



Sistem Tanya-Jawab Berbasis Chatbot Web dan Telegram Tentang Akademik UNAND Menggunakan LLM

Wahyudi¹, Adi Arga Arifnur²

¹Departemen Informatika, Fakultas Teknologi Informasi, Universitas Andalas

²Departemen sistem informasi, Fakultas Teknologi Informasi, Universitas Andalas

¹wahyudi@it.unand.ac.id*, ²adiargaarifnur@it.unand.ac.id

Abstract

Information about Academic topics, particularly derived from Academic Documents, is crucial. There's a need for an information service that can provide answers to questions concerning contemporary Academic matters. One way to achieve this is by leveraging current technologies, such as the chatbot found on Telegram. Utilizing a Telegram chatbot can lead to the creation of a question-answering system about Academic subjects. Therefore, this research aims to implement a Telegram chatbot-based question-answering system focused on Academic information. This study also utilizes the LangChain method and Large Language Models (LLMs) to create an optimal question-answering system. The research successfully demonstrates its effectiveness in providing answers regarding Academic topics, with the results from User Acceptance Test (UAT) and Black Box Testing indicating that the system can deliver accurate information and functions as intended as a question-answering system for Academic content.

Keywords: academic, chatbot, langchain, large language model, Q&A system

Abstrak

Informasi mengenai akademik, terkhusus berasal dari dokumen akademik. Perlunya sebuah layanan informasi yang bisa memberikan jawaban atas pertanyaan seputar akademik kontemporer, salah satunya yaitu memanfaatkan teknologi saat ini seperti chatbot yang ada pada Telegram. Dengan memanfaatkan chatbot pada Telegram, dapat menghasilkan sebuah sistem tanya jawab mengenai akademik. Jadi, penelitian ini bertujuan untuk menerapkan sistem tanya jawab berbasis chatbot Telegram mengenai akademik. Penelitian ini juga memanfaatkan metode langchain dan Large Language Model (LLM) agar dapat menciptakan sistem tanya jawab yang optimal. Penelitian ini berhasil menunjukkan keefektifannya dalam memberikan jawaban mengenai akademik, di mana hasil pengujian User Acceptance Test (UAT) dan Black Box Testing menunjukkan bahwa sistem ini dapat memberikan informasi yang akurat dan juga berfungsi sebagaimana mestinya sebagai sistem tanya jawab tentang akademik.

Kata kunci: akademik, chatbot, langchain, large language model, sistem tanya-jawab

1. Pendahuluan

Lanskap teknologi saat ini telah mengalami kemajuan signifikan, terutama dengan adopsi dan pengembangan teknologi internet yang meluas dalam domain informasi dan penyebaran pengetahuan [1]. Teknologi memfasilitasi komunikasi dan berbagi informasi tanpa hambatan di antara pengguna, terlepas dari lokasi geografis atau batasan waktu, yang sebagian besar disebabkan oleh jangkauan luas platform media sosial [2]. Telegram, sebuah aplikasi pesan instan terkemuka, menawarkan fitur yang patut diperhatikan: chatbot. Chatbot adalah aplikasi yang dirancang untuk berinteraksi dengan pengguna di berbagai platform, umumnya digunakan untuk resolusi pertanyaan otomatis [3]. Selain itu, aplikasi web juga menjadi media vital

yang memungkinkan akses informasi dan interaksi pengguna melalui browser [3].

Dunia akademik, dengan kekayaan disiplin ilmu dan penelitian, mewakili suatu keharusan untuk kemajuan pengetahuan dan pemahaman umat manusia. Berakar pada publikasi ilmiah, jurnal, dan buku teks, informasi akademik memiliki otentisitas yang melekat [4]. Pengetahuan akademik, sebaliknya, berkaitan dengan perkembangan terbaru, teori, dan penemuan dalam berbagai bidang studi, mencakup era pascamodern hingga saat ini. Para sarjana dan peneliti telah memberikan kontribusi signifikan terhadap pengetahuan akademik yang berfungsi sebagai sumber otoritatif untuk interpretasi dan pemahaman isu-isu modern tertentu. Oleh karena itu, terdapat kebutuhan mendesak untuk meningkatkan aksesibilitas dan pemahaman



Lisensi

Lisensi Internasional Creative Commons Attribution-ShareAlike 4.0.

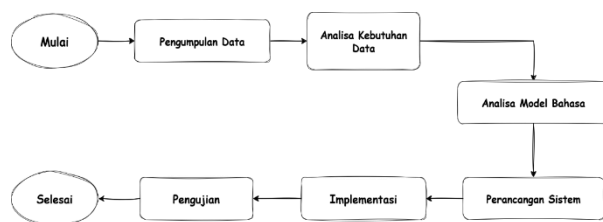
pengetahuan akademik melalui sistem yang mampu menafsirkan dan menanggapi pertanyaan bahasa alami, sehingga menyederhanakan pencarian informasi bagi pengguna [5].

Sistem tanya jawab (QA) menghadirkan solusi yang kuat untuk tantangan ini dan mampu memproses berbagai jenis pertanyaan pengguna. Sistem ini dirancang untuk memberikan respons optimal melalui proses multi-tahap yang melibatkan analisis pertanyaan, pengambilan bagian teks yang relevan, dan ekstraksi jawaban yang potensial. Yang terpenting, sistem QA membutuhkan dua komponen utama: *LangChain* dan *Large Language Models* (LLM) [6].

LLM secara inheren dirancang untuk menghasilkan keluaran yang relevan secara kontekstual dan koheren secara tata bahasa. Model ini memproses masukan tekstual dan selanjutnya menghasilkan keluaran tekstual [7], [8]. Contoh menonjol dari LLM yang mewakili lompatan substansial dalam kecerdasan buatan generatif adalah seri Gemini AI, dengan Gemini 1.5 Pro sebagai iterasi terbaru. Gemini 1.5 Pro dibangun di atas prinsip-prinsip inti pendahulunya sambil menunjukkan kemampuan pemrosesan bahasa alami yang ditingkatkan dan mendukung masukan multimodal, sehingga sangat serbaguna dalam berbagai aplikasi. Meningkatkan keserbagunaannya di berbagai aplikasi [9]. Metodologi LLM menawarkan keuntungan yang jelas dibandingkan dengan pendekatan alternatif, seperti sistem berbasis aturan, karena kapasitasnya yang superior untuk pemahaman, pemrosesan, dan penjelasan, serta kemampuannya untuk menangani pertanyaan kompleks, tidak seperti metode lainnya [10], [11].

