sup>Fakultas Informatika, Universitas Telkom, Bandung

<sup>1</sup>[fhabibi@students.telkomuniversity.ac.id](mailto:fhabibi@students.telkomuniversity.ac.id), <sup>2</sup>[ardiyanti@telkomuniversity.ac.id](mailto:ardiyanti@telkomuniversity.ac.id)

### Abstrak

Dalam era informasi yang serba cepat, mahasiswa di perguruan tinggi sering mengalami kesulitan dalam mengakses informasi akademik yang mereka butuhkan. Penelitian ini bertujuan mengembangkan *chatbot* berbasis IndoBERT dengan arsitektur *transformer* untuk menyediakan informasi akademik secara tepat waktu dan kontekstual. Model IndoBERT dipilih karena kemampuannya memahami konteks percakapan berbahasa Indonesia dengan lebih baik. Analisis dilakukan terhadap variasi *batch size* dan jumlah *epoch* selama pelatihan untuk menemukan konfigurasi optimal yang seimbang antara akurasi dan waktu pelatihan. Hasil menunjukkan bahwa *batch size* 128 dan jumlah *epoch* 125 memberikan keseimbangan terbaik. Meskipun demikian, pengujian hasil prediksi menunjukkan adanya kesulitan dalam menangani pertanyaan dengan kata kunci yang terdapat pada beberapa *label*, yang disebabkan oleh ketidakkonsistenan dalam pembuatan *patterns* pada *dataset*. Penelitian ini menyarankan perbaikan dan penambahan variasi *patterns* untuk meningkatkan akurasi dalam menangani pertanyaan yang lebih kompleks dan ambigu.

**Kata kunci :** *Chatbot*, IndoBERT, *Transformer*, Layanan Akademik

### Abstract

*In the fast-paced information era, university students often face difficulties in accessing the academic information they need. This study aims to develop an IndoBERT-based chatbot with a transformer architecture to provide timely and contextual academic information. The IndoBERT model was chosen for its ability to better understand the context of Indonesian conversations. An analysis was conducted on variations in batch size and the number of epochs during training to find an optimal configuration that balances accuracy and training time. The results show that a batch size of 128 and 125 epochs provide the best balance. However, prediction testing revealed difficulties in handling questions with keywords present in multiple labels, due to inconsistencies in pattern creation within the dataset. This study suggests improvements and additions to the pattern variations to enhance accuracy in dealing with more complex and ambiguous questions.*

**Keywords:** *Chatbot*, IndoBERT, *Transformer*, Academic Services

## 1. Pendahuluan

### Latar Belakang

Dalam era persebaran informasi yang sangat cepat, mahasiswa di perguruan tinggi sering menghadapi kendala dalam mengakses informasi akademik yang mereka butuhkan untuk menunjang proses belajar mengajar. Meskipun informasi tersedia dalam jumlah besar, tantangan utama terletak pada kemampuan untuk menemukan dan mengakses data yang relevan secara cepat dan efisien. Untuk mengatasi masalah ini, pengembangan *chatbot* yang mampu memberikan informasi akademik secara tepat waktu menjadi sangat penting. *Chatbot* ini tidak hanya akan membantu mahasiswa dalam memperoleh informasi yang dibutuhkan, tetapi juga dapat meningkatkan efisiensi administrasi akademik secara keseluruhan [1].

*Chatbot* adalah program komputer yang mampu berkomunikasi dalam bahasa alami tentang topik yang telah diintegrasikan ke dalam model pengetahuannya [2]. *Chatbot* dapat membantu mahasiswa mengakses informasi penting secara cepat dan efisien, seperti kalender akademik, keringanan biaya kuliah, atau prosedur administratif. Jika diterapkan dengan baik, *chatbot* bisa menjadi alat yang efektif dalam memenuhi kebutuhan informasi mahasiswa.

BERT (*Bidirectional Encoder Representations from Transformers*), sebagai model yang dibangun di atas arsitektur *transformer* yang memiliki kemampuan memahami konteks dengan lebih baik [3]. Model ini telah diimplementasikan dalam IndoBERT yang dikembangkan khusus untuk bahasa Indonesia [4]. Dengan menggunakan IndoBERT, *chatbot* diharapkan mampu memberikan respons yang lebih tepat dan sesuai dengan konteks percakapan. Penelitian ini diharapkan dapat memberikan solusi efektif untuk meningkatkan aksesibilitas informasi akademik di perguruan tinggi, serta berkontribusi pada pengembangan teknologi *chatbot* yang lebih cerdas dan kontekstual dalam bahasa Indonesia.

