

Pengembangan Chatbot “ASTIKA” Berbasis Rasa Framework untuk Layanan Akademik Prodi Teknik Informatika Universitas Semarang

Risky Mul Tazam^{*1}, Khoirudin²

^{1,2}Program Studi Teknik Informatika, Fakultas Teknologi Informasi dan Komunikasi,
Universitas Semarang

Email: ^{*1}riskymultazam67@gmail.com, ²khoirudin@usm.ac.id

(Naskah masuk: 1 Juni 2026, diterima untuk diterbitkan: 2 Juli 2026)

Abstrak: Layanan informasi akademik di perguruan tinggi seringkali terkendala oleh keterbatasan jam operasional dan beban pertanyaan berulang yang membebani tenaga administrasi. Penelitian ini mengembangkan chatbot bernama ASTIKA (Asisten Teknik Informatika) berbasis Rasa Open Source yang terintegrasi dengan sistem informasi penunjang berbasis Laravel untuk melayani kebutuhan informasi akademik mahasiswa Program Studi Teknik Informatika Universitas Semarang. Sistem dibangun dengan dua komponen utama, yaitu Rasa Framework sebagai mesin pemrosesan bahasa alami (NLU) dan manajemen dialog, serta Laravel sebagai backend pengelolaan FAQ melalui antarmuka web. Data pelatihan NLU terdiri dari 65 intent yang dikelompokkan ke dalam 6 kategori dengan total 863 contoh kalimat. Evaluasi model NLU menggunakan cross-validation 5-fold menghasilkan akurasi keseluruhan sebesar 95,6% dengan F1-Score rata-rata 0,956. Pengujian Black Box Testing menunjukkan seluruh fitur utama berfungsi sesuai spesifikasi. Kontribusi utama penelitian ini terletak pada kemampuan sistem mengabstraksi kompleksitas konfigurasi Rasa melalui antarmuka web sehingga admin dapat mengelola data pelatihan chatbot tanpa memerlukan keahlian teknis.

Kata Kunci – Chatbot; Rasa Framework; layanan akademik; pemrosesan bahasa alami; Laravel

Development of "ASTIKA" Chatbot Based on Rasa Framework for Academic Services of Informatics Engineering Department

Abstract: Academic information services in higher education institutions are often constrained by limited operational hours and repetitive inquiries that burden administrative staff. This study develops a chatbot named ASTIKA (Asisten Teknik Informatika) based on Rasa Open Source 3.12, integrated with a supporting information system built on Laravel 12, to serve the academic information needs of students in the Informatics Engineering Department at Universitas Semarang. The system is built using two main components, namely Rasa Framework as the natural language understanding (NLU) and dialogue management engine, as well as Laravel as the backend for FAQ management through a web-based interface. The NLU training data consists of 65 intents grouped into 6 categories with a total of 863 sample sentences. NLU model evaluation using 5-fold cross-validation achieved an overall accuracy of 95.6% with a weighted average F1-Score of 0.956. Black Box Testing confirmed that all core features function according to specifications. The primary contribution of this research lies in the system's ability to abstract the complexity of Rasa configuration through a web interface, enabling administrators to manage chatbot training data without requiring technical expertise.

Keywords – Chatbot; Rasa Framework; academic services; natural language processing; Laravel

1. PENDAHULUAN

Kemajuan teknologi informasi mendorong institusi pendidikan tinggi untuk mengadopsi solusi berbasis kecerdasan buatan guna meningkatkan kualitas layanan akademik. Berdasarkan hasil observasi dan wawancara di Program Studi Teknik Informatika Universitas Semarang, mahasiswa kerap membutuhkan informasi akademik yang dapat diakses kapan saja tanpa terbatas jam operasional. Pertanyaan seputar prosedur tugas akhir, jadwal, dan persyaratan akademik sering diajukan berulang kali sehingga membebani tenaga administrasi dan menghambat efisiensi layanan.

Salah satu teknologi yang terbukti efektif untuk menjawab kebutuhan tersebut adalah chatbot berbasis pemrosesan bahasa alami. Chatbot merupakan perangkat lunak yang mensimulasikan percakapan manusia sehingga mampu memahami pertanyaan dan memberikan respons secara otomatis dan real-time[1]. Hidayat dkk. membuktikan bahwa chatbot layanan akademik berbasis Rasa Framework mampu menjawab pertanyaan mahasiswa dengan baik tanpa batasan waktu[2], sementara Saputro dan Al Amin berhasil menerapkan chatbot berbasis Rasa pada layanan informasi publik dengan tingkat penerimaan pengguna yang tinggi[3]. Penerapan chatbot berbasis NLP juga telah terbukti efektif pada layanan helpdesk di luar domain akademik[4].