2. Metode Penelitian

Berikut adalah tahapan-tahapan yang penulis lakukan untuk menerapkan sistem tanya jawab berbasis web dan chatbot Telegram.



Gambar 1. Alur Tahapan Penelitian

Gambar 1 menunjukkan bahwa penelitian ini dilakukan dengan 4 tahapan, yakni pengumpulan data, analisis dan perancangan, implementasi, dan pengujian. Penelitian ini menghasilkan sebuah chatbot telegram yang dapat memberikan jawaban

2.1 Pengumpulan Data

Data pada pembuatan sistem tanya jawab ini bersumber dari data Peraturan Akademik Universitas Andalas.

Terdapat 22 dokumen peraturan akademik berbentuk pdf yang ditunjukkan pada Tabel 1.

Tabel 1. Daftar dokumen akademik UNAND

No	Judul dokumen	Total Halaman
1	Peraturan Menteri Pendidikan Dan Kebudayaan Republik Indonesia Nomor 3 Tahun 2020	75 Halaman
2	Peraturan Rektor Nomor 7 Tahun 2022	91 Halaman
3	Keputusan Rektor Universitas Andalas Nomor 1844/KPT/R/PTN-BH/UNAND/2022	14 Halaman
4	Peraturan Pemerintah Republik Indonesia Nomor 95 Tahun 2021	60 Halaman
5	Peraturan Rektor Universitas Andalas Nomor 8 Tahun 2024	14 Halaman
6	Peraturan Rektor Universitas Andalas Nomor 10 Tahun 2021	41 Halaman
7	Peraturan Menteri Pendidikan Dan Kebudayaan Republik Indonesia Nomor 3 Tahun 2020	70 Halaman
8	Peraturan Menteri Pendidikan Dan Kebudayaan Republik Indonesia Nomor 14 Tahun 2014	17 Halaman
9	Peraturan Menteri Pendidikan Dan Kebudayaan Republik Indonesia Nomor 35 Tahun 2020	11 Halaman
10	Peraturan Menteri Pendidikan, Kebudayaan, Riset, Dan Teknologi Republik Indonesia Nomor 53 Tahun 2023	45 Halaman
11	Peraturan Menteri Pendidikan, Kebudayaan, Riset, Dan Teknologi Republik Indonesia Nomor 23 Tahun 2023	9 Halaman
12	Peraturan Menteri Pendidikan, Kebudayaan, Riset, Dan Teknologi Republik Indonesia Nomor 44 Tahun 2024	28 Halaman
13	Peraturan Menteri Pendidikan, Kebudayaan, Riset, Dan Teknologi Republik Indonesia Nomor 38 Tahun 2021	8 Halaman

2.2 Analisa Kebutuhan Data

Setelah pengumpulan metode dan data, langkah krusial selanjutnya adalah analisis dan perancangan. Tahap ini bertujuan untuk mengkaji lebih lanjut informasi yang telah diperoleh dan memastikan kesesuaiannya dengan kebutuhan sistem yang akan dibangun. Dengan memahami alur kerja sistem secara menyeluruh pada fase ini, proses implementasi dapat berjalan lebih efisien dan terarah [12].

Dalam penelitian ini, analisis kebutuhan data berfokus pada tiga belas dokumen akademik Universitas Andalas yang tersedia dalam format PDF. Penting untuk dicatat bahwa data ini, pada dasarnya, merupakan citra digital atau gambar. Konsekuensinya, isi tekstual dalam dokumen-dokumen tersebut tidak dapat diproses langsung oleh sistem. Sebagai contoh, saat mencoba menyalin dan menempelkan tulisan dari dokumen ke

editor teks, karakter-karakter tersebut sering kali berubah menjadi simbol-simbol yang tidak bermakna.

Oleh karena itu, diperlukan tahapan pembersihan data (*data cleaning*). Pada tahapan ini, *Optical Character Recognition* (OCR) menjadi alat yang esensial. OCR adalah teknik yang memungkinkan konversi teks dari format gambar menjadi karakter digital yang dapat diolah lebih lanjut oleh sistem [13]. Dengan demikian, pada tahapan ini, penulis akan melakukan proses OCR terhadap publikasi akademik kontemporer karya penulis terkemuka. Hasil dari proses ini akan berupa teks digital yang siap untuk diolah oleh sistem. Seluruh teks yang telah dikonversi ini kemudian akan disimpan kembali dalam format DOCX, memastikan data tersedia dalam bentuk yang dapat dibaca dan diproses oleh sistem tanya jawab yang akan dikembangkan.

2.3 Analisa Model Bahasa

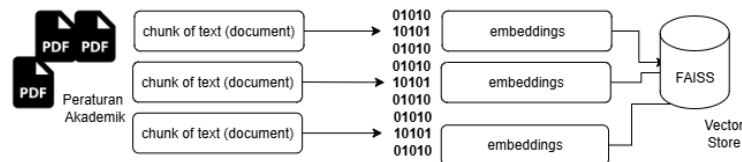
Model bahasa yang akan digunakan dalam penelitian ini adalah *Large Language Model* (LLM). LLM yang akan dipakai menggunakan model Gemini 1.5, yang

merupakan model canggih dari Google dan dapat memakai format percakapan, memungkinkan manusia untuk mengajukan pertanyaan guna mendapatkan informasi [14].