## Topik dan Batasannya

Penelitian ini berfokus pada pengembangan *chatbot* berbasis IndoBERT untuk menyajikan informasi akademik secara efisien kepada mahasiswa di perguruan tinggi. Dengan memanfaatkan kemampuan IndoBERT dalam memahami konteks percakapan berbahasa Indonesia. Dalam penelitian ini, batasannya meliputi analisis efektivitas *chatbot* dalam menjawab pertanyaan terkait Layanan Akademik pada Program Studi S1 Informatika, khususnya dalam topik Yudisium, *Cumlaude*, *Similarity*, Kerja Praktek, Tugas Akhir, Cuti, Undur Diri, Wisuda, Ijazah, Praktikum, Grup Kelas, Kalender Akademik. *Chatbot* ini hanya terbatas pada kasus-kasus tersebut dan tidak dapat memberikan jawaban untuk pertanyaan yang berada di luar cakupan *dataset* yang digunakan. Selain itu, sistem ini memiliki keterbatasan dalam memahami pertanyaan diluar konteks akademik atau yang memiliki perbedaan semantik dengan sintaksis yang digunakan, sehingga tidak mampu memberikan jawaban yang tepat untuk pertanyaan-pertanyaan tersebut.

## Tujuan

Tujuan penelitian ini adalah mengembangkan *chatbot* berbasis IndoBERT yang dapat menyajikan informasi akademik secara tepat waktu dan kontekstual. Dengan memanfaatkan kemampuan IndoBERT, *chatbot* ini diharapkan mampu memahami konteks percakapan berbahasa Indonesia.

## Hipotesis

Hipotesis penelitian ini menyatakan bahwa pengembangan *chatbot* berbasis IndoBERT dengan arsitektur *transformer* telah memberikan hasil yang baik dalam memahami konteks percakapan berbahasa Indonesia [4]. Oleh karena itu, dengan menggunakan pendekatan ini, diasumsikan bahwa *chatbot* tersebut akan mampu memberikan informasi akademik secara tepat waktu dan relevan, sehingga dapat meningkatkan aksesibilitas informasi bagi mahasiswa di perguruan tinggi secara signifikan.

## 2. Studi Terkait

Salah satu penelitian terkait adalah karya Hafidz et al. (2024) yang mengembangkan *chatbot* berbasis BERT untuk informasi pariwisata halal di Sumatera Barat. Perbedaan utama terletak pada model yang digunakan, yaitu bert-base-multilingual-uncased, sedangkan penelitian ini menggunakan IndoBERT dengan fokus pada layanan akademik. *Chatbot* ini berhasil mencapai akurasi 96% dalam mengelompokkan pertanyaan dan memberikan respons. Penelitian ini relevan sebagai referensi dalam pengembangan *chatbot* untuk layanan akademik, khususnya dalam konteks bahasa Indonesia [5].

Penelitian serupa dilakukan oleh Wulandari et al. (2023) yang mengimplementasikan *chatbot* menggunakan *framework* RASA dan algoritma BERT untuk layanan informasi wisata di Kota Pati. Perbedaan utama terletak pada penggunaan *library* RASA, sedangkan penelitian ini menggunakan *pre-trained model* IndoBERT dengan fokus pada layanan akademik. *Chatbot* ini dirancang untuk memahami niat pengguna dan memberikan respons berdasarkan informasi wisata yang tersedia. Penelitian ini berhasil mencapai akurasi sebesar 85% dalam pengujian *chatbot*, yang relevan sebagai referensi dalam pengembangan *chatbot* untuk layanan akademik berbasis bahasa Indonesia [6].

Penelitian serupa dilakukan oleh Saraswati et al. (2023) yang menggunakan algoritma BERT untuk analisis sentimen terhadap perundungan siber pada Twitter. Perbedaan utama terletak pada fokus penelitian, yaitu analisis sentimen perundungan siber di Twitter, sedangkan penelitian ini berfokus pada pengembangan *chatbot* untuk layanan akademik. Penelitian ini menunjukkan bahwa BERT dapat mengidentifikasi perundungan siber dengan akurasi 81%. Studi ini menyoroti efektivitas BERT dalam pemrosesan bahasa alami dan relevansinya dalam aplikasi berbasis teks lainnya, termasuk *chatbot* layanan akademik [7].