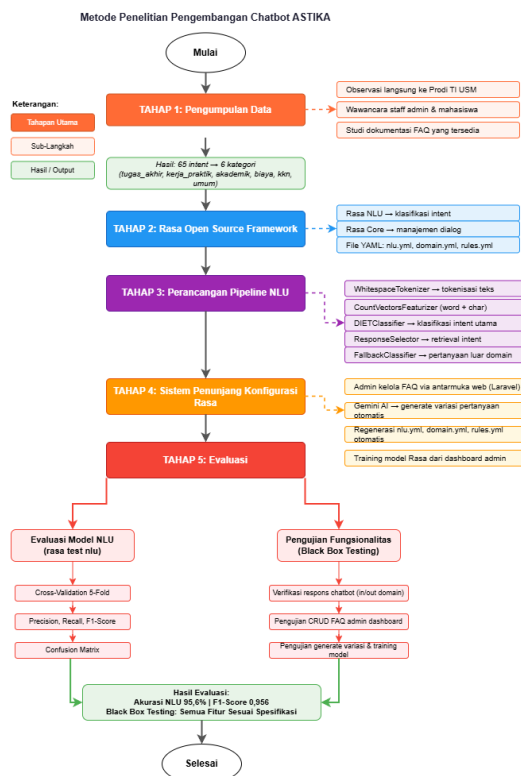
Dari berbagai framework yang tersedia, Rasa Open Source dipilih karena bersifat open source, dapat dikustomisasi penuh sesuai kebutuhan domain, serta memiliki dua komponen inti yakni Rasa NLU untuk klasifikasi intent dan Rasa Core untuk manajemen dialog[5]. Rasa menyediakan arsitektur pipeline modular yang memungkinkan pengembang mengkombinasikan berbagai komponen NLU sesuai karakteristik bahasa dan domain yang dihadapi. Komponen DIETClassifier yang digunakan dalam pipeline Rasa telah terbukti mampu menangani klasifikasi intent dengan baik pada domain akademik[6], sementara mekanisme ResponseSelector memungkinkan pengelompokan banyak sub-intent dalam satu kategori retrieval tanpa memerlukan rule dialog terpisah per pertanyaan.

Namun demikian, penelitian chatbot akademik berbasis Rasa umumnya mengharuskan admin atau pengembang mengedit file konfigurasi YAML secara manual untuk menambah atau mengubah data pelatihan. Hal ini menjadi hambatan dalam pengelolaan jangka panjang, terutama jika chatbot akan diserahkan kepada operator kampus yang tidak memiliki latar belakang teknis. Cannavaro[7] mengimplementasikan chatbot akademik berbasis Rasa dengan fitur Two Stage Fallback, namun pengelolaan data pelatihan masih dilakukan melalui pengeditan file konfigurasi langsung. Laksmi[8] merancang chatbot penerimaan mahasiswa baru menggunakan Rasa yang dilengkapi sistem informasi pemeliharaan berbasis web, namun implementasinya baru sampai tahap perancangan tanpa evaluasi performa NLU.

Berdasarkan gap tersebut, penelitian ini mengembangkan chatbot bernama ASTIKA (Asisten Teknik Informatika) yang memanfaatkan Rasa Open Source Framework sebagai mesin NLU dan dialog management, dengan fokus pada dua aspek utama: (1) perancangan pipeline NLU yang mengkombinasikan DIETClassifier dan ResponseSelector untuk menangani domain FAQ akademik dengan jumlah intent yang besar (65 intent, 6 kategori), serta (2) pengembangan sistem penunjang berbasis Laravel yang mengabstraksi kompleksitas konfigurasi Rasa melalui antarmuka web, terintegrasi dengan Gemini AI sebagai generator variasi pertanyaan otomatis, sehingga admin dapat mengelola data pelatihan tanpa keahlian teknis.

2. METODE PENELITIAN

Penelitian ini menggunakan pendekatan eksperimental dalam pengembangan chatbot ASTIKA dengan fokus pada perancangan pipeline NLU, penyusunan data pelatihan, serta evaluasi performa klasifikasi intent. Tahapan penelitian terdiri dari pengumpulan data, perancangan arsitektur Rasa, pengembangan sistem penunjang konfigurasi, dan evaluasi model, sebagaimana ditunjukkan pada Gambar 1.



Gambar 1. Tahapan Metode Penelitian

2.1. Pengumpulan Data

Pengumpulan data dilakukan melalui observasi langsung terhadap proses layanan informasi di Program Studi Teknik Informatika Universitas Semarang, wawancara dengan tenaga administrasi dan mahasiswa, serta studi dokumentasi terhadap FAQ yang telah tersedia. Dari proses tersebut, teridentifikasi bahwa pertanyaan mahasiswa umumnya berputar pada enam topik utama, yaitu: tugas akhir, kerja praktik, akademik, biaya, KKN, dan informasi umum. Keenam topik ini selanjutnya dijadikan sebagai kategori intent dalam sistem chatbot.

Setiap kategori kemudian dipecah menjadi intent-intent spesifik berdasarkan variasi topik pertanyaan yang ditemukan. Sebagai contoh, kategori tugas_akhir mencakup 24 intent yang masing-masing mewakili satu topik seperti syarat pengambilan tugas akhir, prosedur pengajuan judul, dan ketentuan format penulisan. Total data yang berhasil dikumpulkan terdiri dari 65 intent yang dikelompokkan ke dalam 6 kategori. Distribusi data per kategori disajikan pada Tabel 1.

