EKSPERIMEN MULTI-MODEL DAN MULTI-PERSONA UNTUK MENGANALISIS DAMPAK *PERSONA* TERHADAP PENALARAN, PERILAKU KELUARAN, DAN *HUMAN BIAS* PADA LARGE LANGUAGE MODEL

Proposal Tugas Akhir

Oleh

Abel Apriliani
18222008

PROGRAM STUDI SISTEM DAN TEKNOLOGI INFORMASI
SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA
INSTITUT TEKNOLOGI BANDUNG

Desember 2025

LEMBAR PENGESAHAN

EKSPERIMEN MULTI-MODEL DAN MULTI-PERSONA UNTUK MENGANALISIS DAMPAK *PERSONA* TERHADAP PENALARAN, PERILAKU KELUARAN, DAN *HUMAN BIAS* PADA LARGE LANGUAGE MODEL

Proposal Tugas Akhir

Oleh

Abel Apriliani
18222008

Program Studi Sistem dan Teknologi Informasi
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung

Proposal Tugas Akhir ini telah disetujui dan disahkan
di Bandung, pada tanggal 4 Desember 2025

Pembimbing 1	Pembimbing 2
Dr. Eng. Ayu Purwarianti, S.T., M.T.	Dr. Alham Fikri Aji, S.T., M.Sc.
NIP. x	NIP. x

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

DAFTAR KODE

BAB I

PENDAHULUAN

I.1 Latar Belakang

Kemajuan dalam pengembangan *large language model* (LLM) dalam beberapa tahun terakhir telah mengubah cara sistem komputasi memahami, memproses, dan menghasilkan bahasa alami. Model seperti GPT, LLaMA, Grok, dan Gemini dilatih menggunakan korpus berskala besar dan mampu menyelesaikan berbagai tugas mulai dari penalaran numerik hingga interpretasi skenario sosial (**jurafsky2023slp3**). Pada sejumlah benchmark terstandardisasi, model-model tersebut dapat memberikan jawaban yang akurat dan relevan. Namun, peningkatan kemampuan ini belum sepenuhnya diikuti oleh konsistensi perilaku model dalam percakapan. Perubahan kecil dalam cara pertanyaan disampaikan sering kali menghasilkan respons yang berbeda, meskipun tugas yang diberikan tetap sama (Zhou dkk. 2023).

Fenomena lain yang semakin banyak dibahas dalam penelitian mutakhir adalah bahwa perilaku model tidak hanya dipengaruhi oleh isi instruksi, tetapi juga oleh cara model memersepsi identitas pengguna. Studi mengenai bias penalaran implisit menunjukkan bahwa deskripsi singkat mengenai pengguna dapat mengubah pola penalaran model, termasuk pada tugas-tugas yang tidak memiliki muatan sosial, seperti penalaran numerik atau penyelesaian masalah dasar (Gupta dkk. 2024). Perubahan tersebut mencakup variasi langkah penyelesaian, tingkat kehati-hatian, ataupun kecenderungan preferensi tertentu terhadap kelompok sosial.

Selain persona yang dinyatakan secara langsung, beberapa penelitian menemukan bahwa model dapat mengasosiasikan isyarat linguistik halus—seperti pilihan kata, tingkat formalitas, atau gaya pertanyaan—dengan karakteristik tertentu dari pengguna (Tseng dkk. 2024). Asosiasi ini kemudian berpotensi memengaruhi strategi penyelesaian yang dipilih model, termasuk variasi pada langkah-langkah penalaran yang biasanya tercermin dalam *chain-of-thought*.

Penelitian dalam pemodelan pengguna juga menunjukkan bahwa identitas pengguna—meliputi usia, latar belakang profesional, maupun pengalaman tertentu—dapat memberikan pengaruh terhadap pola respons model (Naous, Roziere, dkk. 2025). Dalam penelitian ini, identitas pengguna direpresentasikan melalui persona yang dibentuk secara eksplisit maupun implisit di dalam prompt. Pendekatan tersebut digunakan untuk mengkaji bagaimana model membangun asumsi mengenai pengguna dan bagaimana asumsi tersebut tercermin pada keluaran model dalam berbagai skenario tugas.