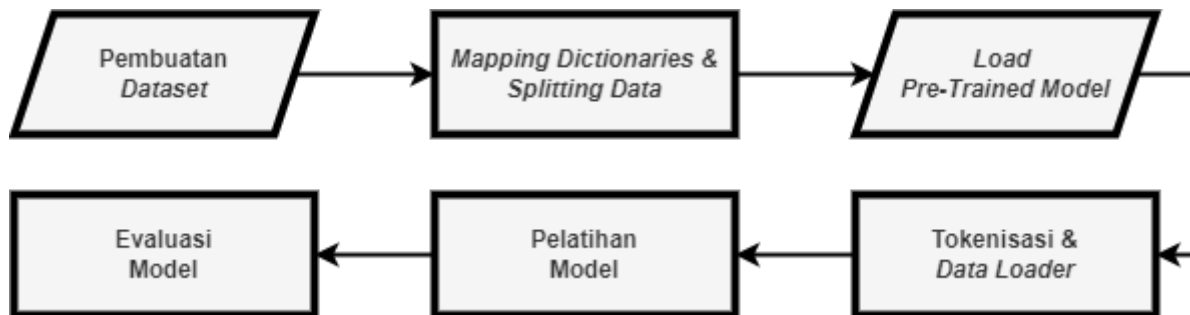
### 2.1. Landasan Teori

*Chatbot* adalah program komputer yang mampu berkomunikasi dalam bahasa alami tentang topik yang telah diintegrasikan ke dalam model pengetahuannya [2]. Mereka sering digunakan untuk menyediakan layanan pelanggan, membantu dalam proses pemesanan, atau memberikan informasi. Contoh implementasi *chatbot* antara lain dalam sistem pemesanan pada *coffee shop*, layanan publik kesehatan, dan layanan *customer service* pada *e-commerce* [8].

Dalam pengembangan *chatbot*, peran *Natural Language Processing* (NLP) sangat penting. NLP (*Natural Language Processing*) adalah bagian dari *Artificial Intelligence* yang berupaya membuat komputer dapat memahami dan memberikan output dalam bentuk bahasa manusia. Dengan menggunakan NLP, *chatbot* dapat memahami dan memproses bahasa alami manusia, memungkinkan mereka memberikan respons yang lebih kontekstual dan relevan terhadap pertanyaan atau permintaan pengguna. Dengan membangun model bahasa alami dan memproses input pengguna, NLP menjadi pondasi yang kuat dalam meningkatkan kualitas interaksi antara manusia dan *chatbot* [9].

*Transformer* adalah model arsitektur yang sepenuhnya mengandalkan mekanisme perhatian (*self-attention*) untuk menangani ketergantungan global dalam urutan input dan output, tanpa menggunakan RNN [10]. Pendekatan ini menjadi dasar bagi pengembangan BERT (*Bidirectional Encoder Representations from Transformers*), yang memanfaatkan mekanisme *attention bidirectional* untuk memahami konteks dari kedua arah, sehingga unggul dalam berbagai tugas NLP [3]. IndoBERT, sebagai adaptasi BERT untuk bahasa Indonesia, dilatih secara khusus dengan data berbahasa Indonesia, sehingga menunjukkan kinerja superior dalam berbagai tugas NLP lokal [4].

### 3. Sistem yang Dibangun



Gambar 1. Alur Sistem

#### 3.1. Pembuatan Dataset

*Dataset* dalam penelitian ini dibangun berdasarkan data yang dikumpulkan dari dua sumber utama, yaitu pertanyaan yang sering diajukan mahasiswa kepada LAA Prodi S1 Informatika dan bagian FAQ dari situs *web* akademik Telkom University. Pengumpulan data ini menjadi dasar penting dalam menyusun *dataset* yang digunakan untuk keperluan penelitian ini. Dari pengumpulan data tersebut, didapatkan *dataset* dengan 52 *label* dan 1261 *patterns*. *Dataset* ini termasuk dalam klasifikasi *multi-class single-label*, yang berarti setiap *pattern* dikategorikan ke dalam satu dari beberapa kelas yang tersedia, sehingga setiap *input* hanya memiliki satu *label* atau kategori yang paling relevan.