Gemini 1.5 menawarkan kecepatan pencarian jawaban yang lebih cepat dan tingkat halusinasi yang lebih kecil dibandingkan dengan model sebelumnya, artinya bisa mendapatkan jawaban yang lebih akurat. LLM di sini digunakan sebagai model untuk penghasil jawaban yang akan dikirim oleh chatbot kepada pengguna. Chatbot dapat menerima berbagai bentuk pertanyaan dari pengguna, karena pertanyaan dari pengguna itu akan di-embed menggunakan Gemini Embedding 001 dan dikirim ke vector store, bukan dilakukan oleh LLM secara langsung.

2.4 Perancangan Sistem Tanya-Jawab

Perancangan pada sistem ini dibagi menjadi dua tahapan, tahapan pertama yaitu pada saat data dimasukkan, seperti yang ditunjukkan pada Gambar 2.



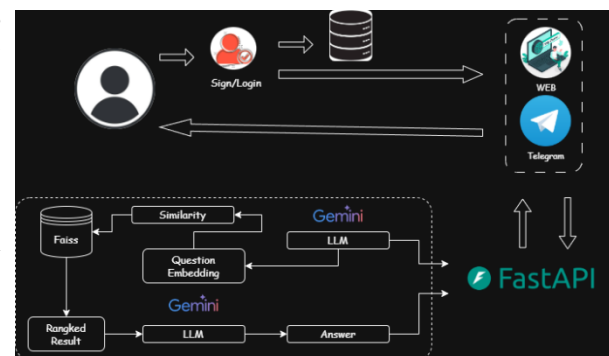
Gambar 2. Alur Input Data

Pada Gambar 2, langkah awal yang dilakukan ketika data sudah dimasukkan adalah *chunk of text* atau pemotongan teks. Pemotongan teks merupakan proses mengubah dokumen menjadi bagian-bagian kecil, yang dibagi agar lebih mudah diproses dan tetap menjaga konteks [15]. Setelah potongan teks dibagi, tahapan selanjutnya adalah embedding.

Proses embedding dilakukan dengan menggunakan Gemini Embedding 001, yang akan mengubah potongan-potongan teks menjadi representasi numerik. Model ini mengubah teks yang tidak terstruktur menjadi daftar angka yang mana angka tersebut mewakili teks tersebut [16]. Data yang sudah melalui tahapan embedding akan dimasukkan ke dalam Knowledge Base. Knowledge Base sendiri adalah tempat dikumpulkannya sejumlah besar informasi yang sudah melalui pelatihan pada kumpulan data teks yang luas. Knowledge Base juga menjadi lebih efisien dalam menghadapi tantangan pemahaman bahasa alami yang kompleks karena memungkinkan LLM untuk menawarkan pendekatan yang telah dilatih sebelumnya [17].

Pada Gambar 3, alur sistem tanya-jawab chatbot menunjukkan serangkaian tahapan yang dilalui setiap kali pesan dimasukkan oleh pengguna, dan seluruh tahapan ini pada akhirnya akan kembali ke pengguna sebagai penerima pesan [18]. Karena knowledge base sudah dibuat, kita bisa melanjutkan ke proses

berikutnya. Pada tahapan selanjutnya, kita akan menggunakan knowledge base sebagai basis data sistem tanya jawab.



Gambar 3. Alur Sistem Tanya-Jawab

Sistem tanya jawab selalu dimulai ketika sistem menerima input dari pengguna, lalu akan terhubung ke API. API di sini bertugas untuk menerima request dari pengguna dan menghubungkannya dengan knowledge base. Dapat dikatakan bahwa API berperan sebagai penghubung antara pengguna dengan basis data [19].

Ketika pengguna bertanya melalui input, pesan input pengguna akan diembed menjadi vektor numerik. Pertanyaan pengguna di-embed menggunakan Gemini Embedding 001. Setelah itu, dilakukan *similarity search* menggunakan vektor numerik tersebut. *Similarity search* digunakan sebagai metode pencarian yang ada di

knowledge base. Yang terjadi di dalam *knowledge base* adalah bahwa hasil *embed* berupa vektor akan dibandingkan dan dicari yang paling mirip dengan vektor pertanyaan pengguna. Vektor yang dibandingkan adalah dokumen yang ada di dalam *vector store*, di mana akan diambil 4 dokumen dengan bagian yang paling mirip dengan pertanyaan dari pengguna. Proses ini menggunakan pustaka FAISS dan mampu mengambil hasil hanya dalam hitungan milidetik [20]. Di dalam *knowledge base* sudah terdapat pertanyaan pengguna dan juga 4 dokumen yang mendekati jawaban pengguna. Maka akan dilakukan *ranked results*, di mana proses ini sangat penting sebelum dokumen dikirim ke LLM.

Setelah dilakukannya *ranked results*, dokumen akan diproses oleh LLM. Di sini tugas LLM adalah membentuk jawaban dari dokumen yang sudah diranking sebelumnya. LLM di sini menggunakan Gemini 1.5 dengan parameter 0,1 yang akan membuat model LLM lebih deterministik dan akurat dalam menjawab pertanyaan teknis. Perlu diingat bahwa di sini tugas LLM adalah sebagai penghasil jawaban dari dokumen yang sudah diranking, dan LLM bukan sebagai pemroses pertanyaan dari pengguna. Pertanyaan dari pengguna murni diproses oleh *embedding* dan juga FAISS sebagai *knowledge base*.

Setelah LLM membentuk respons, respons itu akan dikirimkan ke chatbot Telegram. Respons yang dihasilkan terdiri dari question yang merupakan pertanyaan dari pengguna, answer sebagai jawaban yang dihasilkan, dan source sebagai 22 dokumen sumber dari mana jawaban itu didapatkan.

Response inilah yang akan dikirimkan kepada user. Pada penelitian ini, dikarenakan chatbot menerima *response*, *answer*, dan *source*, hal itu akan dimanfaatkan oleh peneliti sebagai kelebihan. Di sini, user tidak hanya menerima jawaban dari chatbot, tetapi user juga dapat menerima sumber referensi dari jawaban yang dihasilkan. Hal ini bertujuan untuk memberikan transparansi informasi tentang dari mana jawaban chatbot itu didapatkan.