Meskipun terdapat sejumlah temuan penting, penelitian terdahulu masih memiliki keterbatasan. Sebagian besar hanya melibatkan jumlah model yang terbatas, ruang persona yang sempit, atau cakupan tugas yang relatif kecil. Belum banyak penelitian yang secara sistematis membandingkan persona eksplisit dan implisit pada berbagai model dan berbagai jenis penalaran dalam kerangka eksperimen yang konsisten. Selain itu, penelitian mengenai perbedaan antara pendekatan persona berbasis pengguna ("your user is...") dan pendekatan berbasis model ("you are...") juga masih terbatas, padahal kedua bentuk framing tersebut berpotensi menghasilkan respons yang berbeda. Dalam konteks ini, studi seperti HELM (Liang, Bommasani, dkk. 2023) menegaskan bahwa model sensitif terhadap variasi konteks yang tampak kecil, sehingga evaluasi terstruktur menjadi semakin penting.

Keterbatasan tersebut semakin relevan mengingat penerapan model bahasa pada berbagai bidang yang sensitif terhadap identitas pengguna, seperti pendidikan, layanan kesehatan, dan sistem rekomendasi. Ketidakstabilan respons yang dipicu oleh variasi cara model memersepsi pengguna dapat mengurangi keandalan sistem dan menimbulkan bias. Selain itu, variasi hasil antar-*run* pada tugas yang sama menunjukkan perlunya mekanisme evaluasi yang terstruktur dan dapat direproduksi (Turpin dkk. 2023; Cobbe dkk. 2021).

Berangkat dari kebutuhan tersebut, penelitian ini disusun untuk mengevaluasi pengaruh persona eksplisit dan implisit melalui eksperimen terstruktur pada berbagai model dan jenis tugas penalaran. Penelitian ini memanfaatkan pendekatan *spec-driven experiment orchestration* yang memungkinkan pelaksanaan kombinasi persona–model–benchmark secara konsisten dan dapat diulang. Dengan pendekatan tersebut, penelitian ini diharapkan dapat memberikan gambaran yang lebih jelas mengenai bagaimana model menafsirkan identitas pengguna dan bagaimana penafsiran tersebut memengaruhi jawaban dalam berbagai konteks tugas.

I.2 Rumusan Masalah

Penelitian sebelumnya menunjukkan bahwa persona, baik yang diberikan secara eksplisit maupun yang tersirat dari gaya bahasa, dapat memengaruhi cara model menyusun penalaran dan menghasilkan jawaban (Gupta dkk. 2024; Tseng dkk. 2024; Naous, Roziere, dkk. 2025). Namun, kajian yang ada masih terbatas pada jumlah model yang sedikit, ragam persona yang sempit, serta jenis tugas yang belum cukup mencerminkan variasi penalaran yang lebih luas. Kondisi ini menunjukkan perlunya evaluasi yang lebih menyeluruh untuk memahami bagaimana persona memengaruhi perilaku model dalam konteks multi-tugas dan multi-model.

Berdasarkan uraian pada bagian sebelumnya, penelitian ini merumuskan beberapa pertanyaan utama sebagai berikut.

1. Sejauh mana persona yang diberikan secara eksplisit maupun yang muncul secara implisit memengaruhi proses penalaran model pada berbagai jenis tugas, khususnya penalaran numerik dan tugas multi-topik?
2. Bagaimana variasi persona tersebut membentuk karakter keluaran model dan memunculkan pola bias tertentu, termasuk bias sosial maupun preferensi jawaban?
3. Bagaimana perbedaan respons antar model dapat menggambarkan tingkat sensitivitas dan ketahanan masing-masing model terhadap variasi persona dalam suatu kerangka evaluasi yang disusun secara terstruktur?

