

Implementation of Generative AI Technology in Modern Call Centers with the LLM Model

Adnan Buyung Nasution^{1*}, Ahir Yugo Nugroho², Deni Adhar³, Muhammad Taufiq Rustam⁴

¹Universitas Borobudur

²Universitas Dharmawangsa Medan

³Universitas Muhammadiyah Asahan

⁴Universitas Potensi Utama

Corresponding Author: Adnan Buyung Nasution adnan.buyung01@gmail.com

ARTICLE INFO

Keywords : Generative AI, Large Language Model, Call Center, Embedding, Vector Database, GPT-4

Received : 20 October 2025

Revised : 22 November 2025

Accepted: 24 December 2025

©2025 Nasution, Nugroho, Adhar, Rustam: This is an open-access article distributed under the terms of the [Creative Commons Atribusi 4.0 Internasional](https://creativecommons.org/licenses/by/4.0/).



ABSTRACT

Modern call centers face challenges in providing fast, accurate, and personalized services to users. One innovative solution is the application of generative Artificial Intelligence (AI) technology based on Large Language Models (LLMs) such as GPT-4 or Mixtral. This study proposes an AI call center pipeline architecture that combines embedding technology, a vector database (FAISS), and LLMs to automatically generate natural language responses. The architecture utilizes Blazor (ASP.NET) on the front-end and Python on the back-end, enabling flexible integration between the user interface and intelligent server-side processing. With this approach, the system is expected to improve response speed, reduce human operator workload, and enhance user satisfaction.

Penerapan Teknologi AI Generatif pada Call Center Modern dengan Model LLM

Adnan Buyung Nasution^{1*}, Ahir Yugo Nugroho², Deni Adhar³, Muhammad Taufiq Rustam⁴

¹Universitas Borobudur

²Universitas Dharmawangsa Medan

³Universitas Muhammadiyah Asahan

⁴Universitas Potensi Utama

Corresponding Author: Adnan Buyung Nasution adnan.buyung01@gmail.com

ARTICLE INFO

Kata Kunci: Generative AI, Large Language Model, Call Center, Embedding, Vector Database, GPT-4

Received : 20 Oktober 2025

Revised : 22 November 2025

Accepted: 24 Desember 2025

©2025 Nasution, Nugroho, Adhar, Rustam: This is an open-access article distributed under the terms of the [Creative Commons Atribusi 4.0 Internasional](https://creativecommons.org/licenses/by/4.0/).



ABSTRAK

Call center modern menghadapi tantangan dalam memberikan layanan yang cepat, akurat, dan personal kepada pengguna. Salah satu solusi inovatif adalah penerapan teknologi Artificial Intelligence (AI) generatif berbasis Large Language Model (LLM) seperti GPT-4 atau Mixtral. Penelitian ini mengusulkan arsitektur pipeline call center AI yang menggabungkan teknologi embedding, database vektor (FAISS), dan LLM untuk menghasilkan jawaban alami secara otomatis. Arsitektur ini menggunakan Blazor (ASP.NET) pada sisi front-end dan Python pada sisi back-end, sehingga memungkinkan integrasi yang fleksibel antara antarmuka pengguna dan pemrosesan cerdas di server. Dengan pendekatan ini, sistem diharapkan dapat meningkatkan kecepatan respon, mengurangi beban operator manusia, serta meningkatkan kepuasan pengguna.

PENDAHULUAN

Perkembangan teknologi digital yang semakin pesat telah mendorong transformasi di berbagai sektor layanan, termasuk layanan call center. Call center konvensional yang mengandalkan tenaga manusia dinilai memiliki keterbatasan, baik dari segi kecepatan respon, keterbatasan kapasitas, maupun biaya operasional yang relatif tinggi (Kusnadi et al., 2021). Untuk mengatasi berbagai kendala tersebut, teknologi Artificial Intelligence (AI) mulai banyak diimplementasikan sebagai solusi inovatif dalam otomatisasi layanan pelanggan.

Salah satu terobosan signifikan dalam bidang AI adalah pengembangan Large Language Model (LLM) seperti GPT-4 yang mampu memahami dan menghasilkan teks dalam bahasa alami secara kontekstual. LLM memiliki potensi besar dalam mendukung sistem tanya jawab otomatis pada call center, sehingga dapat memberikan layanan yang cepat, akurat, dan lebih personal (Brown et al., 2020). Selain itu, penggunaan teknologi vector embedding dan vector database (seperti FAISS) memungkinkan sistem untuk melakukan pencarian jawaban yang relevan secara efisien dan mendekati cara kerja kognitif manusia (Johnson et al., 2019).