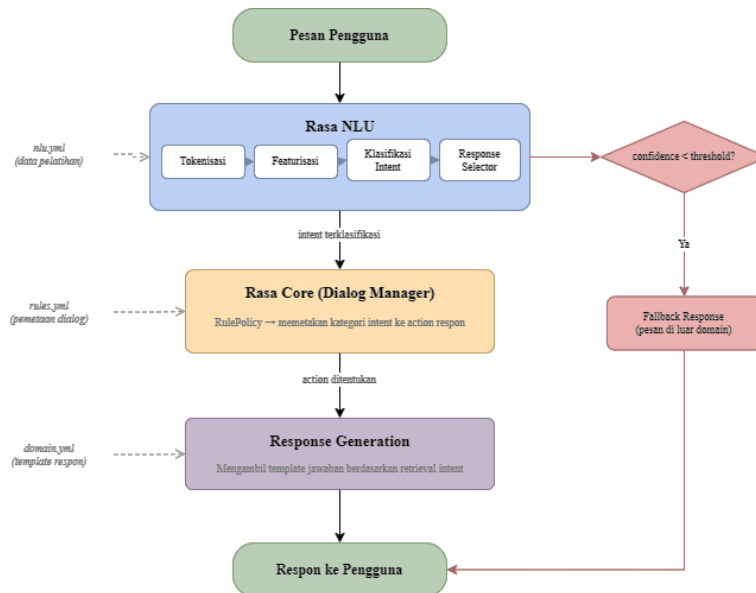
Tabel 1. Distribusi Data Pelatihan NLU Per Kategori

Kategori Intent	Jumlah Intent	Jumlah Contoh Kalimat	Contoh Topik
tugas_akhir	24	394	Syarat TA, prosedur pengajuan, batas waktu
kerja_praktik	25	297	Syarat KP, prosedur seminar, format laporan
akademik	6	60	Pengisian KRS, RKPM, persyaratan SKPI
biaya	7	84	Biaya kuliah, biaya TA, biaya KP, UKT akhir
kkn	1	8	Kewajiban KKN
umum	2	20	Salam, Penutup
Total	65	863	

2.2. Rasa Open Source Framework

Rasa Open Source merupakan framework untuk pengembangan chatbot berbasis machine learning yang terdiri dari dua komponen utama: Rasa NLU untuk klasifikasi intent dari pesan pengguna, dan Rasa Core untuk manajemen dialog yang menentukan action respon berdasarkan hasil klasifikasi. Kedua komponen ini dikonfigurasi melalui beberapa file YAML. File yang

berkaitan langsung dengan data pelatihan dan respon chatbot meliputi nlu.yml yang berisi variasi kalimat per intent, domain.yml yang mendefinisikan daftar intent dan template respon, serta rules.yml yang memetakan intent ke action. File-file ini harus konsisten satu sama lain, ketidakkonsistenan antar file akan menyebabkan error saat proses training. Gambar 2 menunjukkan alur pemrosesan pesan pada Rasa Framework.



Gambar 2. Alur pemrosesan pesan pada Rasa Framework

2.3. Perancangan Pipeline NLU

Pipeline NLU merupakan rangkaian komponen yang memproses pesan pengguna secara berurutan mulai dari tokenisasi teks hingga klasifikasi intent. Pada domain FAQ, setiap pertanyaan bersifat independen sehingga tidak memerlukan manajemen dialog yang kompleks cukup menggunakan pendekatan rule-based melalui RulePolicy[9]. Dengan demikian, perancangan difokuskan pada pipeline NLU yang menentukan seberapa akurat chatbot memahami maksud pertanyaan pengguna. Tabel 2 menyajikan konfigurasi pipeline yang digunakan, dengan mempertimbangkan karakteristik Bahasa Indonesia dan jumlah intent yang besar (65 intent dalam 6 kategori).

Tabel 2. Konfigurasi Pipeline NLU Chatbot ASTIKA

Komponen	Peran
WhitespaceTokenizer	Memecah kalimat menjadi token
RegexFeaturizer	Mengekstrak fitur berdasarkan pola regex untuk mengenali pola kosakata spesifik
LexicalSyntacticFeaturizer	Mengekstrak fitur leksikal dan sintaktik dari konteks kata sekitar
CountVectorsFeaturizer (word)	Menghitung frekuensi kemunculan kata dan bigram sebagai representasi fitur bag-of-words
CountVectorsFeaturizer (char)	Menghitung frekuensi n-gram karakter untuk toleransi terhadap kesalahan ketik
DIETClassifier	Mengklasifikasikan intent pengguna
ResponseSelector	Memilih jawaban paling relevan dari kumpulan respon dalam satu kategori retrieval intent
FallbackClassifier	Mengklasifikasikan pertanyaan sebagai out-of-scope jika confidence di bawah threshold.

Pada tahap featurisasi, digunakan dua CountVectorsFeaturizer: word n-gram (1-2) untuk menangkap pola kata, dan character n-gram (1-4) untuk toleransi terhadap kesalahan ketik seperti

"tugas akhir" atau "kerja pretek". DIETClassifier dipilih sebagai classifier utama karena kemampuannya menangani dataset dengan jumlah intent yang besar[10].

ResponseSelector digunakan untuk menangani mekanisme retrieval intent, yaitu pengelompokan banyak sub-intent dalam satu kategori menggunakan format hierarkis kategori/nama_intent, misalnya tugas_akhir/tanya_syarat_ta. Dengan pendekatan ini, hanya diperlukan 6 rule untuk menangani seluruh 65 intent, dibandingkan 65 rule jika menggunakan pendekatan intent konvensional. FallbackClassifier digunakan untuk menangani pertanyaan di luar domain.