I.3 Tujuan Penelitian

Tujuan penelitian ini disusun sebagai tindak lanjut dari rumusan masalah yang telah dijelaskan sebelumnya. Secara umum, penelitian ini bertujuan untuk memperoleh pemahaman yang lebih jelas mengenai pengaruh persona terhadap perilaku dan penalaran *large language model*. Secara khusus, penelitian ini bertujuan untuk:

1. Menganalisis sejauh mana persona yang diberikan secara eksplisit maupun yang muncul secara implisit memengaruhi proses *reasoning* model pada berbagai jenis tugas.
2. Mengidentifikasi perubahan karakter keluaran dan pola *bias* yang muncul pada model sebagai akibat dari variasi persona.
3. Mengevaluasi perbedaan respons antar model untuk menilai tingkat *sensitivity* dan *robustness* masing-masing model terhadap variasi persona dalam pengaturan eksperimen yang disusun secara terstruktur.

I.4 Batasan Masalah

Batasan masalah diperlukan agar ruang lingkup penelitian tetap jelas dan terarah. Penelitian ini tidak mencakup seluruh aspek perilaku *large language model*, tetapi memfokuskan kajian pada bagaimana variasi persona memengaruhi respons model pada sejumlah tugas penalaran. Adapun batasan penelitian ini adalah sebagai berikut.

1. Penelitian hanya mempertimbangkan dua bentuk persona yang berorientasi pada pengguna, yaitu persona eksplisit yang dinyatakan secara langsung di dalam prompt, serta persona implisit yang muncul dari variasi gaya bahasa dan cara pengguna menyampaikan pertanyaan. Kajian ini tidak mencakup *role-playing persona* yang menetapkan identitas tertentu pada model, maupun pendekatan *personalization* yang bergantung pada riwayat atau profil pengguna.
2. Model yang ditelaah terbatas pada model bahasa berbasis teks yang tersedia melalui antarmuka API. Model multimodal serta model yang memerlukan proses *fine-tuning* atau pelatihan ulang tidak menjadi bagian dari penelitian ini.
3. Evaluasi dilakukan pada tugas-tugas berbasis teks, meliputi penalaran numerik, penalaran logis, pertanyaan pengetahuan umum, serta skenario sosial dan moral. Penelitian ini tidak membahas tugas multimodal maupun tugas berbasis *speech*.
4. Penilaian terhadap respons model dilakukan melalui evaluasi otomatis dan analisis komparatif. Penelitian tidak melibatkan penilaian dengan partisipan manusia.
5. Seluruh eksperimen dijalankan melalui pendekatan *prompt-based evaluation* tanpa melakukan perubahan terhadap parameter internal model.
6. Analisis bias dibatasi pada *human bias* yang timbul sebagai konsekuensi variasi persona. Penelitian tidak mengevaluasi bias yang berasal dari data pelatihan model atau faktor struktur model lainnya.

I.5 Metodologi Penelitian

Penelitian ini menggunakan pendekatan eksperimental berbasis pemanggilan model melalui prompt untuk melihat bagaimana persona memengaruhi respons sejumlah *large language model*. Metodologi dirancang agar alur evaluasi jelas dan dapat dijalankan kembali apabila diperlukan. Tahapan penelitian disajikan sebagai berikut.

1. Perumusan spesifikasi eksperimen.

Tahap ini diawali dengan menyusun dokumen spesifikasi yang memetakan kombinasi persona, model, bentuk interaksi, dan jenis tugas yang akan diuji. Spesifikasi tersebut dipakai sebagai acuan sehingga pelaksanaan eksperimen berjalan dengan alur yang tetap.