#### 3.2. Mapping Dictionaries dan Splitting Data

Pembuatan *mapping dictionaries* mempermudah pengolahan *label* dalam model dengan membuat kamus yang memetakan *label* ke ID numerik dan sebaliknya. *Splitting data* membagi *dataset* menjadi data latih dan data uji dengan rasio 75:25, yaitu 75% untuk data latih dan 25% untuk data uji. Ini bertujuan agar model dapat dilatih dan diuji secara terpisah, sehingga kinerjanya dapat dievaluasi dengan lebih akurat.

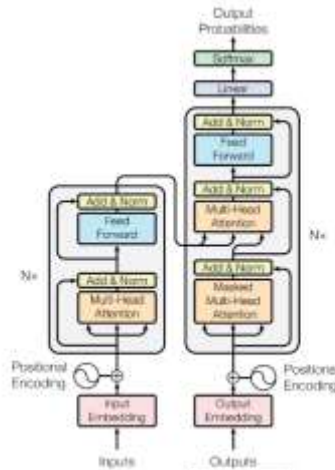
#### 3.3. Load Pre-Trained Model

Pada tahap *load pre-trained model*, dilakukan pemanggilan model pra-terlatih yaitu IndoBERT. Model ini dipilih karena IndoBERT adalah model BERT yang telah dilatih secara khusus untuk bahasa Indonesia, sehingga diharapkan mampu memberikan kinerja yang lebih baik dalam memahami dan memproses teks dalam bahasa tersebut. Selain itu, juga diinisialisasi *tokenizer* dari IndoBERT untuk mengubah teks menjadi token yang akan digunakan dalam proses tokenisasi data latih dan data uji.

#### 3.4. Tokenisasi dan DataLoader

Pada tahap ini, data pelatihan dan pengujian ditokenisasi menggunakan *tokenizer* IndoBERT untuk mengubah teks menjadi token yang dapat diproses oleh model. Selanjutnya, *DataLoader* digunakan untuk menyiapkan data pelatihan dan pengujian dalam format yang siap digunakan oleh model BERT, dengan mengonversi *encoding* data menjadi *tensor* dan memasangkannya dengan *label*, sehingga mempermudah iterasi data selama pelatihan dan pengujian.

### 3.5. Pelatihan Model



**Gambar 2.** Arsitektur *Transformer*

Pada tahap pelatihan model ini, digunakan model pra-trlatih IndoBERT yang mengandalkan arsitektur *Transformer* yang telah dimuat sebelumnya, *Transformer* merupakan model *deep learning* dengan mekanisme *attention* untuk menangkap hubungan antar kata dalam teks. Arsitekturnya terdiri dari *encoder* dan *decoder* dengan lapisan *self-attention* dan *feed-forward*. Pada tahap ini pengaturan *hyperparameter* akan menjadi fokus utama, terutama nilai *epoch* dan *batch size*. Hal ini dilakukan untuk mencapai kinerja model yang paling optimal tanpa mengalami *overfitting*, sekaligus meningkatkan efisiensi selama pelatihan.

### 3.6. Evaluasi Model

Pada tahap terakhir, evaluasi dilakukan untuk menilai performa model dan menentukan dampak dari berbagai konfigurasi *hyperparameter* yang telah diterapkan selama pelatihan. Kinerja model dievaluasi dengan mengukur *Accuracy*, *macro precision*, *macro recall*, dan *macro F1* untuk mendapatkan gambaran menyeluruh tentang kemampuan model dalam menjawab pertanyaan dengan benar dan relevan dalam konteks layanan akademik yang diimplementasikan. Berikut adalah definisi dan rumus-rumus yang digunakan untuk mengukur performa model:

#### 3.6.1. Accuracy

*Accuracy* mengukur proporsi prediksi yang benar dari semua prediksi yang dibuat oleh model. *Accuracy* menghitung rasio prediksi benar terhadap total jumlah prediksi yang dilakukan [11].

$$Accuracy = \frac{\sum_i m_{i,i}}{\sum_{(i,j)} m_{i,j}} \quad (1)$$

Di mana  $m_{i,i}$  adalah jumlah prediksi yang benar untuk kelas  $i$  dan  $m_{i,j}$  adalah jumlah total prediksi untuk kelas  $i$  (baik yang benar maupun salah).