2.5 Implementasi

Tahapan analisis merupakan penentu apakah sistem yang sebelumnya sudah dianalisis dan dirancang dapat berjalan dan digunakan sebagaimana mestinya.

Implementasi Back-End: Peneliti menggunakan Python sebagai bahasa implementasi sistem tanya jawab. Python merupakan bahasa pemrograman *open source* yang memungkinkan pengembangan dan penghematan biaya. Python juga memiliki banyak *library* yang dapat digunakan untuk pembuatan API dan juga chatbot Telegram [21]. Untuk *back-end service chatbot*, API akan diimplementasikan secara langsung menggunakan *library* Python yang relevan untuk penanganan *request* dan *respons* [22].

Implementasi Front-End: Salah satu hal yang ada pada implementasi front-end yaitu UI (*User Interface*). Dalam penelitian ini, peneliti menggunakan salah satu fitur Telegram, yaitu *chatbot*, sebagai tampilan antarmuka dari sistem tanya jawab untuk menghubungkan pengguna dengan sistem. Telegram merupakan aplikasi pesan instan dan dapat menjawab pertanyaan secara otomatis melalui *chatbot*. Aplikasi Telegram banyak disukai karena multiplatform dan ringan. Telegram juga menyediakan dokumentasi yang lengkap untuk pengembangan sehingga pembuatan *chatbot* dapat dilakukan dengan baik [23].

Dan untuk implementasinya pada web, kami menggunakan JavaScript (JS) untuk membangun antarmuka *front-end* sistem tanya jawab. JS merupakan pilihan yang ideal karena kemampuannya dalam menciptakan aplikasi web yang interaktif dan responsif, sehingga pengguna dapat terhubung dengan sistem secara dinamis dan *real-time*. Pemanfaatan JS sebagai *front-end* memungkinkan kami untuk mengembangkan antarmuka yang fleksibel dan multiplatform yang dapat diakses melalui berbagai browser dan perangkat tanpa memerlukan instalasi aplikasi tambahan.

2.6 Pengujian

Pengujian atau evaluasi merupakan salah satu tahapan yang penting dilakukan. Pengujian dilakukan untuk memastikan bahwa sistem tetap terjaga dan berfungsi optimal. Proses ini bertujuan untuk menilai sejauh mana model berhasil diterapkan pada dataset [24]. Pada penelitian ini, evaluasi akan dilakukan dengan cara memberikan kuesioner kepada responden melalui UAT (*User Acceptance Test*). Responden merupakan dosen universitas andalas. Total responden sejumlah 3 orang yang akan diberikan 10 pertanyaan. Untuk menentukan kualitas dari jawaban *chatbot* Akademik UNAND .

Penelitian ini juga akan melakukan pengujian dengan cara *black box testing*, yang merupakan metode pengujian yang dilakukan untuk menyesuaikan antara rancangan dan hasil implementasi sistem, sehingga sistem yang dihasilkan bisa sesuai dengan rancangan dan dapat berjalan sesuai dengan kebutuhan [25].

3. Hasil dan Pembahasan

3.1 Hasil Pengumpulan Data

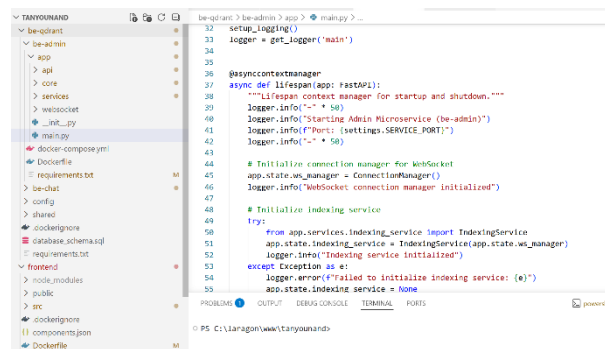
Data pada penelitian ini menggunakan 13 data akademik UNAND. Data tersebut memang berbentuk PDF, namun isinya merupakan gambar sehingga teks di dalamnya tidak dapat langsung diolah. Oleh karena itu, perlu dilakukan tahapan pembersihan data (*data cleaning*), yaitu proses OCR (*Optical Character Recognition*) pada publikasi-publikasi tersebut.

Proses OCR dalam penelitian ini akan memanfaatkan Google Lens sebagai alat OCR gratis. Hasil dari data yang sudah di-OCR akan berupa teks di dalam PDF yang siap untuk diolah oleh sistem.

3.2 Hasil Implementasi

Implementasi yang dilakukan pada penelitian ini dimulai dari pembuatan API yang merupakan model chatbot, lalu Telegram sebagai *interface* dari sistem tanya jawab ini.

Pada bagian *back-end*, sebuah API dibuat menggunakan Python. API ini nantinya akan berfungsi sebagai inti model untuk chatbot Anda. Model yang dibangun dengan Python ini akan menghasilkan data dalam format JSON dengan atribut "question", "answer", dan "source". Data ini kemudian akan dikelola dan dihubungkan secara langsung melalui kode Python yang bertindak sebagai penghubung antara chatbot Telegram dan *knowledge base*. Implementasi *back-end* dapat dilihat pada Gambar 4.