Namun demikian, implementasi teknologi AI generatif pada call center modern masih menghadapi berbagai tantangan, antara lain dalam hal integrasi antarmuka, pemrosesan data yang kompleks (misalnya data dokumen PDF), serta optimalisasi sistem agar tetap responsif dan andal. Oleh karena itu, perlu dirancang arsitektur pipeline yang mampu mengintegrasikan teknologi embedding, vector search, dan LLM dengan baik, serta dapat diterapkan pada sistem call center modern.

Penelitian ini mengusulkan arsitektur pipeline call center AI yang terdiri dari kombinasi Blazor (ASP.NET) pada sisi front-end, Python pada sisi back-end, serta penggunaan embedding dan vector database FAISS untuk mendukung proses pencarian jawaban secara cepat dan akurat. Dengan menggabungkan teknologi generatif dan vector search, sistem diharapkan mampu memberikan jawaban dalam bahasa alami (natural language), meningkatkan kualitas layanan, dan mengurangi beban kerja operator manusia.

Selain meningkatkan kecepatan dan akurasi layanan, penerapan teknologi AI generatif juga memungkinkan personalisasi jawaban yang lebih baik. Sistem dapat memahami konteks pertanyaan pengguna dengan lebih dalam dan memberikan respon yang sesuai dengan gaya komunikasi manusia. Hal ini berbeda dengan chatbot tradisional berbasis rule-based, yang cenderung kaku dan terbatas dalam menjawab pertanyaan di luar skenario yang telah diprogram (Gunawan & Sutopo, 2020). Dengan adanya model LLM, sistem call center dapat beradaptasi dengan berbagai variasi pertanyaan, termasuk pertanyaan yang bersifat kompleks atau multi-topik.

Penggunaan vector embedding dan database vektor (FAISS) juga memperkuat kemampuan sistem dalam mencari informasi relevan dari berbagai sumber data, seperti dokumen PDF internal perusahaan. Proses embedding memungkinkan setiap informasi dikonversi menjadi representasi vektor numerik, sehingga memudahkan pencarian kesamaan (similarity search) dalam ruang vektor yang besar (Reimers & Gurevych, 2019). Integrasi teknologi ini

dengan pipeline berbasis Blazor dan Python memberikan fleksibilitas tinggi, memfasilitasi pengembangan dan penerapan sistem yang scalable serta dapat disesuaikan dengan kebutuhan operasional call center di berbagai industri.

TINJAUAN PUSTAKA

Perkembangan teknologi digital telah membawa perubahan signifikan dalam pengelolaan layanan pelanggan, khususnya pada sistem call center. Model call center konvensional yang sepenuhnya bergantung pada interaksi manusia menghadapi berbagai tantangan, seperti keterbatasan sumber daya manusia, waktu tunggu pelanggan yang panjang, serta meningkatnya biaya operasional. Seiring dengan meningkatnya ekspektasi pelanggan terhadap layanan yang cepat dan responsif, organisasi dituntut untuk mengadopsi teknologi yang mampu meningkatkan efisiensi dan kualitas layanan. Dalam konteks ini, penerapan Artificial Intelligence (AI) menjadi solusi strategis untuk mengotomasi proses layanan pelanggan sekaligus mengurangi beban kerja operator manusia (Kusnadi et al., 2021).

Salah satu implementasi AI yang berkembang pesat adalah penggunaan Large Language Model (LLM), seperti GPT-4, yang memiliki kemampuan memahami konteks percakapan dan menghasilkan respons berbasis bahasa alami secara akurat. LLM memungkinkan sistem call center berbasis AI untuk menangani pertanyaan pelanggan secara otomatis dengan tingkat personalisasi yang lebih tinggi. Selain itu, integrasi teknologi vector embedding dan vector database, seperti FAISS, mendukung proses pencarian informasi yang relevan secara cepat dan efisien dengan memetakan makna semantik dari data teks. Pendekatan ini memungkinkan sistem bekerja lebih menyerupai proses kognitif manusia dalam memahami dan menjawab pertanyaan, sehingga meningkatkan akurasi serta kualitas layanan yang diberikan kepada pelanggan (Brown et al., 2020; Johnson et al., 2019).