2.4. Sistem Penunjang Konfigurasi Rasa

Salah satu tantangan dalam penerapan Rasa Framework adalah keharusan mengedit file konfigurasi YAML secara manual setiap kali terdapat penambahan atau perubahan data pelatihan[11]. Bagi admin program studi yang tidak memiliki latar belakang IT, proses ini menjadi hambatan dalam pengelolaan chatbot secara berkelanjutan. Untuk mengatasi hal tersebut, dikembangkan sistem penunjang berbasis Laravel yang mengabstraksi kompleksitas konfigurasi Rasa melalui antarmuka web.

Melalui sistem ini, admin cukup mengelola data FAQ melalui formulir web tanpa perlu memahami format YAML Rasa. Ketika admin menambahkan FAQ baru, data tersimpan ke database dengan status draft. Karena performa model NLU Rasa sangat bergantung pada jumlah dan keragaman variasi kalimat dalam data pelatihan, sistem mengintegrasikan Gemini AI sebagai alat bantu untuk menghasilkan variasi kalimat secara otomatis dari satu seed question. Variasi yang dihasilkan mencakup pola kalimat formal, informal, dan singkatan lazim mahasiswa, sehingga admin tidak perlu memiliki pemahaman mendalam tentang penyusunan data pelatihan NLU. Hasil generate ditampilkan dalam bentuk checklist sehingga admin cukup menyeleksi variasi yang relevan atau menambahkan variasi secara manual sebelum menyimpannya.

Setelah data FAQ siap, admin menekan tombol Update Rasa Config untuk meregenerasi nlu.yml, domain.yml, dan rules.yml secara otomatis dari seluruh data di database. Selanjutnya admin dapat langsung memicu proses pelatihan model dari dashboard tanpa mengakses server secara langsung.

2.5. Evaluasi

Evaluasi dilakukan menggunakan dua metode untuk mengukur performa sistem dari dua aspek yang berbeda.

Pertama, evaluasi model NLU dilakukan menggunakan perintah rasa test nlu dengan teknik cross-validation 5-fold untuk mengukur kemampuan chatbot dalam mengklasifikasikan intent dari pesan pengguna. Teknik cross-validation dipilih untuk memastikan hasil evaluasi lebih representatif dengan memanfaatkan seluruh data pelatihan secara bergantian sebagai data uji[12]. Metrik evaluasi yang digunakan meliputi Precision, Recall, dan F1-Score beserta confusion matrix dari prediksi intent[13].

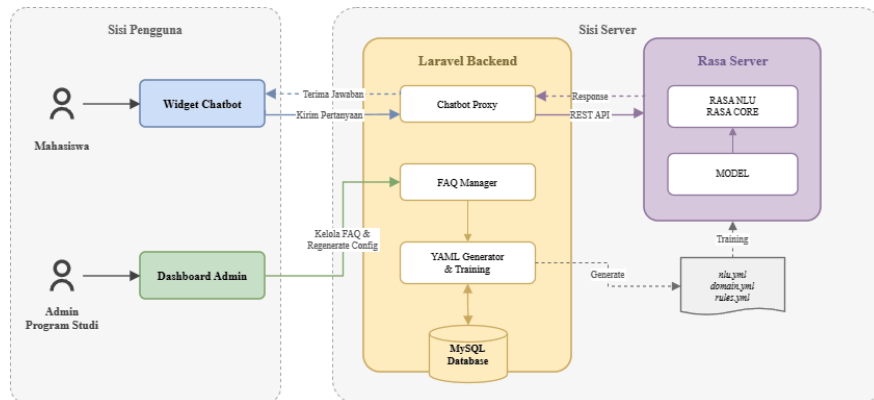
Kedua, pengujian fungsionalitas sistem dilakukan menggunakan metode Black Box Testing untuk memvalidasi kesesuaian keluaran sistem terhadap spesifikasi yang telah ditentukan. Pengujian ini mencakup verifikasi respon chatbot terhadap pertanyaan dalam dan luar domain, pengelolaan data FAQ pada dashboard admin, generate variasi pertanyaan menggunakan Gemini AI, regenerasi file konfigurasi Rasa dari database, serta pelatihan model Rasa dari dashboard admin.

3. HASIL DAN PEMBAHASAN

3.1. Arsitektur Sistem

Chatbot ASTIKA dibangun dengan arsitektur client-server yang memisahkan sisi pengguna dan sisi server. Pada sisi pengguna, terdapat widget chatbot yang terintegrasi pada halaman website program studi untuk mahasiswa dan dashboard admin untuk pengelolaan FAQ. Pada sisi server, Laravel backend menghubungkan kedua antarmuka tersebut dengan Rasa Server sebagai mesin

pemrosesan bahasa alami, MySQL sebagai penyimpanan data FAQ, serta file konfigurasi YAML yang di-generate otomatis untuk melatih model Rasa. Gambar 3 menunjukkan arsitektur sistem secara keseluruhan.



