

Pembuatan Chatbot pada Website Persatuan Gizi Kabupaten Karawang menggunakan IndoBERT

1st Adnan Ega Maulana
Fakultas Teknik Informatika
Telkom University Purwokerto
Purwokerto, Indonesia
adnangandul@student.telkomuniversity.ac.id

2nd Yesy Diah Rosita, S.Kom., M.Kom
Fakultas Teknik Informatika
Telkom University Purwokerto
Purwokerto, Indonesia
yesydr@telkomuniversity.ac.id

3rd Dian Kartika Sari, S.Si., M.Pd.
Fakultas Teknik Informatika
Telkom University Purwokerto
Purwokerto, Indonesia
dianks@telkomuniversity.ac.id

Abstrak — PERSAGI (Persatuan Gizi) Kabupaten Karawang memiliki *website* yang digunakan sebagai layanan informasi kesehatan gizi. Namun, penyebaran informasi kurang efektif karena permintaan informasi yang banyak dan kompleks. Permasalahan seperti ini dapat diselesaikan salah satunya menggunakan chatbot. Tujuannya untuk mengembangkan chatbot berbasis metode *Bidirectional Encoder Representations from Transformers* (BERT) untuk mendukung penyebaran informasi di *website* PERSAGI Kabupaten Karawang secara lebih efektif, dan fleksibel. Dalam proses pembangunan *chatbot* peneliti menggunakan metode *Bidirectional Encoder Representations from Transformers* (BERT) khususnya IndoBERT. BERT merupakan model besar dari kumpulan data atau *corpus* Bahasa Indonesia yang diambil dari Wikipedia, dan sumber informasi lainnya. Dalam proses pengembangan, dataset yang digunakan berjumlah 1.864 data yang dikumpulkan melalui wawancara, observasi, studi literatur, dan tinjauan pustaka. Hasil penelitian menunjukkan bahwa chatbot yang dikembangkan mampu mencapai performa yang baik dengan *validation loss* sebesar 0,57, akurasi 87,67%, *f1-score* 84,64%, presisi 85,35%, dan *recall* 86,30%. Hasil ini menunjukkan bahwa metode IndoBERT efektif dalam membangun chatbot yang dapat memenuhi kebutuhan informasi pengunjung *website* secara cepat dan akurat.

Kata kunci— *Chatbot*, *Natural Language Processing*, *Bidirectional Encoder Representations from Transformers*.

I. PENDAHULUAN

Di era digitalisasi, kebutuhan akan informasi yang cepat dan akurat, terutama di bidang kesehatan dan gizi, semakin meningkat. Masyarakat membutuhkan akses mudah untuk mendapatkan informasi terkait gizi, namun layanan langsung sering terkendala oleh masalah efisiensi dan keterbatasan sumber daya, khususnya di daerah seperti Kabupaten Karawang. Persatuan Gizi Kabupaten Karawang (PERSAGI), sebagai lembaga yang berperan penting dalam penyebaran informasi gizi, telah mengembangkan *website* yang menyediakan berbagai layanan, seperti konsultasi gizi dan kalkulator Indeks Massa Tubuh (IMT) untuk semua usia. Namun, menurut Ibu Lelyana Indriasari, A.Md.Gz, sekretaris

Persatuan Gizi Kabupaten Karawang, layanan ini masih belum optimal.

Ibu Lelyana menyatakan bahwa dibutuhkan sistem yang dapat memberikan informasi kesehatan secara cepat, fleksibel, dan dapat diakses 24 jam oleh masyarakat. Penyampaian informasi secara manual dinilai tidak efektif dan tidak mampu menangani permintaan secara terus-menerus. Oleh karena itu, diperlukan solusi teknologi yang dapat mengatasi kendala ini. Salah satu solusi yang diusulkan adalah pengembangan *chatbot* berbasis *Artificial Intelligence* (AI) atau kecerdasan buatan. AI, khususnya cabang *Deep Learning* dengan pendekatan *Natural Language Processing* (NLP), dapat digunakan untuk membuat *chatbot* yang mampu memahami dan merespons pertanyaan pengguna secara alami [1].