Gambar 4. Output dari API Service

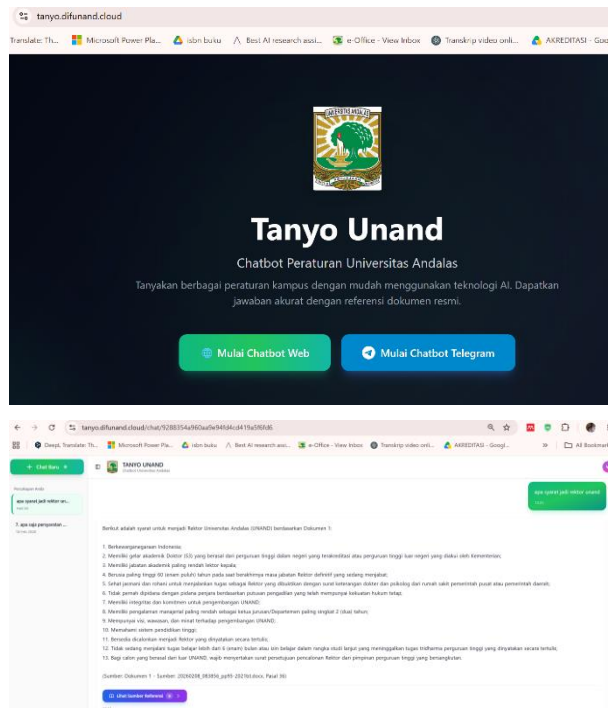
Gambar 4 menampilkan lingkungan pengembangan Visual Studio Code yang fokus pada proses *back-end* untuk chatbot berbasis AI. Pada jendela editor utama, terlihat kode Python yang menginisialisasi API menggunakan framework FastAPI, lengkap dengan definisi judul, deskripsi, dan versi API yang menyoroti perannya sebagai backend untuk chatbot Universitas Andalas. Konfigurasi CORS (*Cross-Origin Resource Sharing*) juga diatur, memungkinkan komunikasi yang aman antara API dan front-end yang mungkin berada di origin berbeda, serta penanganan exception global untuk validasi request guna memastikan respons JSON yang konsisten saat terjadi kesalahan.

Bagian terminal di bawah menunjukkan proses krusial dari proyek ini: skrip Python bernama `real_embeddings_ingest.py` sedang dieksekusi. Output terminal secara jelas mengindikasikan bahwa sistem sedang melakukan processing embedding batch, dengan detail kemajuan seperti "111/127" dan "Generated 10 real embeddings". Hal ini menunjukkan bahwa data sedang diubah menjadi representasi numerik (vektor) secara bertahap, sebuah langkah penting dalam membangun atau memperbarui *knowledge base chatbot* untuk pencarian kemiripan yang efisien.

UI atau User Interface pada sistem ini menggunakan Telegram dan antarmuka berbasis web. Ketika pengguna bertanya melalui chatbot Telegram atau melalui aplikasi web, akan membentuk relasi dari pengguna ke sistem.

Pada saat pertanyaan masuk ke dalam sistem, baik dari chatbot Telegram maupun dari aplikasi web, pesan akan diteruskan kepada FastAPI sebagai service API dari sistem tanya-jawab ini. Ketika chatbot atau antarmuka web mengirim request kepada API, API akan menerimanya lalu mengirimkannya kembali sebagai response, lalu response tersebut akan dikirimkan sebagai jawaban dari chatbot Telegram atau ditampilkan pada aplikasi web kepada pengguna. Hasil dari sistem tanya-jawab dapat dilihat pada Gambar 5.

Implementasi chatbot berbasis web yang ditampilkan dirancang untuk menyediakan informasi terkait Universitas Andalas dengan nama "TANYO UNAND". Pengguna berinteraksi melalui browser web, mengajukan pertanyaan seperti "Apa syarat mahasiswa terkena drop out?" melalui kolom input yang tersedia. Sistem kemudian memproses pertanyaan ini melalui *back-end* API (yang menggunakan FastAPI) yang terintegrasi dengan *Large Language Model* (LLM) dan *vector store* (*knowledge base*) berisi data akademik. Hasilnya adalah jawaban yang detail dan terstruktur mengenai syarat-syarat *drop out* untuk berbagai jenjang program studi yang kemudian ditampilkan secara langsung di antarmuka chat.



Gambar 5. Tampilan Sistem Tanya-Jawab Berbasis Chatbot Telegram

Aspek paling penting dari implementasi ini, yang terlihat jelas pada Gambar 5, adalah penyertaan referensi sumber ("Peraturan Rektor Universitas Andalas Nomor 7 Tahun 2022") yang ditampilkan secara transparan bersama dengan jawaban, yang menegaskan fitur akuntabilitas dan kepercayaan informasi yang dihasilkan chatbot.

Pengujian UAT bertujuan memverifikasi kebenaran isi, relevansi, dan kelengkapan jawaban chatbot terhadap pertanyaan di domain akademik UNAND dengan cara membandingkan langsung jawaban chatbot dengan jawaban referensi yang bersumber dari dokumen/peraturan akademik resmi. Hasil UAT memberikan ukuran kelayakan sistem untuk digunakan oleh sivitas akademika.

Pengujian UAT dilakukan dengan cara memvalidasi jawaban yang dihasilkan oleh chatbot. Pengujian dilakukan dengan pendekatan validasi berbasis konten oleh ahli. Validasi dilakukan oleh tiga orang dosen validator yang kompeten. Pemilihan validator dilakukan secara purposif dengan mempertimbangkan kualifikasi akademik, penguasaan terhadap akademik unand, serta pengalaman dalam akademik kontemporer. Para validator merupakan dosen yang berasal dari Universitas Andalas yang dapat dilihat pada Tabel 2.

Tabel 2 Responden dan validator Pengujian

No	Nama Dosen	Fakultas
1	Lutfil Khairi	FTI
2	Nurfiah	FTI
3	Arifan Rahman	FTI

Tiga orang responden sekaligus validator yang ada pada Tabel 2 akan mengajukan 10 pertanyaan terkait akademik unand dan melakukan validasi jawaban dari chatbot. Para responden akan memberikan nilai atas jawaban yang dihasilkan oleh chatbot web dan Telegram menurut pengetahuan dan pemahaman para responden. Pada pengujian terdapat dua kategori yang dapat dipilih oleh responden, yaitu Sesuai (S) atau Tidak Sesuai (TS). Daftar pertanyaan dan hasil validasi responden dapat dilihat pada Tabel 3.