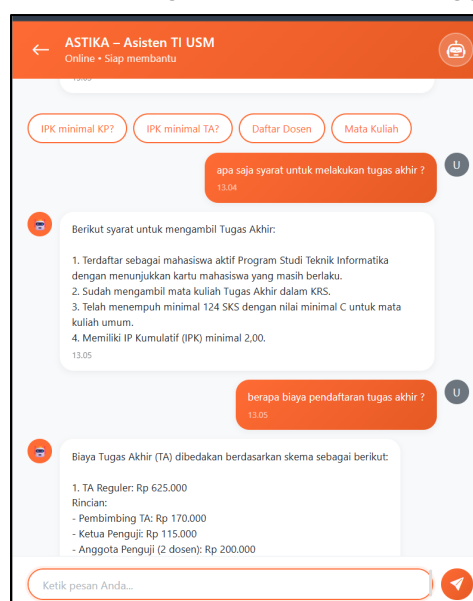
Gambar 3. Arsitektur sistem chatbot ASTIKA

3.2. Implementasi Antarmuka Sistem

Antarmuka sistem chatbot ASTIKA terdiri dari dua sisi utama, yaitu sisi frontend berupa widget chatbot yang terintegrasi pada website program studi, dan sisi backend berupa dashboard admin untuk mengelola data FAQ.

3.2.1. Widget Chatbot

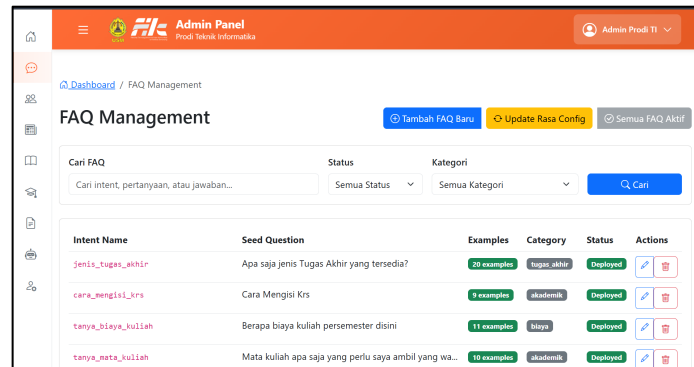
Widget chatbot ASTIKA ditempatkan pada halaman website Program Studi Teknik Informatika dengan tampilan antarmuka yang memudahkan pengguna dalam berinteraksi. Seperti yang ditunjukkan pada Gambar 4, yang memperlihatkan skenario percakapan nyata di mana pengguna mengajukan pertanyaan mengenai syarat tugas akhir, kemudian melanjutkan dengan pertanyaan terkait biaya pendaftaran tugas akhir. Meskipun pertanyaan kedua mengandung kata kunci dari dua kategori sekaligus (biaya dan tugas_akhir), chatbot berhasil mengklasifikasikan masing-masing intent ke kategori yang tepat dan memberikan respons yang terstruktur sesuai data yang telah dikonfigurasi melalui sistem admin. Hasil ini menunjukkan bahwa model NLU mampu menangani ambiguitas kosakata lintas kategori pada kondisi penggunaan langsung.



Gambar 4. Tampilan percakapan chatbot ASTIKA

3.2.2. Dashboard Admin Kelola FAQ

Sistem penunjang berbasis Laravel dikembangkan untuk memungkinkan admin program studi mengelola data FAQ tanpa memerlukan keahlian teknis dalam konfigurasi Rasa. Seperti yang terlihat pada Gambar 5, halaman FAQ Management menyajikan daftar seluruh FAQ yang terdaftar dalam sistem beserta informasi nama intent, seed question, jumlah examples, dan kategori. Halaman ini dilengkapi fitur pencarian berdasarkan kata kunci dan filter berdasarkan kategori, serta tombol aksi utama untuk menambah FAQ dan memperbarui konfigurasi Rasa.



Gambar 5. Tampilan halaman FAQ Management pada dashboard admin

3.2.3. Fitur Tambah FAQ

Admin dapat menambahkan FAQ baru melalui formulir yang dirancang dengan alur sederhana dan intuitif. Seperti yang ditunjukkan pada Gambar 6, formulir tersebut terdiri dari empat field utama, yaitu Intent Name sebagai identitas unik FAQ, Category untuk mengelompokkan FAQ ke dalam topik yang telah ditentukan, Seed Question berupa satu contoh pertanyaan utama yang mewakili intent, dan Answer berupa respons yang akan diberikan chatbot ketika intent terdeteksi.



Gambar 6. Tampilan formulir tambah FAQ baru

3.2.4. Fitur Edit FAQ dan Generate Variasi Pertanyaan

Halaman edit FAQ sekaligus menyediakan fitur AI Examples Generator yang memungkinkan admin menghasilkan variasi pertanyaan secara otomatis menggunakan Gemini AI. Seperti yang ditunjukkan pada Gambar 7, setelah admin menekan tombol "Generate Examples with AI", sistem mengirimkan seed question beserta konteks jawaban ke Gemini AI melalui API dan hasilnya ditampilkan dalam bentuk checklist di sisi kanan halaman. Variasi yang dihasilkan mencakup pola kalimat formal dan informal serta penggunaan singkatan lazim mahasiswa, sehingga admin cukup menyeleksi variasi yang relevan atau menambahkan variasi secara manual sebelum menyimpannya

Gambar 7. Tampilan halaman edit FAQ dengan hasil generate variasi pertanyaan menggunakan Gemini AI

3.3. Pengujian Fungsionalitas Sistem

Pengujian fungsionalitas sistem dilakukan menggunakan metode Black Box Testing dengan mengevaluasi kesesuaian keluaran sistem terhadap masukan yang diberikan pada setiap fitur utama. Hasil pengujian disajikan pada Tabel 3.