Dalam *Deep Learning*, arsitektur *Transformers* menjadi salah satu teknologi kunci dalam NLP. *Transformers* adalah model *neural network* yang menggunakan mekanisme *self-attention* untuk memproses urutan kata, sehingga dapat memahami konteks kalimat dengan lebih baik [2]. Salah satu model populer yang menggunakan arsitektur ini adalah *Bidirectional Encoder Representations from Transformers* (BERT). BERT bekerja dengan memeriksa setiap kata dalam kalimat dari kedua arah (kanan dan kiri), sehingga mampu menangkap konteks secara lebih mendalam [3]. Dengan memanfaatkan BERT, *chatbot* dapat dirancang untuk memberikan informasi gizi yang akurat dan responsif.

Penelitian ini bertujuan untuk menghasilkan sistem *chatbot* yang dapat membantu kinerja PERSAGI dalam penyebaran informasi. Proses pembangunan model BERT akan diuji untuk menghasilkan tingkat akurasi, presisi, *recall*, dan *f1-score* yang baik untuk memahami pertanyaan dan memberikan jawaban yang sesuai.

II. KAJIAN TEORI

A. *Chatbot*

Chatbot merupakan suatu teknologi dalam bidang komunikasi berbasis kecerdasan buatan (AI) yang dapat mensimulasikan percakapan manusia dan komputer dalam bahasa alami. *Chatbot* menggunakan teknik kecerdasan buatan seperti *Natural Language Processing* (NLP) [4].

Teknologi ini juga dikenal sebagai asisten digital yang mampu memahami permintaan pengguna, memproses informasi, dan memberikan tanggapan yang sesuai dengan cepat dan relevan [5].

B. Natural Language Processing

Natural language processing (NLP) atau pemrosesan bahasa alami adalah salah satu cabang ilmu *artificial intelligence* khususnya dalam bidang *linguistic computational*. Tujuan NLP adalah dapat berinteraksi antara mesin dan manusia menggunakan bahasa alami manusia [6]. NLP memiliki tahapan utama seperti pra-pemrosesan, pemrosesan teks, analisis teks.



Gambar 1 Alur kerja NLP

Pada Gambar 1, alur kerja dari NLP ditampilkan, dimulai dengan pengumpulan data teks yang akan digunakan dalam analisis NLP. Pemrosesan teks dilakukan dengan memuat data yang telah diformat sebagai JSON dan memasukkannya ke dalam daftar yang disesuaikan berdasarkan tag. Ekstraksi fitur (*feature engineering*) melibatkan pengambilan fitur dari data mentah dan mengubahnya menjadi format yang sesuai untuk model pembelajaran mesin. Pada tahap pemodelan, model mesin dilatih menggunakan data yang telah di-vektorisasi, di mana model mempelajari pola-pola dalam data latihan dan menyesuaikan parameter internalnya untuk meningkatkan performa. Pada tahap evaluasi, model diuji dan dievaluasi untuk menilai kualitas dan kinerjanya [7].

C. Transformers

Transformers atau transformers-based model adalah arsitektur pemodelan data yang digunakan dalam tugas NLP. Transformers diperkenalkan oleh Vaswani et al pada tahun 2018. Transformers memiliki 2 bagian utama yaitu *encoder* dan *decoder*. *Encoder* berfungsi untuk Mengambil input, memprosesnya, dan menghasilkan representasi konteksnya. Sedangkan *decoder* berfungsi untuk mengambil output dari *encoder* dan menghasilkan output akhir, seperti kalimat dalam bahasa target pada kasus penerjemahan [8]. Arsitektur transformers adalah tipe *neural network* berbasis *self-attention* untuk memproses inputan dalam bentuk urutan kata (*sequence*), sehingga dapat memahami konteks kata dalam kalimat lebih efektif. Model ini juga melahirkan berbagai model NLP seperti salah satunya adalah BERT (*Bidirectional Encoder Representations from Transformers*).