Tabel 3. Hasil Pengujian UAT

No	Pertanyaan	R1	R2	R3
1	Apakah Anda dapat berhasil mendaftar mata kuliah sesuai dengan KRS (Kartu Rencana Studi) yang telah disetujui?	S	S	S
2	Apakah dosen dan pegawai dapat mengajukan cuti akademik?	S	S	S
3	Apakah mahasiswa dapat mengetahui jumlah SKS sesuai IP (indeks prestasi)?	S	S	S
4	Apakah mahasiswa dapat mengajukan cuti akademik dan apa persyaratannya?	S	S	S
5	Bagaimana proses penilaian perkuliahan di unand?	S	S	S
6	Bagaimana tahapan dan syarat kenaikan pangkat dosen?	S	S	S
7	Apa saja persyaratan menjadi rektor unand?	S	S	S
8	Apakah Dosen boleh melakukan aktivitas jual beli di dalam lingkungan kampus?	TS	TS	TS
9	Berapa bobot sks penelitian dosen?	S	TS	TS
10	Apa Kriteria Mahasiswa Kena Dropout?	S	S	S

Keterangan: S = sesuai, TS = tidak sesuai, R1 = Responden 1, R2=Responden 2, R3= Responden 3

Pengujian selanjutnya yang akan dilakukan adalah *black box testing*. Pada pengujian ini akan dibuat beberapa skenario untuk menentukan apakah sistem yang dibuat berjalan sesuai fungsinya, yang dapat dilihat pada Tabel 4.

Tabel 4 Pengujian *Black box Testing*

No	Skenario Uji	Langkah Uji / Input	Output yang Diharapkan	Status
1	Memulai chatbot	Kirim /start	Pesan sambutan ditampilkan	Berhasil
2	Pertanyaan akademik valid	Ajukan: 'Apa kriteria DO?'	Jawaban sesuai dokumen + referensi	Berhasil
3	Menampilk an referensi	Klik tombol/ tautan referensi	Sumber dokumen terbuka/terlihat	Berhasil
4	Input tidak valid	Kirim emoji/karakter acak	Pesan validasi input (permintaan perjas)	Berhasil
5	Pertanyaan di luar domain	Ajukan: 'Harga iPhone 17?'	Chatbot menyatakan di luar domain & memberi arahan	Berhasil
6	Pertanyaan ambigu	Input: 'Bagaimana prosedurnya ?'	Chatbot meminta klarifikasi	Berhasil
7	Gangguan API	Putus koneksi API lalu ajukan pertanyaan	Pesan error yang ramah & instruksi coba lagi Chatbot	Berhasil
8	Pertanyaan sangat panjang	Input > 500 karakter	memproses/meringkas atau meminta penjelasan	Berhasil
9	Konsistensi multi-platfo rm	Bandingkan hasil Web vs Telegram (pertanyaan sama)	Konten jawaban konsisten (perbedaan hanya UI)	Berhasil
10	Waktu respons	Ukur durasi kirim→terima pada jam normal	< 3 detik (target)	Berhasil

Dari 10 pertanyaan yang diuji, 8 pertanyaan (80%) berstatus Lulus berdasarkan mayoritas dari tiga validator, sedangkan 2 pertanyaan (20%) berstatus Tidak Lulus. Akumulasi 30 penilaian menghasilkan distribusi S = 25 (83,33%) dan TS = 5 (16,67%). Tingkat kesepakatan antar-validator mencapai 90% (9 butir dinilai secara unanimitas) dengan Fleiss' kappa $\approx 0,76$, yang menunjukkan substantial agreement. Hasil ini mengindikasikan bahwa chatbot telah mampu memberikan jawaban yang akurat dan konsisten pada mayoritas skenario. Sebagai catatan, untuk butir No. 8 (aktivitas jual beli di kampus) dan No. 9 (bobot SKS penelitian dosen) memang tidak ada dokumen resmi yang merujuk pada hal tersebut.

Black-box testing digunakan untuk memverifikasi fungsi-fungsi eksternal *chatbot* tanpa melihat struktur internal. Pengujian dilakukan berbasis skenario fungsional yang mencakup beragam jenis input (valid, tidak valid, ambigu), interaksi API (FastAPI), serta respons pada dua platform (Web dan Telegram). Setiap

skenario memuat: kondisi awal, langkah uji/input, output yang diharapkan, hasil aktual, dan status (Berhasil/Gagal). Tujuannya adalah memastikan chatbot: (i) menerima dan memvalidasi input dengan benar, (ii) melakukan pencarian konteks di *knowledge base* melalui API, (iii) menyajikan jawaban beserta referensi, (iv) menangani input tidak valid dan pertanyaan di luar domain, serta (v) menampilkan pesan kesalahan yang ramah saat terjadi gangguan layanan.

Seluruh 10 skenario fungsional memperoleh status Berhasil (100%). Tidak ditemukan error fatal; respons sesuai rancangan termasuk penampilan referensi, validasi input, penanganan luar domain, dan pesan kesalahan saat gangguan API. Hal ini mengonfirmasi stabilitas integrasi API dan konsistensi keluaran pada platform web dan Telegram.

4. Kesimpulan

Penelitian ini berhasil menerapkan pembuatan sistem tanya-jawab berbasis chatbot Telegram, di mana data yang digunakan yaitu publikasi akademik kontemporer karya penulis terkemuka. Penelitian ini menggunakan Large Language Model (LLM) yang terintegrasi dengan struktur LangChain sebagai kerangka sistem tanya-jawab. Penelitian ini juga menerapkan arsitektur microservice yang mana struktur model API dan pengembangan chatbot Telegram-nya dibuat terpisah. Pembuatan API menggunakan FastAPI memungkinkan komunikasi antara knowledge base dan chatbot menjadi efisien. Penelitian ini juga memberikan transparansi informasi dari jawaban yang dihasilkan oleh chatbot dengan menambahkan fitur InlineKeyboard pada chatbot berbasis web dan Telegram yang memungkinkan pengguna untuk melihat referensi dari jawaban yang dihasilkan oleh chatbot.