Tabel 3. Hasil Pengujian Black Box Testing

Skenario Pengujian	Masukan	Keluaran yang Diharapkan	Hasil
Chatbot merespons pertanyaan sesuai domain	Mengetik: "Apa syarat ambil TA?"	Jawaban sesuai intent ditampilkan	Sesuai
Chatbot menangani pertanyaan di luar domain	Mengetik: "Siapa presiden Indonesia?"	Pesan fallback ditampilkan	Sesuai
CRUD FAQ pada admin panel	Tambah, edit, dan hapus data FAQ	FAQ tersimpan, diperbarui, dan terhapus sesuai aksi	Sesuai
Generate variasi pertanyaan menggunakan Gemini AI	Klik "Generate Examples with AI" pada halaman edit FAQ	Daftar variasi muncul dalam bentuk checklist	Sesuai
Regenerasi file konfigurasi Rasa dari database	Klik "Update Rasa Config"	File nlu.yml, domain.yml, dan rules.yml ter-regenerate	Sesuai
Training model Rasa dari dashboard admin	Klik tombol training model	Job training berjalan dan model baru dihasilkan	Sesuai

Seluruh skenario pengujian menghasilkan keluaran yang sesuai dengan spesifikasi, menunjukkan bahwa integrasi antara widget chatbot, Rasa Server, dan sistem admin Laravel berjalan dengan baik.

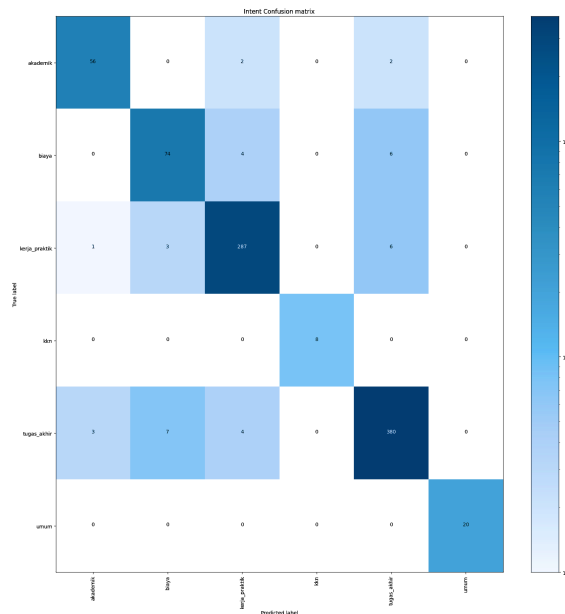
3.4. Evaluasi Model NLU

Evaluasi performa model NLU dilakukan menggunakan perintah rasa test nlu dengan teknik cross-validation 5-fold untuk mengukur kemampuan chatbot dalam mengklasifikasikan intent. Total data yang diuji adalah 863 sampel kalimat yang mencakup 6 kategori intent dengan metrik Precision, Recall, dan F1-Score.

Tabel 4. Hasil Evaluasi Performa Model NLU Per Kategori Intent

Kategori Intent	Support	Precision	Recall	F1-Score
tugas_akhir	394	0,964	0,964	0,964
kerja_praktik	297	0,966	0,966	0,966
biaya	84	0,881	0,881	0,881
akademik	60	0,933	0,933	0,933
umum	20	1,000	1,000	1,000
kkn	8	1,000	1,000	1,000
Weighted Average / Accuracy	863	0,956	0,956	0,956

Berdasarkan Tabel 4, model NLU memperoleh akurasi keseluruhan sebesar 95,6% dengan F1-Score rata-rata 0,956. Kategori kkn dan umum mencapai F1-Score sempurna karena pola kalimatnya sangat khas dan tidak beririsan dengan kategori lain. Kategori kerja_praktik dan tugas_akhir juga menunjukkan performa tinggi masing-masing 0,966 dan 0,964 meskipun keduanya memiliki jumlah intent dan sampel terbesar.



Gambar 8. Confusion matrix hasil klasifikasi intent chatbot ASTIKA

Analisis confusion matrix pada Gambar 8 mengkonfirmasi bahwa kesalahan klasifikasi utama terjadi akibat kemiripan kosakata antar kategori yang saling berkaitan, khususnya antara biaya, tugas_akhir, dan kerja_praktik. Sementara itu, analisis distribusi confidence score menunjukkan bahwa mayoritas prediksi benar memiliki confidence tinggi (0,98–1,00), namun sebagian prediksi salah juga memiliki confidence serupa, mengindikasikan perlunya penambahan variasi data pelatihan yang lebih distinktif pada intent-intent yang masih sering tertukar.