D. Bidirectional Encoder Representations from transformers (BERT)

BERT atau *Bidirectional Encoder Representations from Transformer* adalah salah satu model yang berdiri di dalam arsitektur *transformers* yang dirancang khusus untuk tugas NLP. BERT dikenalkan oleh Devlin et al dan dikembangkan oleh Google pada tahun 2018 [9]. Arsitektur dalam BERT termasuk bagian dari *encoder transformers* yang terdiri dari lapisan *multi-head self-attention* dan *feed-forward neural networks* yang membantu model memahami hubungan antara

kata-kata dalam teks input. Dengan pendekatan *bidirectional* maka setiap kata dalam inputan kalimat diperiksa dari sisi kanan dan kirim untuk menangkap konteks dalam kalimat [3].

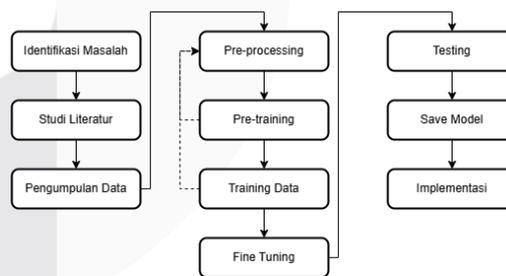
Pada BERT terdapat proses tokenisasi untuk mengubah kalimat menjadi sebuah token token kecil. Proses tokenisasi ini dilakukan di awal sebelum proses *pre-training* sebelum teks di lanjutkan ke proses *modelling*. Dalam BERT memiliki proses khusus dalam tokenisasi yaitu *Wordpiece Tokenizer* [10]. Proses tokenisasi pada BERT pertama diberi token khusus pada setiap kalimatnya seperti:

- [CLS]: adalah tanda untuk awal kalimat.
- [CLS]: adalah tanda untuk awal kalimat.
- [SEP]: adalah tanda untuk akhir kalimat.
- [PAD]: adalah tanda untuk menambahkan token kosong untuk mengisi panjang token yang sudah ditentukan [25].
- [##]: adalah tanda untuk memisahkan kata hubung dengan kata baku nya, tanda ## menandakan bahwa kalimat tersebut adalah bagian dari kalimat sebelumnya.

III. METODE

Penelitian ini dirancang dengan mengikuti prosedur sistematis untuk mencapai tujuan yang telah ditetapkan. Langkah-langkah penelitian dimulai dengan identifikasi masalah, dilanjutkan dengan studi literatur, pengumpulan data, analisis data, dan diakhiri dengan penyusunan kesimpulan.

Sumber data utama dalam penelitian ini adalah menggunakan teknik wawancara dengan narasumber Ibu Lelyana Indriasari A.Md.Gz. Data diperoleh melalui Teknik wawancara, tinjauan pustaka, observasi, dan *prompting ChatGPT*. Metode yang digunakan dalam penelitian ini adalah kualitatif dengan pendekatan eksperimental. Analisis data dilakukan menggunakan *Deep Learning* dengan produk *Transformers* seperti *Bidirectional Encoder Representations from Transformers* (BERT).



Gambar 2 Diagram alir penelitian

IV. HASIL DAN PEMBAHASAN

Data yang didapatkan merupakan informasi mengenai kesehatan gizi dan dinas kesehatan Kabupaten Karawang, data yang telah dikumpulkan akan dari berbagai pertanyaan akan dijadikan sebagai jawaban dari chatbot. Data tersebut diambil dari hasil wawancara dan dengan sekretaris dinas kesehatan Kabupaten Karawang yaitu Ibu Lelyana Indriasari, A.Md.Gz. Dataset juga dikumpulkan menggunakan metode lain seperti, observasi, studi literatur, dan tinjauan Pustaka.

A. Pengumpulan data

Berdasarkan hasil dari pengumpulan data tersebut, tercatat 1864 data yang berhasil terkumpul yang terdiri dari 71 category. Data yang terkumpul akan menjadi acuan sebagai pembuatan dataset yang terdiri dari intent, category, question, dan answer ke dalam format *xlsx*. kemudian akan di proses oleh bahasa pemrograman python melalui Google Colab. Dataset yang sudah disiapkan diproses ke dalam variabel *category*, *questions*, *answers* menggunakan library *Pandas*, untuk memudahkan dalam pengelolaan data setiap *category* akan diberi label untuk mengidentifikasi setiap *category*.