2. Penyusunan persona eksplisit dan implisit.
   Persona eksplisit dituliskan secara langsung di dalam prompt, sedangkan persona implisit dibangun melalui variasi gaya bahasa pengguna tanpa menyebutkan identitas secara eksplisit. Kedua bentuk persona digunakan untuk melihat bagaimana model memahami karakter pengguna dari konteks yang berbeda.
3. Pemilihan model dan ruang evaluasi.
   Penelitian menggunakan beberapa model bahasa berbasis teks yang tersedia melalui API tanpa proses *fine-tuning*. Tugas yang digunakan mencakup penalaran numerik, penalaran logis, pertanyaan pengetahuan umum, serta skenario sosial dan moral.
4. Pelaksanaan eksperimen terotomatisasi.
   Setiap kombinasi persona, model, dan tugas dieksekusi menggunakan pendekatan *prompt-based evaluation*. Seluruh proses dijalankan secara otomatis untuk mengurangi variasi yang tidak diperlukan dan menjaga alur pengujian tetap seragam.
5. Pengolahan respons dan analisis perbandingan.
   Respons model dicatat dan dianalisis berdasarkan ketepatan jawaban serta pola perubahan respons yang muncul akibat perbedaan persona. Perbandingan antar model dilakukan untuk melihat sejauh mana masing-masing model peka terhadap perubahan persona.
6. Analisis bias.
   Analisis difokuskan pada *human bias* yang muncul selama proses tanya jawab akibat variasi persona. Penelitian ini tidak meninjau bias yang berasal dari data pelatihan atau arsitektur model.

Metodologi ini menjadi dasar untuk pelaksanaan eksperimen dan pembahasan pada bab selanjutnya.

BAB II

STUDI LITERATUR

II.1 Large Language Model

Large language model (LLM) adalah model berbasis arsitektur transformator yang dilatih menggunakan korpus teks dalam jumlah sangat besar. Melalui proses pelatihan ini, model mempelajari hubungan antar-token, pola semantik, serta variasi penggunaan bahasa yang umum ditemukan pada teks manusia (Bommasani, Hudson, Adeli, dkk. 2021). Dengan kemampuan tersebut, LLM dapat menghasilkan jawaban yang relevan meskipun hanya diberikan instruksi berbasis teks.

Salah satu ciri penting LLM adalah sensitivitasnya terhadap konteks. Model tidak hanya melihat makna literal suatu kata, tetapi juga memperhatikan gaya penulisan, struktur kalimat, dan isyarat pragmatik yang terdapat pada instruksi. Penelitian menunjukkan bahwa perubahan kecil dalam cara instruksi ditulis—misalnya perbedaan nada atau tingkat formalitas—dapat menghasilkan jawaban yang berbeda meskipun maksud pengguna tetap sama (Zhou dkk. 2023).

Sifat ini juga berpengaruh pada bentuk penalaran yang dihasilkan. Turpin (Turpin dkk. 2023) menemukan bahwa langkah penalaran LLM dapat berubah hanya karena variasi kecil dalam formulasi instruksi. Temuan ini menunjukkan bahwa LLM tidak selalu mengikuti jalur penalaran yang konsisten, tetapi menyesuaikannya dengan konteks linguistik yang diterima saat inferensi dilakukan.

Selain peka terhadap bahasa, LLM juga mempelajari pola sosial dari data pelatihan. Weidinger (Weidinger dkk. 2021) menunjukkan bahwa model dapat menyerap preferensi sosial atau bias yang muncul dalam korpus pelatihan. Dalam praktiknya, gaya penulisan pengguna dapat ditafsirkan sebagai sinyal identitas tertentu. Studi mengenai *role-playing* juga menunjukkan bahwa isyarat persona dapat memengaruhi gaya penjelasan, tingkat kehati-hatian, dan pola penalaran yang dihasilkan model

(Tseng dkk. 2024).

Karena persona merupakan variasi konteks linguistik yang dapat mengubah cara model menafsirkan instruksi, pemahaman mengenai sensitivitas ini menjadi dasar penting bagi penelitian. Pengaruh persona terhadap penalaran model perlu dianalisis secara sistematis pada berbagai LLM dengan ukuran dan karakteristik yang berbeda.