3.5. Pembahasan

Akurasi 95,6% menandakan bahwa dari 863 sampel, sebanyak 825 berhasil diklasifikasikan dengan tepat. Performa terendah pada kategori biaya (F1-Score = 0,881) disebabkan oleh sifat pertanyaan biaya yang selalu terikat konteks topik lain, misalnya “berapa biaya pendaftaran kerja praktik” yang mengandung kata kunci dari dua kategori sekaligus. Peningkatan performa dapat dilakukan melalui penambahan variasi contoh kalimat yang lebih distinktif serta penyesuaian konfigurasi pipeline NLU, seperti peningkatan jumlah epochs pada DIETClassifier atau penambahan RegexFeaturizer yang lebih spesifik untuk kata kunci biaya.

Perbandingan dengan penelitian sejenis disajikan pada Tabel 5. Chatbot LIANA dengan 12 intent dan 188 sampel memperoleh akurasi 0,995, sementara ASTIKA dengan domain jauh lebih besar (65 intent, 863 sampel) memperoleh 0,956. Sebagai pembandingan lintas metode, chatbot FAQ berbasis Naive Bayes Classifier untuk layanan umum [14] hanya memperoleh akurasi 0,692, mengindikasikan bahwa pemilihan arsitektur model, khususnya kombinasi DIET Classifier dengan ResponseSelector, berpengaruh signifikan terhadap performa klasifikasi intent, terlepas dari perbedaan domain. Perbedaan ini wajar mengingat semakin banyak intent yang harus dibedakan, semakin besar peluang tumpang tindih pola antar kategori. Meskipun demikian, akurasi 95,6% menunjukkan bahwa kombinasi CountVectorsFeaturizer dengan DIETClassifier dan ResponseSelector mampu menangani domain FAQ akademik yang lebih kompleks dengan baik.

Tabel 5. Perbandingan Hasil Evaluasi dengan Penelitian Sejenis

Aspek	LIANA [1]	FAQ [14]	Astika	Keterangan
Framework	Rasa	Naive Bayes	Rasa	
Jumlah Intent	12	6	65	ASTIKA domain terbesar
Jumlah Sampel	188	30	863	
Akurasi	0,995	0,692	0,956	Tetap tinggi meski domain lebih besar
F1-Score	0,995	0,605	0,956	
Admin FAQ	Tidak ada	Tidak ada	Ada (Laravel)	Keunggulan ASTIKA
AI Generator	Tidak ada	Tidak ada	Gemini AI	Keunggulan ASTIKA

Di luar performa NLU, kontribusi utama penelitian ini adalah sistem informasi penunjang yang memungkinkan admin mengelola data pelatihan chatbot tanpa keahlian teknis. Berbeda dengan penelitian sebelumnya [1], [14] yang mengharuskan pengeditan file konfigurasi Rasa secara manual, ASTIKA mengabstraksi proses tersebut melalui antarmuka web. Integrasi Gemini AI sebagai generator variasi pertanyaan juga mengurangi beban penyusunan data pelatihan, di mana admin cukup menyediakan satu seed question untuk menghasilkan 10–20 variasi kalimat secara otomatis.

Penelitian ini memiliki beberapa keterbatasan. Pertama, evaluasi model NLU dilakukan menggunakan data pelatihan dengan teknik cross-validation, sehingga hasil akurasi belum mencerminkan performa chatbot pada kondisi percakapan nyata. Kedua, chatbot belum mendukung percakapan multi-turn sehingga setiap pertanyaan diproses secara independen tanpa mempertimbangkan konteks percakapan sebelumnya.

4. KESIMPULAN

Penelitian ini berhasil mengembangkan chatbot ASTIKA berbasis Rasa Open Source untuk mendukung layanan informasi akademik mahasiswa Program Studi Teknik Informatika Universitas Semarang. Sistem dirancang dengan memanfaatkan pipeline NLU yang mengkombinasikan beberapa komponen featurization serta DIETClassifier sebagai model utama klasifikasi intent dan ResponseSelector untuk mekanisme retrieval intent pada domain FAQ akademik. Pendekatan ini memungkinkan chatbot menangani jumlah intent yang relatif besar, yaitu 65 intent yang dikelompokkan ke dalam 6 kategori.

Evaluasi model NLU menggunakan cross-validation 5-fold terhadap 863 sampel kalimat pada 65 intent menghasilkan akurasi keseluruhan sebesar 95,6% dengan F1-Score rata-rata 0,956. Kategori umum dan KKN mencapai skor sempurna, sementara kategori biaya memperoleh performa terendah (F1-Score 0,881) akibat tumpang tindih kosakata dengan kategori tugas akhir dan kerja praktik.

Sebagai solusi atas kompleksitas konfigurasi Rasa yang umumnya memerlukan pengeditan file YAML secara manual, penelitian ini juga mengembangkan sistem penunjang berbasis Laravel yang mengabstraksi proses pengeditan file YAML melalui antarmuka web, terintegrasi dengan Gemini AI sebagai generator variasi pertanyaan otomatis, sehingga admin dapat mengelola data pelatihan chatbot tanpa keahlian teknis. Adapun keterbatasan penelitian meliputi evaluasi yang belum menggunakan data di luar data pelatihan serta belum didukungnya percakapan multi-turn. Pengembangan selanjutnya dapat diarahkan pada pengujian dengan data real dari pengguna, penambahan fitur percakapan kontekstual, serta penguatan variasi data pelatihan pada kategori yang masih mengalami kesalahan klasifikasi.