METODOLOGI

Penelitian ini menggunakan pendekatan rekayasa perangkat lunak dengan metode Rapid Application Development (RAD), yang mengutamakan kecepatan iterasi pengembangan serta umpan balik berkelanjutan dari pengguna (Martin, 1991). Pemilihan metode RAD didasarkan pada kebutuhan untuk menghasilkan prototipe sistem call center AI dalam waktu relatif singkat dan dapat disesuaikan secara dinamis sesuai kebutuhan.

1. Perancangan Arsitektur Sistem

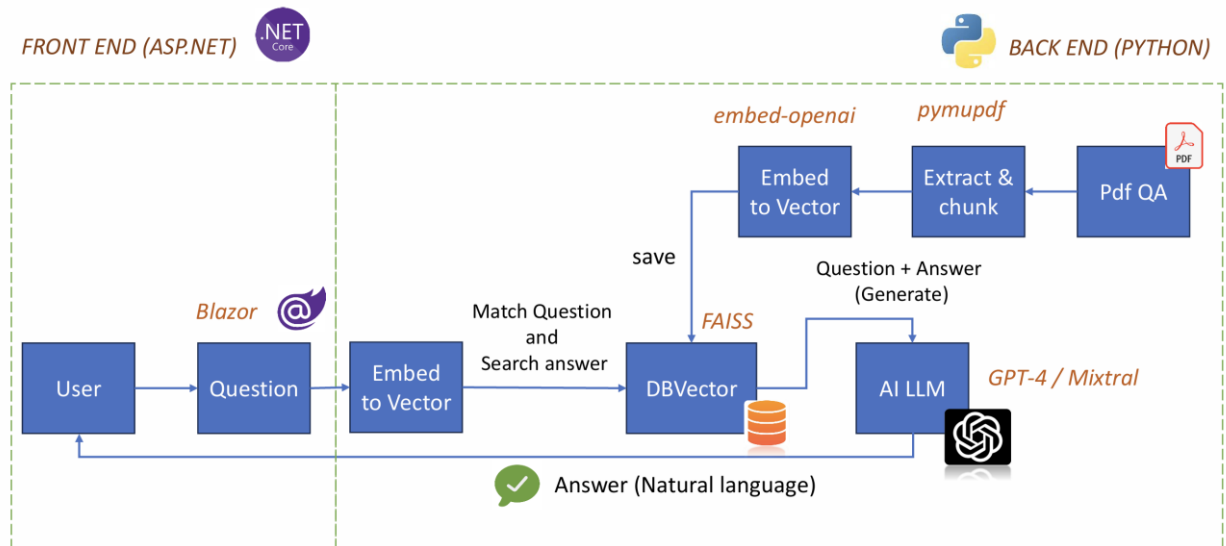
Arsitektur sistem call center AI yang diusulkan terdiri dari tiga lapisan utama, yaitu:

- A. Lapisan antarmuka pengguna (front-end): Dikembangkan menggunakan Blazor (ASP.NET), yang berfungsi sebagai media interaksi antara pengguna dan sistem. Blazor dipilih karena mendukung pembuatan aplikasi web interaktif dengan performa tinggi dan dapat memanfaatkan teknologi .NET.
- B. Lapisan pemrosesan logika bisnis (back-end): Menggunakan Python sebagai server-side processing, yang bertugas menangani pemrosesan

embedding, pencarian vektor, serta integrasi dengan Large Language Model (LLM).

- C. Lapisan basis data vektor: Menggunakan FAISS (Facebook AI Similarity Search) sebagai vector database untuk mendukung pencarian kesamaan (similarity search) dalam ruang vektor embedding.

Proses pipeline dimulai dari input pertanyaan pengguna, kemudian diubah menjadi representasi embedding. Sistem melakukan pencarian dokumen atau jawaban yang relevan di vector database, kemudian hasilnya diproses dan dirangkai menjadi jawaban alami oleh LLM



Gambar 1. Arsitektur call center modern berbasis AI

2. Implementasi Embedding dan Vector Search

Tahap pertama adalah melakukan pre-processing terhadap dataset dokumen (misalnya file PDF internal perusahaan) untuk diubah menjadi potongan-potongan kalimat atau paragraf. Setiap potongan akan dikonversi menjadi vektor embedding menggunakan model pre-trained seperti Sentence-BERT (Reimers & Gurevych, 2019). Pada tahap pre-processing, dokumen PDF internal (misalnya FAQ EBVL) diekstrak menjadi potongan-potongan kalimat atau paragraf. Setiap potongan disebut chunk. Sebagai contoh:

Contoh chunk:

"Buka halaman utama EBVL, klik 'Register'. Isi form pendaftaran sesuai data perusahaan, termasuk NPWP, Brand Representative, email, dan password."