II.2 Persona dalam Interaksi Model Bahasa

Persona dalam konteks LLM merujuk pada karakteristik identitas yang tercermin melalui cara seseorang menulis atau berkomunikasi. Ciri ini dapat berupa pilihan kosakata, tingkat formalitas, struktur kalimat, atau pola penyampaian informasi. Dalam komunikasi manusia, perbedaan persona membantu lawan bicara menafsirkan maksud dan menyesuaikan respons. Hal serupa juga terjadi pada LLM karena model belajar dari data yang memuat berbagai gaya komunikasi (Tseng dkk. 2024).

Dalam interaksi dengan LLM, persona dapat muncul dalam dua bentuk, yaitu eksplisit dan implisit. Persona eksplisit muncul ketika identitas disebutkan secara langsung, seperti "Sebagai mahasiswa teknik informatika…". Informasi seperti ini memberikan sinyal identitas yang jelas sehingga model dapat menyesuaikan gaya atau struktur jawaban. Gupta (Gupta dkk. 2024) menunjukkan bahwa penugasan persona eksplisit dapat menghasilkan bentuk penalaran yang berbeda meskipun tugas yang diberikan sama.

Persona implisit muncul ketika model menyimpulkan identitas pengguna dari gaya penulisan tanpa adanya pernyataan langsung. Misalnya, gaya formal sering diasosiasikan dengan konteks akademis, sedangkan gaya santai lebih banyak ditemukan pada percakapan sehari-hari. Ketika pola tertentu muncul dalam instruksi, model dapat menafsirkannya sebagai identitas tertentu dan menyesuaikan jawabannya. Tseng (Tseng dkk. 2024) menunjukkan bahwa inferensi identitas seperti ini dapat terjadi hanya dari perbedaan pola bahasa.

Persona dapat memengaruhi dua aspek utama dalam respons model. Pertama, bentuk jawaban. Variasi persona dapat mengubah panjang penjelasan, pilihan kosakata, atau tingkat formalitas. Model cenderung menyesuaikan gaya jawaban agar sesuai dengan persona yang muncul pada instruksi.

Kedua, penalaran. Karena LLM peka terhadap formulasi instruksi (Zhou dkk. 2023) dan dapat menghasilkan langkah penalaran yang berbeda meskipun tugasnya sama

(Turpin dkk. 2023), perubahan persona dapat memicu variasi dalam cara model mencapai kesimpulan. Persona tertentu dapat membuat model menyusun penalaran yang lebih panjang atau lebih berhati-hati, sedangkan persona lain dapat menghasilkan penalaran yang lebih ringkas.

Penelitian mengenai bias juga menunjukkan bahwa persona dapat berinteraksi dengan pola sosial yang dipelajari model. Weidinger (Weidinger dkk. 2021) menunjukkan bahwa model dapat memperlakukan persona tertentu secara berbeda jika pola tersebut sering muncul dalam data pelatihan. Dalam beberapa situasi, persona dapat menggeser preferensi model dalam memilih sudut pandang atau jenis penjelasan yang diberikan (Gupta dkk. 2024).

Persona bukan sekadar tambahan informasi dalam instruksi, tetapi merupakan bagian dari konteks yang dipertimbangkan model ketika membentuk jawaban. Karena persona dapat mengubah gaya maupun penalaran model, analisis pengaruh persona menjadi penting untuk memahami konsistensi respons LLM pada kondisi identitas pengguna yang berbeda.

II.3 Pengaruh Persona terhadap Perilaku LLM

Persona dapat memengaruhi cara LLM memahami instruksi dan menghasilkan jawaban. Efek ini muncul karena model belajar dari data pelatihan yang berisi berbagai gaya bahasa dan situasi komunikasi. Ketika instruksi ditulis dengan gaya tertentu, model dapat menafsirkannya sebagai sinyal identitas dan menyesuaikan cara menjawab.

II.3.1 Pengaruh Persona terhadap Penalaran

Penelitian menunjukkan bahwa langkah penalaran LLM tidak selalu konsisten. Gupta (Gupta dkk. 2024) menemukan bahwa menambahkan persona eksplisit ke dalam instruksi dapat membuat model menghasilkan penalaran yang berbeda meskipun tugasnya sama. Perbedaan ini dapat terlihat pada urutan penjelasan, tingkat kehati-hatian, atau cara model menyusun argumen.