UCAPAN TERIMA KASIH

Peneliti menyampaikan terima kasih kepada Universitas Semarang serta seluruh pihak yang telah berkontribusi dalam penelitian ini. Semoga kita semua senantiasa dirahmati oleh Allah SWT.

DAFTAR PUSTAKA

- [1] D. G. S. Ruindungan and A. Jacobus, "Chatbot Development for an Interactive Academic Information Services using the Rasa Open Source Framework," *Tek. Elektro dan Komput.*, vol.

- 10, no. 1, pp. 61–68, 2021.
- [2] I. N. Hidayat, D. Christanto, A. N. Rifai, and M. Bahrul Ulum, “Implementasi Rasa Framework Pada Chatbot Layanan Akademik (Studi Kasus: Fakultas Ilmu Komputer Universitas Esa Unggul),” *Komputa J. Ilm. Komput. dan Inform.*, vol. 13, no. 2, pp. 79–90, 2024, doi: 10.34010/komputa.v13i2.12178.
- [3] W. A. S. Adi and I. H. A. A. Imam, “Peningkatan Layanan Informasi BRT Trans Jateng berbasis Chatbot Telegram menggunakan Framework Rasa,” *J. Tek. Inform. UNIKA St. Thomas*, vol. 10, pp. 49–59, 2025.
- [4] M. Mukrodin and N. Mega Sasmita, “Artificial Intelligence Dalam Aplikasi Chatbot Sebagai Helpdesk Obyek Wisata Dengan Permodelan Natural Language Processing (Studi Kasus: Kabupaten Cilacap),” *Smart Comp Jurnalnya Orang Pint. Komput.*, vol. 10, no. 1, pp. 7–14, 2021, doi: 10.30591/smartcomp.v10i1.2135.
- [5] C. B. Vista, M. Tundjung, T. Fatmawati, E. S. Sintiya, J. T. Informasi, and P. N. Malang, “Pengembangan Chatbot Berbasis Framework RASA pada Website Bank Sampah Sriwilis,” *J. Inform. Polinema*, vol. 12, no. 2, pp. 271–278, 2026.
- [6] W. Astuti, A. P. Wibawa, H. Haviluddin, and H. Darwis, “DIET Classifier Model Analysis for Words Prediction in Academic Chatbot,” *Ilk. J. Ilm.*, vol. 16, no. 1, pp. 59–67, 2024, doi: 10.33096/ilkom.v16i1.1598.59-67.
- [7] N. Cannavaro, “Aplikasi Chatbot untuk Layanan Akademik Menggunakan Platform RASA Open Source dengan Fitur Two Stage Fallback,” *J. Ilmu Komput. dan Inform.*, vol. 3, no. 1, pp. 53–64, 2023, doi: 10.54082/jiki.73.
- [8] A. Laksmi, “Analisis Dan Perancangan Aplikasi Chatbot Menggunakan Framework Rasa Dan Sistem Informasi Pemeliharaan Aplikasi (Studi Kasus: Chatbot Penerimaan Mahasiswa Baru Politeknik Astra),” *J. Teknol. Inf. dan Ilmu Komput.*, vol. 10, no. 2, pp. 291–300, 2023, doi: 10.25126/jtiik.2022106409.
- [9] & S. Pratama, R. Y., “Pengembangan Sistem Chatbot Cerdas Berbasis Natural Language Processing (Nlp) Untuk Peningkatan Layanan Informasi Hotel,” *J. Inform. dan Tek. Elektro Terap.*, vol. 14, no. 1, 2026, doi: 10.23960/jitet.v14i1.8528.
- [10] G. Bercaru, C. O. Truică, C. G. Chiru, and T. Rebedea, “Improving Intent Classification Using Unlabeled Data from Large Corpora,” *Mathematics*, vol. 11, no. 3, pp. 1–20, 2023, doi: 10.3390/math11030769.
- [11] I. K. Resika Arthana, L. J. E. Dewi, K. A. Seputra, and N. W. Marti, “Undiksha Virtual Assistant (Shavira): Integration Frequency Asked Question With Rasa Framework,” *JST (Jurnal Sains dan Teknol.*, vol. 10, no. 2, pp. 264–273, 2021, doi: 10.23887/jstundiksha.v10i2.39863.
- [12] W. Wijiyanto, A. I. Pradana, S. Sopingi, and V. Atina, “Teknik K-Fold Cross Validation untuk Mengevaluasi Kinerja Mahasiswa,” *J. Algoritma*, vol. 21, no. 1, pp. 239–248, 2024, doi: 10.33364/algoritma/v.21-1.1618.
- [13] S. Sathyanarayanan, “Confusion Matrix-Based Performance Evaluation Metrics,” *African J. Biomed. Res.*, vol. 27, no. 4, pp. 4023–4031, 2024, doi: 10.53555/ajbr.v27i4s.4345.
- [14] F. A. Bima, A. Sanjaya, and R. A. Ramadhani, “Perancangan Chatbot FAQ Dengan Algoritma Naive Bayes Classifier Untuk Layanan Umum,” in *SEMNAS INOTEK (Seminar Nasional Inovasi Teknologi)*, 2025, pp. 800–809. [Online]. Available: <https://proceeding.unpkediri.ac.id/index.php/inotek/>