Category	Questions	Answers	Labels
0	salam	hi	hi
1	salam	hai	hai, apakah ada yang bisa saya bantu
2	salam	hallo	hallo, ada yang bisa saya bantu
3	salam	assalamualaikum	waalaikumsalam
4	salam	assalamualaikum wr wb	waalaikumsalam warahmatullah wabarakatuh

Gambar 3 Dataset pandas

B. Pre-processing

a. Remove Punctuation

Remove punctuation untuk menghilangkan tanda baca, simbol, dan hal yang tidak penting. Pada penelitian ini menggunakan library string dalam proses *remove punctuation*. Hasil dari proses *remove punctuation* dapat dilihat pada tabel 1.

Tabel 1. Hasil proses *remove punctuation*

No	Sebelum	Sesudah
1	apa faktor-faktor yang mempengaruhi gizi	apa faktorfaktor yang mempengaruhi gizi buruk.

b. Lowercase

Lowercase untuk membuat kata menjadi bentuk huruf kecil. Hasil dari proses *lowercase* dapat dilihat pada tabel 2.

Tabel 2. Hasil proses *lowercase*

No	Sebelum	Sesudah
1	Penyembuhan	penyembuhan

C. Pre-training

Selanjutnya adalah proses *pre-training* dataset, sebelum itu dataset yang sudah di jadikan *dataframe* dibagi menjadi data *training* dan data *testing*. Dalam penelitian ini menggunakan library *sklearn* dan membagi data *training* dan data *testing* dengan rasio 80:20. Dataset yang sebelumnya berjumlah 1864 dibagi menjadi data *train* berjumlah 1491 dan data *test* 373. Dalam membagi dataset menggunakan parameter tambahan seperti *test_size* adalah rasio perbandingan antara data *train* dan data *test*, dan

random_state adalah untuk mengacak data yang akan masuk ke *train* data dan *test* data.

```
data questions sebelum split : 1864
data answers sebelum split : 1864
data question sesudah split : 1491
data answers sesudah split : 373
```

Gambar 4. Hasil split data

Pada penelitian ini menggunakan model *indobert* "*indobenchmark/indobert=large-pl*" dengan *max_len* 256 sebagai panjang maksimal token yang akan digunakan sebagai parameter. Selanjutnya, membuat *tokenizer* menggunakan *BertTokenizer.from_pretrained(model_name, max_length=max_len)* dan model dengan *BertForSequenceClassification.from_pretrained(model_name, num_label=num_label, id2label=id2label, label2id=label2id)*.

Selanjutnya, membuat *encoding* menggunakan *tokenizer* dengan parameter *data*, *truncation*, *padding*, dan *max_length*. Dalam proses *encoding* terdapat proses tokenisasi dari BERT, tokenisasi adalah proses mengubah kalimat menjadi sebuah token kecil.

Tabel 3. Hasil Encoding

Data	Apa yang dimaksud dengan gizi
Token	["[CLS]", "apa", "yang", "dimaksud", "dengan", "gizi", "[PAD]", "[PAD]", "[PAD]", ... "[SEP]"]
Token id	["2", "387", "34", "2143", "79", "4709", "3", "0", "0", "0", ...]
Attention Mask	[1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, ...]

Pada hasil diatas kalimat akan melalui proses tokenisasi dengan dijadikan token kecil yang memiliki tanda khusus seperti [CLS] sebagai awal dari kalimat, [PAD] sebagai padding untuk menyamakan dari panjang maksimal token, dan [SEP] sebagai tanda akhir dari kalimat. Selanjutnya, token akan di beri id untuk mengidentifikasikan tiap kata, dan attention mask sebagai penanda token valid yang didefinisikan dengan 1 dan token padding didefinisikan dengan 0.