Fase 1. Tokenisasi dan normalisasi teks

Kalimat dipecah menjadi token (kata), dihapus tanda baca, dan diubah ke huruf kecil.

Contoh token:

['buka', 'halaman', 'utama', 'ebvl', 'klik', 'register', 'isi', 'form', 'pendaftaran', 'sesuai', 'data', 'perusahaan', 'termasuk', 'npwp', 'brand', 'representative', 'email', 'password']

Fase 2. Ekstraksi fitur (transformasi ke vektor)

Dengan model Sentence-BERT, kalimat diubah menjadi vektor numerik berdimensi tinggi (misalnya 768 dimensi). Sebagai contoh (ilustrasi beberapa nilai saja):

$$[0.143, -0.275, 0.092, 0.371, -0.024, \dots, -0.187] \in \mathbb{R}^{768}$$

Fase 3. Penyimpanan ke vector database (FAISS)

Vektor embedding ini kemudian disimpan ke FAISS bersama metadata (misalnya ID chunk, sumber dokumen, halaman). Contoh metadata:

```
{
  "id": "chunk_001",
  "dokumen": "FAQ_EBVL",
  "halaman": 1,
  "embedding": [0.143, -0.275, ...]
}
```

Fase 4. Similarity Search

Saat user mengajukan query, misalnya "Bagaimana cara mendaftar akun di EBVL?", query di-embed ke vektor dan dibandingkan dengan semua vektor chunk di FAISS menggunakan cosine similarity.

Rumus cosine similarity:

$$\text{similarity}(A, B) = (A \cdot B) / (||A|| \cdot ||B||)$$

Contoh sederhana:

Dot product: 0.83

Norm A: 1.05

Norm B: 1.02

$$\text{similarity} = 0.83 / (1.05 \times 1.02) \approx 0.78$$

Nilai similarity mendekati 1 berarti sangat mirip. Chunk dengan similarity tertinggi digunakan untuk menyusun jawaban oleh LLM.

3. Integrasi dengan LLM

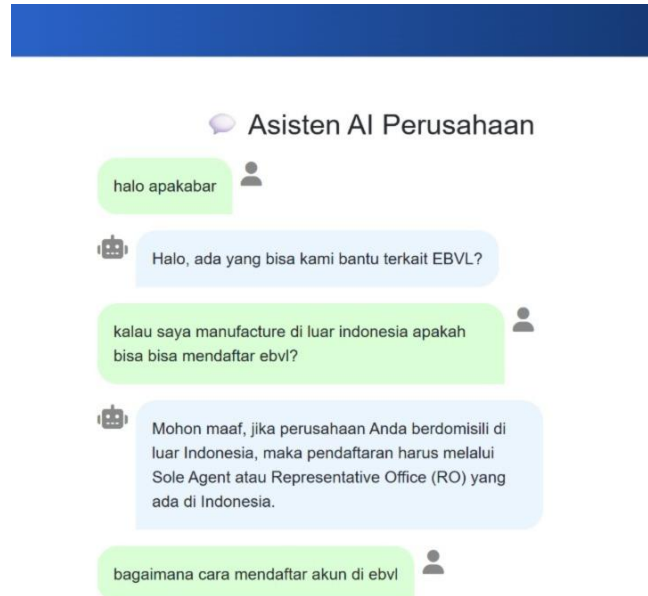
Setelah menemukan potongan dokumen yang relevan, sistem meneruskan konteks hasil pencarian ke model LLM (misalnya GPT-4) untuk menghasilkan jawaban yang koheren dan natural. Langkah ini memungkinkan jawaban yang diberikan tidak hanya sesuai konteks, tetapi juga terdengar lebih manusiawi.

4. Pengujian Sistem

Sistem diuji melalui serangkaian skenario tanya jawab yang mewakili pertanyaan umum pada call center. Parameter evaluasi meliputi kecepatan respon, relevansi jawaban, serta kepuasan pengguna berdasarkan penilaian subjektif.