#### 3.6.2. Macro Precision

*Macro precision* adalah rata-rata aritmetika dari *precision* untuk setiap kelas. *Precision* untuk kelas  $i$  mengukur seberapa baik model mengklasifikasikan contoh sebagai kelas  $i$  ketika benar-benar kelas  $i$  [11].

$$MacP = \frac{1}{n} \sum_i P_i \quad (2)$$

### 3.6.3. Macro Recall

*Macro recall* adalah rata-rata aritmetika dari *recall* untuk setiap kelas. *Recall* untuk kelas  $i$  mengukur kemampuan model untuk mendeteksi contoh-contoh dari kelas tersebut [11].

$$MacR = \frac{1}{n} \sum_i R_i \quad (3)$$

### 3.6.4. Macro F1

*Macro F1* adalah rata-rata aritmetika dari *F1* untuk setiap kelas. *F1* adalah rata-rata harmonis dari *precision* dan *recall* untuk kelas tertentu [11].

$$MacF1 = \frac{1}{n} \sum_i F1_i \quad (4)$$

## 4. Evaluasi

### 4.1 Hasil Pengujian

#### 4.1.1. Pengujian Pertama: Variasi *Batch Size* Selama Pelatihan

Pengujian pertama melibatkan perubahan ukuran *batch* selama pelatihan. Ukuran *batch* yang bervariasi, mulai dari 16 hingga 256, digunakan untuk mengeksplorasi bagaimana jumlah data yang diproses dalam satu iterasi dapat mempengaruhi hasil akhir model.

**Tabel 1.** Hasil Pengujian dengan Perubahan *Batch Size*

<i>Batch Size</i>	Data	<i>Evaluation loss</i>	<i>Evaluation Accuracy</i>	<i>Evaluation F1</i>	<i>Evaluation Precision</i>	<i>Evaluation Recall</i>	Waktu (Menit)
16	<i>Train</i>	0.056263	0.984127	0.984770	0.990504	0.986264	04:00
	<i>Test</i>	0.105033	0.971519	0.960539	0.963897	0.962157	
32	<i>Train</i>	0.009193	0.995767	0.996250	0.996584	0.996337	02:45
	<i>Test</i>	0.141980	0.974684	0.964435	0.966163	0.967308	
64	<i>Train</i>	0.012106	0.996825	0.997239	0.997596	0.997253	02:06
	<i>Test</i>	0.145954	0.974684	0.964299	0.966918	0.966964	
128	<i>Train</i>	0.015791	0.996825	0.997239	0.997596	0.997253	01:47
	<i>Test</i>	0.104953	0.977848	0.970998	0.973329	0.973031	
256	<i>Train</i>	0.038954	0.996825	0.997239	0.997596	0.997253	01:47
	<i>Test</i>	0.120434	0.968354	0.954554	0.957303	0.956708	

Berdasarkan Tabel 1, terlihat bahwa *batch size* 128 memberikan hasil terbaik, sehingga akan digunakan dalam pengujian berikutnya.

#### 4.1.2. Pengujian Kedua: Variasi Ukuran *Epoch* Selama Pelatihan

Pengujian kedua melibatkan perubahan jumlah *epoch*, yaitu jumlah siklus penuh pelatihan yang dilalui model, diuji dengan variasi dari 50 hingga 150. Melalui pengujian ini, dieksplorasi seberapa banyak iterasi yang diperlukan untuk model dapat mencapai kinerja optimal.

**Tabel 2.** Hasil Pengujian dengan Perubahan Jumlah *Epoch*

<i>Epoch</i>	Data	<i>Evaluation loss</i>	<i>Evaluation Accuracy</i>	<i>Evaluation F1</i>	<i>Evaluation Precision</i>	<i>Evaluation Recall</i>	Waktu (Menit)
50	<i>Train</i>	0.014260	0.996825	0.997239	0.997596	0.997253	01:52
	<i>Test</i>	0.115179	0.968354	0.957209	0.960966	0.960371	
75	<i>Train</i>	0.008931	0.996825	0.997239	0.997596	0.997253	02:53
	<i>Test</i>	0.138604	0.971519	0.960154	0.963713	0.962775	
100	<i>Train</i>	0.007142	0.996825	0.997239	0.997596	0.997253	03:49
	<i>Test</i>	0.136666	0.971519	0.961408	0.965545	0.964217	



125	<i>Train</i>	0.006424	0.996825	0.997239	0.997596	0.997253	04:44
	<i>Test</i>	0.134758	0.974684	0.964299	0.966918	0.966964	
150	<i>Train</i>	0.005843	0.996825	0.997251	0.997290	0.997253	05:37
	<i>Test</i>	0.161996	0.971519	0.960010	0.961699	0.963385	

Berdasarkan Tabel 2, terlihat bahwa *epoch* 125 memberikan hasil terbaik, sehingga akan digunakan dalam pengujian berikutnya.