D. Training data

Pada *training* model akan di definisikan *compute matrices* untuk mengukur performa model menggunakan library *precision_recall_fscore_support* dan *accuracy_score* dari *sklearn*. Parameter yang digunakan adalah *accuracy*, *precision*, *recall*, dan *f1-score*. Selanjutnya, membuat *training argument* untuk mengatur control dari proses training model. Parameter yang digunakan seperti :

1. *output_dir* : Menentukan direktori untuk menyimpan model yang telah dilatih.
2. *do_train* : Menentukan apakah proses *training* (pelatihan) akan dilakukan.
3. *do_eval* : Menentukan apakah proses evaluasi pada data validasi akan dilakukan selama *training*.
4. *num_train_epochs* : Jumlah *epoch* (putaran penuh data *training* melalui model).

5. *per_device_train_batch_size* : Ukuran *batch* untuk proses *training* (per perangkat, seperti GPU atau CPU).
6. *per_device_eval_batch_size* : Ukuran *batch* untuk proses evaluasi (per perangkat).
7. *warmup_steps* : jumlah langkah pemanasan untuk meningkatkan *learning rate* secara bertahap sebelum mencapai nilai maksimum.
8. *weight_decay* : Menentukan tingkat *weight decay*, yang merupakan teknik regulasi untuk mencegah *overfitting*.
9. *logging_dir* : Direktori untuk menyimpan log selama proses *training*.
10. *logging_strategy* : Menentukan kapan *logging* dilakukan.
11. *evaluation_strategy* : Menentukan kapan evaluasi dilakukan.
12. *save_strategy* : Menentukan kapan model disimpan selama *training*
13. *load_best_model_at_end* : Menentukan apakah model terbaik (berdasarkan metrik tertentu) akan dimuat setelah proses *training* selesai.
14. *Optim* : Menentukan *optimizer* yang digunakan untuk memperbarui bobot model selama *training*.
15. *learning_rate* : Tingkat pembelajaran (*learning rate*), yang menentukan seberapa besar bobot model diperbarui pada setiap langkah.
16. *metric_for_best_model* : Menentukan metrik yang digunakan untuk memilih model terbaik.
17. *greater_is_better* : Menentukan apakah metrik model terbaik lebih besar atau lebih kecil lebih baik.

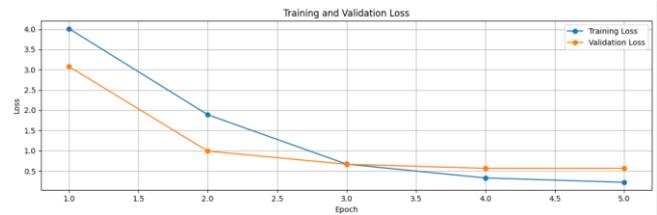
Selanjutnya, adalah proses *training* menggunakan *library Trainer* dari *transformers*, menggunakan parameter model yang digunakan, *argument*, *train encoding*, *test encoding* dan fungsi *compute matrices*.

Tabel 4. Hasil Training

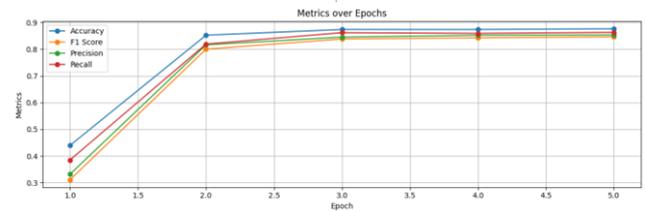
Epo ch	Train ing loss	Valida tion loss	accur acy	F1- score	precis ion	recall
1	4.013 900	3.0775 27	0.439 678	0.311 270	0.331 874	0.385 118
2	1.889 100	0.9906 08	0.852 547	0.800 234	0.816 487	0.819 398
3	0.668 900	0.6658 92	0.873 995	0.838 396	0.845 434	0.862 012
4	0.327 600	0.5655 07	0.873 995	0.842 658	0.852 108	0.859 240
5	0.219 500	0.5651 68	0.876 676	0.846 409	0.853 511	0.862 972

Pada tabel 4. menunjukkan bahwa model yang dilatih dengan 5 epoch mendapatkan hasil *training loss* sebesar 0.219500, *validation loss* sebesar 0.565168, *accuracy* sebesar 0.876676 atau 87,67%, *f1-score* 0.846409 atau 84, 64%, *precision* sebesar 0.853511 atau 85,35%, dan *recall* sebesar 0.862972 atau 86, 30%. Dari hasil *training* menunjukkan peningkatan *accuracy* dari tiap *epoch*. Gambar 5. dan

Gambar 6. adalah grafik *loss* dan hasil dari proses *training model*.