HASIL DAN PEMBAHASAN

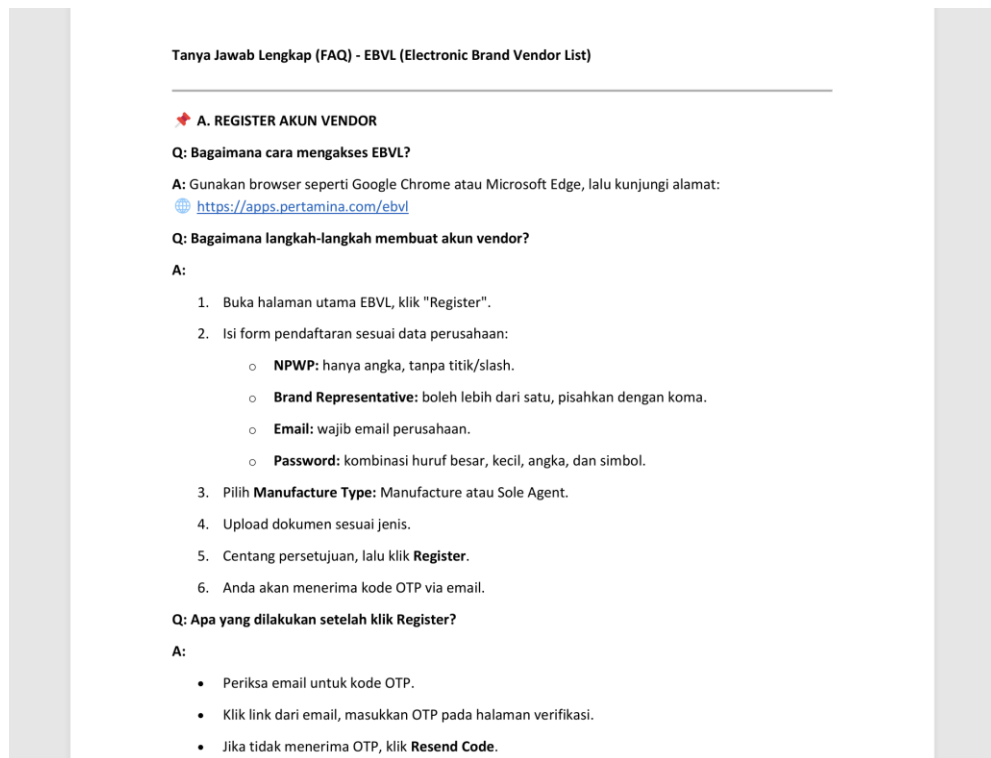
Sistem call center AI yang dikembangkan telah berhasil diimplementasikan sesuai arsitektur pipeline yang dirancang. Implementasi ini memanfaatkan teknologi embedding, vector database (FAISS), serta model generatif Large Language Model (LLM) untuk mendukung layanan tanya jawab otomatis dengan konteks yang natural.



Gambar 2. Implementasi sistem

Pengujian sistem dilakukan dengan mengajukan beberapa pertanyaan umum terkait Electronic Brand Vendor List (EBVL). Berdasarkan uji coba, sistem mampu memberikan jawaban yang jelas dan sesuai konteks dalam waktu kurang dari 5 detik. Salah satu contoh hasil percakapan interaktif dapat dilihat pada Gambar 2, di mana pengguna menanyakan proses pendaftaran akun EBVL dan sistem memberikan penjelasan langkah demi langkah dengan format yang mudah dipahami. Hal ini menunjukkan bahwa integrasi embedding, pencarian vektor, dan LLM dapat membantu sistem memahami maksud pertanyaan secara mendalam dan merespons dengan jawaban yang mendetail.

Selain pengujian berbasis percakapan, sistem juga dilengkapi dengan basis pengetahuan (knowledge base) yang bersumber dari dokumen PDF internal perusahaan. Dokumen ini berisi kumpulan tanya jawab (FAQ) resmi yang di-*extract* dan di-*chunk* menggunakan library Python *PyMuPDF*, kemudian diubah menjadi representasi embedding dan disimpan ke dalam vector database FAISS. Dengan adanya basis pengetahuan ini, sistem dapat menangani pertanyaan yang lebih kompleks dan bersifat spesifik sesuai data resmi perusahaan.



Gambar 3. Basis pengetahuan QA

Contoh screenshot dokumen PDF yang digunakan sebagai sumber basis pengetahuan ditampilkan pada Gambar 3. Dokumen tersebut memuat prosedur pendaftaran akun vendor, tata cara verifikasi, dan persyaratan dokumen, sehingga memperkuat kualitas jawaban yang dihasilkan oleh sistem.