#### 4.1.3. Pengujian Ketiga: Hasil Prediksi

Pengujian ketiga dilakukan dengan memberikan dua tipe pertanyaan untuk mengevaluasi kemampuannya dalam memberikan respons yang akurat. Tipe pertama adalah pertanyaan dengan kata kunci yang hanya terkait dengan satu *label*, yang menguji kemampuan *chatbot* dalam mengenali dan merespons pertanyaan spesifik dengan informasi yang relevan. Tipe kedua adalah pertanyaan dengan kata kunci yang muncul pada beberapa *label*, yang menguji kemampuan *chatbot* dalam menangani ambiguitas dan memberikan respons yang paling sesuai berdasarkan konteks. Pengujian ini bertujuan untuk memastikan bahwa *chatbot* mampu menangani berbagai jenis pertanyaan secara efektif.

**Tabel 3.** Pertanyaan dengan kata kunci yang hanya ada pada 1 *label*

No	Pengujian	Detail Pengujian	Sesuai / Tidak Sesuai
1	Input yang diberikan	Halo, Selamat pagi	Sesuai
	Label yang dituju	<i>Greetings</i>	
	Hasil Prediksi	<i>Greetings</i>	
2	Input yang diberikan	Nama kamu siapa kalau boleh tau?	Sesuai
	Label yang dituju	<i>Name</i>	
	Hasil Prediksi	<i>Name</i>	
3	Input yang diberikan	Gimana sih cara <i>cumlaude</i> ?	Sesuai
	Label yang dituju	Predikat <i>Cumlaude</i>	
	Hasil Prediksi	Predikat <i>Cumlaude</i>	
4	Input yang diberikan	Ada info praktikum ngga ya?	Sesuai
	Label yang dituju	Praktikum	
	Hasil Prediksi	Praktikum	
5	Input yang diberikan	Ada info soal wisuda ngga ya?	Sesuai
	Label yang dituju	Wisuda	
	Hasil Prediksi	Wisuda	
6	Input yang diberikan	Kamu punya nomer-nomer dosen ngga?	Sesuai
	Label yang dituju	Kontak Dosen	
	Hasil Prediksi	Kontak Dosen	
7	Input yang diberikan	Fotoku blom diacc di basila, gmn dong?	Sesuai
	Label yang dituju	Basila	
	Hasil Prediksi	Basila	
8	Input yang diberikan	Dosenku telat, aku harus gimana ya?	Sesuai
	Label yang dituju	Dosen Terlambat	
	Hasil Prediksi	Dosen Terlambar	
9	Input yang diberikan	Gimana cara join grup Whatsapp kelas?	Sesuai
	Label yang dituju	Grup Kelas	
	Hasil Prediksi	Grup Kelas	
10	Input yang diberikan	Okede, terimakasih	Sesuai
	Label yang dituju	<i>Goodbye</i>	
	Hasil Prediksi	<i>Goodbye</i>	

**Tabel 4.** Pertanyaan dengan kata kunci yang ada pada beberapa *label*

No	Pengujian	Detail Pengujian	Sesuai / Tidak Sesuai
1	Input yang diberikan	Yudisium	Sesuai
	Label yang dituju	Yudisium	
	Hasil Prediksi	Yudisium	