Tabel 5. Grafik loss



Tabel 6. Grafik Accuracy, F1-score, precision, recall

E. Testing

Langkah awal untuk mendefinisikan model dan *library* yang dibutuhkan. *Sentencetransformer*("all-MiniLM-l6-v2") dari *transformers* digunakan untuk mencari kemiripan dari input dan dataset yang sudah di *train*. *BertForSequenceClassification.from_pretrained* () digunakan untuk *load* model yang sudah di *train*. *BertTokenizerFast.from_pretrained*() untuk *load tokenizer* dari model yang sudah di *train*. *Pipeline*() digunakan untuk menentukan tugas khusus dalam hal ini adalah *text-classification*. Dalam proses klasifikasi ini akan menghasilkan label yang di prediksi dan *score*, label akan digunakan untuk filter dataset sehingga akan menghasilkan dataframe berdasarkan hasil label tersebut. Hasil *score* menunjukkan tingkat performa prediksi berdasarkan input, semakin tinggi hasil *score* maka dapat dinyatakan bahwa model yang di *train* memiliki performa yang bagus. Hasil dari proses prediksi label bisa dilihat pada tabel 5.

Tabel 7. Hasil klasifikasi label

Input	Label Prediksi	Score	Keterangan
Apa yang dimaksud dengan gizi	gizi	0.83629268 40782166 atau 83%	Label yang di prediksi benar sesuai dengan dataset yang ada

Selanjutnya, data label yang di prediksi akan di proses untuk mengambil seluruh data yang memiliki label tersebut. Proses ini adalah *filtering* data agar pada proses *similarity* hanya memproses data dari label yang di prediksi saja. Dalam hal ini membuat performa *chatbot* lebih baik karena tidak mengecek semua data dalam dataset, hanya data yang difilter yang akan di cek validasi *similarity* nya.

Tabel 8. Hasil Similarity

Input			
apa yang dimaksud dengan gizi			
No.	Questions dataset	Score Similarity	Answer dataset
1	apa yang dimaksud dengan gizi	1.0	gizi adalah zat-zat yang diperlukan tubuh untuk pertumbuhan, perkembangan, dan menjaga fungsi tubuh.
2	bagaimana cara memperbaiki gizi sejak dini	0.5282564	memberikan ASI eksklusif selama 6 bulan, setelah itu tambahkan makanan pendamping asi (MPASI) dengan gizi seimbang
3	apa saja fungsi utama dari gizi bagi tubuh	0.6823064	Fungsi utama gizi adalah sebagai sumber energi, mendukung pertumbuhan dan perbaikan jaringan tubuh, serta mengatur proses metabolisme dan fungsi organ tubuh.
4	mengapa gizi penting bagi kesehatan	0.62400234	gizi sangat penting karena berperan dalam menjaga kesehatan tubuh, mendukung pertumbuhan, dan menjaga daya tahan tubuh.
5	cara memperbaiki gizi	0.46863914	konsumsi makanan bergizi seimbang, atur pola makan teratur, hindari kebiasaan tidak sehat

Selanjutnya diambil *score similarity* yang tertinggi menggunakan *argmax*, dengan hasil tersebut dapat digunakan untuk mencari *index* dari *dataframe* hasil *similarity* untuk mengambil nilai answer atau jawaban dari pertanyaan yang terpilih.