Hasil evaluasi menunjukkan performa yang sangat baik. Pada User Acceptance Test (UAT) yang melibatkan tiga validator ahli dari FTI Universitas Andalas, dilakukan penilaian terhadap 10 pertanyaan terkait kebijakan akademik UNAND. Berdasarkan penilaian per-validator, terkumpul 30 penilaian dengan distribusi Sesuai (S) = 25 penilaian (83,33%) dan Tidak Sesuai (TS) = 5 penilaian (16,67%). Berdasarkan status mayoritas, 8 dari 10 pertanyaan (80%) dinyatakan Lulus, menunjukkan bahwa chatbot mampu menghasilkan jawaban yang akurat, relevan, serta konsisten dengan dokumen peraturan akademik yang berlaku. Tingkat kesepakatan antar-validator juga tinggi, dengan percent agreement mencapai 90% dan Fleiss' kappa $\approx 0,76$, menunjukkan substantial agreement serta reliabilitas penilaian yang kuat. Evaluasi UAT terhadap 10 pertanyaan oleh tiga validator menghasilkan S = 25 (83,33%) dan TS = 5 (16,67%), dengan 8/10 pertanyaan (80%) berstatus Lulus (mayoritas sesuai), percent agreement 90%, dan Fleiss' kappa $\approx 0,76$ (substantial agreement). Black-box testing pada 10 skenario fungsional menunjukkan 100% berhasil, menegaskan

stabilitas integrasi API dan konsistensi keluaran lintas platform. Sistem dinilai layak dan efektif sebagai kanal informasi akademik, dengan ruang perbaikan pada topik aktivitas non-akademik di kampus dan bobot SKS penelitian melalui penyempurnaan dataset dokumen.

Ucapan Terimakasih

Terimakasih Terimakasih kepada LPPM Universitas Andalas yang sudah mendanai penelitian berdasarkan Surat Keputusan Nomor 32/UN16.19/KPT/PT.01.00/2025 dan Perjanjian /Kontrak Nomor 465/UN16.19/PT.01.03/PLB/2025, sehingga artikel ini dapat diterbitkan.

Daftar Rujukan

- [1] Sindy Nova, Nurul Khotimah, and Maria Y Aryati Wahyuningrum, "Pemanfaatan Chatbot Menggunakan Natural Language Processing Untuk Pembelajaran Dasar-Dasar GUI Tkinter Pada Bahasa Pemrograman Python," *J. Ilm. Tek.*, Vol. 3, No. 1 Se-Articles, Pp. 58–65, Jan. 2024, Doi: [10.56127/Juit.V3i1.1162](https://doi.org/10.56127/Juit.V3i1.1162).
- [2] L. Aldri Permana, Nazruddin Safaat H, Lestari Handayani, And Muhammad Affandes, "Implementation Of Question Answering Feature In Mobile-Based Qur'an Application Using Flutter Framework," *Jupiter J. Penelit. Ilmu Dan Teknol. Komput.*, Vol. 16, No. 1 Se-Articles, Pp. 301–312, Mar. 2024, Doi: [10.5281/Zenodo.10868272](https://doi.org/10.5281/Zenodo.10868272).
- [3] N. S. Harahap, A. Saad, And N. H. Ubaidullah, "Comprehensive Bibliometric Literature Review Of Chatbot Research: Trends, Frameworks, And Emerging Applications," *Int. J. Adv. Comput. Sci. Appl.*, Vol. 16, No. 1, 2025, Doi: <https://doi.org/10.14569/Ijcsa.2025.0160185>.
- [4] I. Nur, M. N. Muttaqin, And A. A. Wakhid, "Reformulasi Al-Qawā'id Al-Uṣūliyyah Al-Tashrī'iyyah Dalam Pengembangan Fiqih Kontemporer," *Anal. J. Stud. Keislam.*, Vol. 20, No. 1, Pp. 47–68, 2020, Doi: <https://doi.org/10.24042/Ajsk.V20i1.6573>.
- [5] Meilani, Misbahuddin, And Shuhufi, "Fikih Kontemporer: Analisis Faktor-Faktor Yang Mendasari Dan Strategi Penyelesaiannya," *J. Kolaboratif Sains*, Vol. 7, No. 6, Pp. 1925–1932, 2024, Doi: <https://doi.org/10.56338/Jks.V7i6.5381>.
- [6] Y. Alkhurayyif And A. R. W. Sait, "Developing An Open Domain Arabic Question Answering System Using A Deep Learning Technique," *Ieee Access*, Vol. 11, Pp. 69131–69143, 2023, Doi: [10.1109/Access.2023.3292190](https://doi.org/10.1109/Access.2023.3292190).
- [7] S. M, "Leveraging Langchain Agents To Automate Data Analysis For Saas," *Artif. Intell.*, 2024, Doi: <https://doi.org/10.15407/Jai2024.02.107>.
- [8] A. T. Kumar, C. Wang, A. Dong, And J. Rose, "Generation Of Backward-Looking Complex Reflections For A Motivational Interviewing-Based Smoking Cessation Chatbot Using Gpt-4: Algorithm Development And Validation," *Jmir Ment Heal.*, Vol. 11, P. E53778, 2024, Doi: [10.2196/53778](https://doi.org/10.2196/53778).
- [9] A. Koubaa, "Gpt-4 Vs. Gpt-3.5: A Concise Showdown," *Preprints*, 2023, Doi: [10.20944/Preprints202303.0422.V1](https://doi.org/10.20944/Preprints202303.0422.V1).
- [10] F. Fransisco, F. Akbar, S. Kurniasari, M. I. Hibban, T. Desyani, And E. Nirmala, "Penerapan Chatbot Auto Reply Pada Telegram Terhadap Pengembangan Bisnis Penjualan Sepatu Menggunakan Artificial Intelligence Markup Language," *J. Teknol. Sist. Inf. Dan Apl.*, Vol. 5, No. 2 Se-Article, Pp. 85–90, Apr. 2022, Doi: [10.32493/Jtsi.V5i2.14733](https://doi.org/10.32493/Jtsi.V5i2.14733).
- [11] F. R. A. Sutiyo, N. S. Harahap, S. Agustian, And R. M. Candra, "Implementasi Question Answering Berbasis Chatbot Telegram Pada Tafsir Al-Jalalain Menggunakan Langchain Dan Llm," *Klik Kaji. Ilm. Inform. Dan Komput.*, Vol. 4, No. 5, Pp. 2464–2472, 2025, Doi: <https://doi.org/10.30865/Klik.V4i5.1784>.
- [12] M. Furqan, S. Sriani, And M. N. Shidqi, "Chatbot Telegram