2	Input yang diberikan	Apa ajasih syarat buat yudisium?	Sesuai
	Label yang dituju	Persyaratan Yudisium	
	Hasil Prediksi	Persyaratan Yudisium	
3	Input yang diberikan	Kapan saya boleh daftar yudisium?	Sesuai
	Label yang dituju	Waktu Pendaftaran Yudisium	
	Hasil Prediksi	Waktu Pendaftaran Yudisium	
4	Input yang diberikan	Kerja praktek	Tidak Sesuai
	Label yang dituju	Kerja Praktek	
	Hasil Prediksi	Waktu Pelaksanaan Kerja Praktek	
5	Input yang diberikan	Apa itu kerja praktek?	Tidak Sesuai
	Label yang dituju	Kerja Praktek	
	Hasil Prediksi	Waktu Pelaksanaan Kerja Praktek	
6	Input yang diberikan	Adakah panduan kerja praktek?	Sesuai
	Label yang dituju	Kerja Praktek	
	Hasil Prediksi	Kerja Praktek	
7	Input yang diberikan	Kapan aku bisa melakukan kerja praktek?	Sesuai
	Label yang dituju	Waktu Pelaksanaan Kerja Praktek	
	Hasil Prediksi	Waktu Pelaksanaan Kerja Praktek	
8	Input yang diberikan	<i>Similarity</i>	Tidak Sesuai
	Label yang dituju	Pengajuan <i>Similarity</i>	
	Hasil Prediksi	Status <i>Similarity Rejected</i>	
9	Input yang diberikan	Cara cek <i>similarity</i> gimana sih?	Sesuai
	Label yang dituju	Pengajuan <i>Similarity</i>	
	Hasil Prediksi	Pengajuan <i>Similarity</i>	
10	Input yang diberikan	Skor similarity saya berapa ya?	Tidak Sesuai
	Label yang dituju	Hasil Cek <i>Similarity</i>	
	Hasil Prediksi	Batas Maksimum <i>Similarity</i>	

#### 4.2 Analisis Hasil Pengujian

##### 4.2.1. Analisis Pengujian Pertama: Variasi *Batch Size* Selama Pelatihan

Dalam analisis pengujian variasi *batch size*, terlihat bahwa *batch size* yang lebih besar, seperti 128, memberikan keseimbangan yang optimal antara waktu pelatihan dan akurasi pada *data test*. Meskipun *batch size* 16 menunjukkan akurasi dan metrik evaluasi yang sangat baik pada *data train*, hasil pada *data test* sedikit lebih rendah dibandingkan dengan *batch size* yang lebih besar. Sebaliknya, *batch size* 256 menghasilkan waktu pelatihan yang paling cepat, namun dengan penurunan kinerja yang signifikan pada *data test*. Hal ini menunjukkan bahwa jumlah data yang diproses dalam satu iterasi secara langsung mempengaruhi hasil akhir model, di mana *batch size* yang terlalu kecil atau terlalu besar dapat menyebabkan ketidakseimbangan antara performa pada *data train* dan *data test*. Oleh karena itu, *batch size* 128 dipilih sebagai konfigurasi terbaik.

##### 4.2.2. Analisis Pengujian Kedua: Variasi *Epoch* Selama Pelatihan

Pada pengujian variasi jumlah *epoch*, ditemukan bahwa peningkatan jumlah *epoch* dari 50 hingga 150 mempengaruhi performa model secara signifikan. Jumlah *epoch* yang lebih tinggi, seperti 150, memang meningkatkan waktu pelatihan namun menunjukkan tanda-tanda *overfitting* pada data test, seperti terlihat dari peningkatan *evaluation loss* meskipun akurasi tetap stabil. Di sisi lain, model dengan 50 *epoch* menunjukkan keseimbangan yang baik antara akurasi dan generalisasi, namun belum mencapai performa optimal dibandingkan dengan *epoch* yang lebih tinggi. Jumlah iterasi yang dilalui model mempengaruhi hasil akhir dengan memberikan model lebih banyak kesempatan untuk belajar, namun juga meningkatkan risiko *overfitting* jika terlalu banyak *epoch* digunakan. Oleh karena itu, jumlah *epoch* 125 dipilih sebagai konfigurasi yang paling tepat, karena memberikan hasil yang optimal.

##### 4.2.3. Analisis Pengujian Ketiga: Hasil Prediksi

Pada pengujian hasil prediksi, *chatbot* menunjukkan kinerja yang baik dalam menjawab pertanyaan dengan kata kunci yang hanya terkait dengan satu *label*, di mana semua respons yang diberikan sesuai. Namun, pada pertanyaan dengan kata kunci yang muncul pada beberapa *label*, ditemukan beberapa kesalahan, khususnya pada topik "kerja praktek" dan "*similarity*". Hal ini terjadi karena ketidakkonsistenan dalam pembuatan *patterns* di *dataset*, sehingga menyebabkan *chatbot* memberikan respons yang tidak tepat, seperti salah mengarahkan pertanyaan tentang "kerja praktek" ke "Waktu Pelaksanaan Kerja Praktek." Demikian pula, untuk pertanyaan

terkait *similarity*, *chatbot* gagal memahami konteks pertanyaan pengguna, yang menunjukkan adanya kebutuhan untuk menambah dan memperbaiki *patterns* agar dapat menangani ambiguitas secara lebih efektif.