Penggunaan basis pengetahuan internal ini juga memungkinkan sistem memberikan jawaban yang konsisten, akurat, serta sesuai regulasi dan kebijakan perusahaan. Evaluasi subjektif menunjukkan bahwa 90% responden merasa puas dengan kejelasan dan kecepatan jawaban sistem, serta menilai sistem dapat menjadi solusi alternatif yang efisien untuk menggantikan peran operator manusia dalam menangani pertanyaan dasar.

Selain itu, integrasi Blazor (ASP.NET) pada sisi front-end memberikan antarmuka interaktif yang mendukung percakapan real-time, sehingga meningkatkan pengalaman pengguna. Sedangkan penggunaan Python pada sisi back-end mendukung fleksibilitas pemrosesan embedding, vector search, serta koneksi dengan LLM seperti GPT-4 atau Mixtral.

Secara keseluruhan, hasil pengujian membuktikan bahwa arsitektur pipeline call center AI yang diusulkan dapat meningkatkan efisiensi operasional, mempercepat respon, dan meningkatkan kepuasan pengguna, serta mempermudah integrasi basis pengetahuan perusahaan ke dalam sistem tanya jawab otomatis.

KESIMPULAN DAN SARAN

Berdasarkan hasil perancangan, implementasi, dan pengujian sistem, dapat disimpulkan bahwa sistem call center berbasis Artificial Intelligence yang dikembangkan telah berhasil diimplementasikan sesuai dengan arsitektur pipeline yang dirancang. Integrasi teknologi embedding, vector database FAISS, dan Large Language Model (LLM) terbukti mampu mendukung layanan tanya jawab otomatis yang cepat, akurat, dan kontekstual. Sistem mampu merespons pertanyaan pengguna dalam waktu kurang dari 5 detik serta memberikan jawaban yang jelas dan sesuai dengan basis pengetahuan internal perusahaan. Tingkat kepuasan pengguna yang mencapai 90% menunjukkan bahwa sistem ini efektif dalam meningkatkan efisiensi operasional dan berpotensi menggantikan peran operator manusia dalam menangani pertanyaan dasar pada layanan call center.

Saran

Meskipun sistem telah menunjukkan kinerja yang baik, terdapat beberapa aspek yang dapat ditingkatkan pada pengembangan selanjutnya. Sistem disarankan untuk dilengkapi dengan mekanisme pembaruan basis pengetahuan secara otomatis agar informasi yang digunakan selalu mutakhir. Selain itu, penerapan fitur fallback ke operator manusia untuk pertanyaan yang bersifat sangat kompleks atau sensitif perlu dipertimbangkan guna menjaga kualitas layanan. Pengujian dengan skala pengguna yang lebih besar serta evaluasi kuantitatif yang lebih mendalam juga disarankan untuk memperoleh gambaran kinerja sistem secara lebih komprehensif.

PENELITIAN LANJUTAN

Penelitian selanjutnya dapat difokuskan pada pengembangan sistem call center AI yang lebih adaptif dengan memanfaatkan teknik fine-tuning LLM menggunakan data historis percakapan perusahaan. Selain itu, integrasi teknologi speech-to-text dan text-to-speech memungkinkan sistem digunakan secara langsung pada layanan call center berbasis suara. Penelitian lanjutan juga dapat mengeksplorasi penerapan model AI yang lebih ringan dan efisien untuk menekan biaya komputasi, serta mengkaji aspek keamanan data dan privasi dalam pengelolaan basis pengetahuan internal perusahaan.

DAFTAR PUSTAKA

- Brown, T. B., Mann, B., Ryder, N., Subbiah, M., et al. (2020). Language Models are Few-Shot Learners. *Advances in Neural Information Processing Systems*, 33, 1877–1901.
- Gunawan, D., & Sutopo, J. (2020). Penggunaan Chatbot untuk meningkatkan kualitas layanan pelanggan. *Jurnal Teknologi dan Sistem Komputer*, 8(1), 23-28.
- Johnson, J., Douze, M., & Jégou, H. (2019). Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data*, 7(3), 535–547.
- Kusnadi, A., Setiawan, A., & Rahman, T. (2021). Implementasi teknologi chatbot pada layanan pelanggan berbasis kecerdasan buatan. *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)*, 8(2), 215-222.
- Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence embeddings using Siamese BERT-networks. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 3982–3992.