V. KESIMPULAN

Berdasarkan hasil penelitian, dapat disimpulkan bahwa pembuatan chatbot pada website Persatuan Gizi Kabupaten

Karawang menggunakan *Bidirectional Encoder Representations from Transformers* melibatkan proses pengelolaan dataset, *remove punctuation*, *lowercase*, *pre-training*, *training model*, dan *testing* untuk memahami pertanyaan dan memberikan jawaban yang dengan tingkat akurasi sebesar 87,67%, *loss* sebesar 0,2%, presisi sebesar 85,35%, *recall* sebesar 86,30%, dan *F1-score* sebesar 84,64%. Dari hasil kemiripan pertanyaan dan data pertanyaan yang dilatih menjadi parameter yang cocok untuk mendapatkan jawaban yang akurat.

REFERENSI

- [1] M. Siahaan, C. Harsana Jasa, K. Anderson, M. V. Rosiana, S. Lim, and W. Yudianto, "Penerapan Artificial Intelligence (AI) Terhadap Seorang Penyandang Disabilitas Tunanetra," 2020.
- [2] Z. Liang, Z. Liang, Y. Zheng, B. Liang, and L. Zheng, "Data analysis and visualization platform design for batteries using flask-based python web service," *World Electric Vehicle Journal*, vol. 12, no. 4, Dec. 2021, doi: 10.3390/wevj12040187.
- [3] A. Babu and S. B. Boddu, "BERT-Based Medical Chatbot: Enhancing Healthcare Communication through Natural Language Understanding," *Exploratory Research in Clinical and Social Pharmacy*, vol. 13, Mar. 2024, doi: 10.1016/j.rcsop.2024.100419.
- [4] T. A. Zuraiyah, D. K. Utami, and D. Herlambang, "Implementasi Chatbot Pada Pendaftaran Mahasiswa Baru Menggunakan Recurrent Neural Network," *Jurnal Ilmiah Teknologi dan Rekayasa*, vol. 24, no. 2, pp. 91–101, 2019, doi: 10.35760/tr.2019.v24i2.2388.
- [5] Nuzul Hikmah, Dyah Ariyanti, and Ferry Agus Pratama, "Implementasi Chatbot Sebagai Virtual Assistant di Universitas Panca Marga Probolinggo menggunakan Metode TF-IDF," *JTIM: Jurnal Teknologi Informasi dan Multimedia*, vol. 4, no. 2, pp. 133–148, Aug. 2022, doi: 10.35746/jtim.v4i2.225.
- [6] "View of Penerapan Natural Language Processing Dalam Aplikasi Chatbot Sebagai Media Pencarian Informasi Dengan Menggunakan React (Studi Kasus_ Institut Bisnis Dan Informatika Kwik Kian Gie)".
- [7] M. Rizky Suherlan and A. Pambudi, "Jurnal Informatika Terpadu Ummibot Sebagai Media Layanan Informasi Penerimaan Mahasiswa Baru Universitas Muhammadiyah Sukabumi," *Jurnal Informatika Terpadu*, vol. 9, no. 2, pp. 82–91, 2023, [Online]. Available: <https://journal.nurulfikri.ac.id/index.php/JIT>
- [8] R. Sudheesh *et al.*, "Analyzing Sentiments Regarding ChatGPT Using Novel BERT: A Machine Learning Approach," *Information (Switzerland)*, vol. 14, no. 9, Sep. 2023, doi: 10.3390/info14090474.
- [9] D. E. Taşar *et al.*, "Performance Trade-Off for Bert Based Multi-Domain Multilingual Chatbot Architectures," *Journal of Artificial Intelligence and Data Science (JAIDA)*, vol. 1, no. 2, pp. 144–149, 2021, [Online]. Available: <https://jaida.ikcu.edu.tr>

[10] M. N. Hidayat and R. Pramudita, "Analisis Sentimen Terhadap Pembelajaran Secara Daring Pasca Pandemi Covid-19 Menggunakan Metode IndoBERT," *Information Management for Educators and Professionals*, vol. 8, no. 2, pp. 161–170, 2023.

[11] "Implementasi NLP Pada Chatbot Layanan Akademik Dengan Algoritma Bert Implementation Of NLP On Academic Service Chatbot With Bertalgorithm."