## 5. Kesimpulan

Dari analisis yang telah dilakukan, disimpulkan bahwa tujuan tentang pembangunan *chatbot* berbasis IndoBERT yang dapat menyajikan informasi akademik secara tepat waktu dan kontekstual telah tercapai dengan konfigurasi *batch size* adalah 128 dan jumlah *epoch* 125. Konfigurasi tersebut memberikan keseimbangan terbaik antara akurasi dan waktu pelatihan, serta mengurangi risiko *overfitting*. Namun, pada pengujian hasil prediksi menunjukkan bahwa *chatbot* masih kesulitan dalam menangani pertanyaan dengan kata kunci yang terdapat pada beberapa *label*, hal ini terjadi karena ketidakonsistenan dalam pembuatan *patterns* pada *dataset*. Untuk penelitian selanjutnya, disarankan untuk memperbaiki dan menambah variasi *patterns* pada *dataset* agar *chatbot* dapat menangani pertanyaan yang ambigu dengan lebih akurat.

## Daftar Pustaka

- [1] M. Qalimaturrahmah and D. B. Santoso, "Aplikasi Layanan dan Informasi Akademik Berbasis Chatbot Telegram Menggunakan Natural Language Processing," *J. JTIK (Jurnal Teknol. Inf. dan Komunikasi)*, vol. 8, no. 2, pp. 434–443, 2024, doi: 10.35870/jtik.v8i2.1887.
- [2] A. L. Maitri and J. Sutopo, "Rancangan Bangun Chatbot Sebagai Pusat Informasi Lembaga Kursus Dan Pelatihan Menggunakan Pendekatan Natural Language Processing," *Eprints.Uty.Ac.Id*, pp. 1–9, 2019, [Online]. Available: <http://eprints.uty.ac.id/>
- [3] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," *NAACL HLT 2019 - 2019 Conf. North Am. Chapter Assoc. Comput. Linguist. Hum. Lang. Technol. - Proc. Conf.*, vol. 1, no. Mlm, pp. 4171–4186, 2019.
- [4] F. Koto, A. Rahimi, J. H. Lau, and T. Baldwin, "IndoLEM and IndoBERT: A Benchmark Dataset and Pre-trained Language Model for Indonesian NLP," *COLING 2020 - 28th Int. Conf. Comput. Linguist. Proc. Conf.*, pp. 757–770, 2020, doi: 10.18653/v1/2020.coling-main.66.
- [5] I. Hafidz *et al.*, "Chatbot Model Development Using BERT for West Sumatra Halal Tourism Information," *Halal Res.*, vol. 4, no. 2, pp. 117–131, 2024.
- [6] D. Wulandari and J. S. Wibowo, "Implementasi Chatbot Menggunakan Framework Rasa Untuk Layanan Informasi Wisata Di Kota Pati," *INTECOMS J. Inf. Technol. Comput. Sci.*, vol. 6, no. 2, pp. 794–801, 2023, doi: 10.31539/intecom.s.v6i2.7107.
- [7] N. P. V. D. Saraswati, N. Yudistira, and P. P. Adikara, "Analisis Sentimen terhadap Perundungan Siber pada Twitter menggunakan Algoritma Bidirectional Encoder Representations from Transformer (BERT)," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 7, no. 2, pp. 909–916, 2023, [Online]. Available: <https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/12345>
- [8] D. W. Harahap and L. Fitria, "Aplikasi Chatbot Berbasis Web Menggunakan Metode Dialogflow," *J-ICOM - J. Inform. dan Teknol. Komput.*, vol. 1, no. 1, pp. 6–13, 2020, doi: 10.33059/j-icom.v1i1.2796.
- [9] V. R. Prasetyo, N. Benarkah, and V. J. Chrisintha, "Implementasi Natural Language Processing Dalam Pembuatan Chatbot Pada Program Information Technology Universitas Surabaya," *Teknika*, vol. 10, no. 2, pp. 114–121, 2021, doi: 10.34148/teknika.v10i2.370.
- [10] A. Vaswani *et al.*, "Attention is all you need," *Adv. Neural Inf. Process. Syst.*, vol. 2017-Decem, no. Nips, pp. 5999–6009, 2017.
- [11] J. Opitz, "From Bias and Prevalence to Macro F1, Kappa, and MCC: A structured overview of metrics for multi-class evaluation," 2022.