- Menggunakan Natural Language Processing,” *Walisono J. Inf. Technol.*, Vol. 5, No. 1, Pp. 15–26, Jun. 2023, Doi: [10.21580/Wjit.2023.5.1.14793](https://doi.org/10.21580/Wjit.2023.5.1.14793).
- [13] A. Asroni, G. Indrawan, And L. J. Erawati Dewi, “Implementasi Hirarki Dataset Dalam Membangun Model Language Aksara Bali Menggunakan Framework Tesseract Ocr,” *J. Resist. (Rekayasa Sist. Komputer)*, Vol. 6, No. 1 Se-Articles, Pp. 20–28, Apr. 2023, Doi: [10.31598/Jurnalresistor.V6i1.1345](https://doi.org/10.31598/Jurnalresistor.V6i1.1345).
- [14] W. Suharmawan, “Pemanfaatan Chat Gpt Dalam Dunia Pendidikan,” *Educ. J. J. Educ. Res. Dev.*, Vol. 7, No. 2 Se-Articles, Pp. 158–166, Jan. 2024, Doi: [10.31537/Ej.V7i2.1248](https://doi.org/10.31537/Ej.V7i2.1248).
- [15] J. Benzinho Et Al., “Llm Based Chatbot For Farm-To-Fork Blockchain Traceability Platform,” *Applied Sciences*, Vol. 14, No. 19, P. 8856, 2024, Doi: [10.3390/App14198856](https://doi.org/10.3390/App14198856).
- [16] O. Topsakal And T. C. Akinci, “Creating Large Language Model Applications Utilizing Langchain: A Primer On Developing Llm Apps Fast,” *Int. Conf. Appl. Eng. Nat. Sci.*, Vol. 1, No. 1 Se-Conference Papers, Pp. 1050–1056, Jul. 2023, Doi: [10.59287/Icaens.1127](https://doi.org/10.59287/Icaens.1127).
- [17] G. K. S. Ram And V. Muthumanikandan, “Retracted: Visistant: A Conversational Chatbot For Natural Language To Visualizations With Gemini Large Language Models,” *Ieee Access*, Vol. 12, Pp. 138547–138563, 2024, Doi: [10.1109/Access.2024.3465541](https://doi.org/10.1109/Access.2024.3465541).
- [18] F. Ishlakhuddin, A. Basir, And N. Nurlaela, “Rancang Bangun Sistem Tanya-Jawab Berbasis Aturan Stmik Muhammadiyah Paguyangan Brebes Dengan Menggunakan Telegram Chatbot,” *J. Inform. J. Pengemb. It*, Vol. 5, Pp. 100–105, Sep. 2020, Doi: [10.30591/Jpit.V5i3.2900](https://doi.org/10.30591/Jpit.V5i3.2900).
- [19] A. Nugraheni And M. Maryam, “Penerapan Teknologi Quick Response Code Dan Application Programming Interface Pada Perancangan Aplikasi Perpustakaan (Studi Kasus : Smp Negeri 25 Surakarta),” *Jipi (Jurnal Ilm. Penelit. Dan Pembelajaran Inform.)*, Vol. 7, No. 3, Pp. 100–105, 2022, Doi: <https://doi.org/10.30591/Jpit.V5i3.2900>.
- [20] M. Schwarz, K. Chapman, And B. Häussler, “Multilingual Medical Entity Recognition And Cross-Lingual Zero-Shot Linking With Facebook Ai Similarity Search,” In *Ceur Workshop Proceedings (Ceur-Ws.Org)*, 2022. [Online]. Available: <https://Ceur-Ws.Org/Vol-3202/Livingner-Paper12.Pdf>
- [21] J. S. P. Kase And P. O. N. Saian, “Pengembangan Aplikasi Reporting Penentuan Kuantitas Maksimum Pada Sistem Aplikasi Operasi Perusahaan Xyz Menggunakan Flask Framework,” *Jipi (Jurnal Ilm. Penelit. Dan Pembelajaran Inform.)*, Vol. 8, No. 4, Pp. 1288–1299, 2023, Doi: <https://doi.org/10.29100/Jipi.V8i4.4231>.
- [22] C. Wijayanto And Y. A. Susetyo, “Implementasi Flask Framework Pada Pembangunan Aplikasi Sistem Informasi Helpdesk (Sih),” *Jipi (Jurnal Ilm. Penelit. Dan Pembelajaran Inform.)*, Vol. 7, No. 3, 2022, [Online]. Available: <https://Jurnal.Stkipggritlungagung.Ac.Id/Index.Php/Jipi/Article/View/3161/1328>
- [23] A. F. R. Arianto And Suryanti, “Pengembangan Media Chatbot Telegram Menggunakan Metode Natural Language Processing Bagi Sekolah Dasar Kelas Iv Materi Selalu Berhemat Energi,” *J. Penelit. Pendidik. Guru Sekol. Dasar*, Vol. 10, No. 3, Pp. 589–597, 2022, [Online]. Available: <https://Ejournal.Unesa.Ac.Id/Index.Php/Jurnal-Penelitian-Pgsd/Article/View/46121>
- [24] F. Nurmahya And D. Diana, “Implementasi Penggunaan Chatbot Dalam Memberikan Informasi Pendukung Program Mbkm Dengan Menggunakan Metode Prototype,” *J. Teknol. Sist. Inf. Dan Apl.*, Vol. 6, No. 4, Pp. 750–758, 2023, Doi: <https://doi.org/10.32493/Jtsi.V6i4.34289>.
- [25] D. L. N. N. T. M. Zaini, “Rancang Bangun Marketplace Pada Pasar Tradisional Berbasis Android Bandar Lampung,” *Teknika*, No. Vol. 17 No. 2 (2023): Teknika Juli-Desember 2023, Pp. 347–357, 2023, [Online]. Available: <https://Jurnal.Polsri.Ac.Id/Index.Php/Teknika/Article/View/7009/2799>