Pada persona implisit, perubahan penalaran muncul dari hal-hal yang lebih halus, seperti pilihan kata atau tingkat formalitas. Instruksi dengan gaya formal sering membuat model memberikan penjelasan yang lebih terstruktur. Sebaliknya, gaya penulisan yang santai dapat memicu jawaban yang lebih ringkas. Temuan Turpin (Turpin dkk. 2023) menunjukkan bahwa perubahan kecil pada formulasi instruksi

dapat mengubah langkah penalaran yang dihasilkan model. Kondisi ini membuat persona menjadi salah satu faktor yang dapat memicu perbedaan tersebut.

II.3.2 Pengaruh Persona terhadap Gaya Respons

Selain penalaran, persona juga memengaruhi cara model menyampaikan jawaban. Tseng (Tseng dkk. 2024) menunjukkan bahwa model dapat menyesuaikan gaya bahasa meskipun identitas pengguna tidak disebutkan secara langsung. Efek ini dapat terlihat dari panjang kalimat, tingkat formalitas, atau nada penjelasan.

Jika model mengaitkan persona tertentu dengan konteks profesional, respons yang diberikan cenderung lebih sistematis dan terstruktur. Sebaliknya, pada persona yang diasosiasikan dengan percakapan santai, jawaban yang muncul biasanya lebih singkat dan langsung.

II.3.3 Faktor yang Memengaruhi Efek Persona

Pengaruh persona dapat menjadi lebih kuat ketika framing instruksi konsisten. Selain itu, jenis tugas juga berperan. Pada tugas yang lebih terbuka, seperti skenario sosial, efek persona cenderung lebih terlihat dibandingkan pada tugas yang memiliki jawaban pasti. Ukuran dan kapasitas model juga memengaruhi sejauh mana persona berdampak terhadap respons. Model yang lebih besar umumnya lebih sensitif terhadap variasi gaya bahasa.

II.4 Bias dalam Respons LLM

Bias pada LLM muncul karena model belajar dari data yang mengandung kecenderungan tertentu. Selain informasi faktual, data pelatihan juga memuat pola sosial, stereotip, atau kebiasaan bahasa yang umum digunakan. Pola tersebut dapat terbawa ke dalam jawaban model.

II.4.1 Bentuk-bentuk Bias

Weidinger (Weidinger dkk. 2021) menunjukkan bahwa model dapat meniru stereotip yang ada pada data pelatihan. Bias seperti ini disebut bias representasional, misalnya ketika model menggambarkan suatu profesi atau kelompok sosial dengan cara yang tidak seimbang.

Selain itu, terdapat bias inferensial, yaitu ketika model menarik kesimpulan berdasarkan asosiasi yang tidak relevan. Model dapat menambahkan detail yang tidak

disebutkan pengguna hanya karena pola tersebut sering muncul dalam data pelatihan.

Bias penalaran juga dapat muncul. Gupta (Gupta dkk. 2024) menemukan bahwa persona tertentu dapat mendorong model menggunakan pola penjelasan tertentu yang tidak selalu muncul pada instruksi netral.

II.4.2 Dampak Bias terhadap Keluaran

Bias dapat memengaruhi ketepatan jawaban. Model dapat memberikan respons yang terdengar meyakinkan tetapi tidak sesuai dengan konteks yang diminta. Bias juga dapat memengaruhi panjang atau gaya penjelasan. Dalam beberapa kasus, model memberikan penjelasan lebih rinci kepada persona tertentu dan lebih singkat kepada persona lainnya.

Bias juga dapat memperkuat pola sosial tertentu secara tidak langsung. Misalnya, pemilihan kata atau nada penjelasan dapat mencerminkan kecenderungan tertentu tanpa disadari.

II.4.3 Kaitannya dengan Persona

Persona dapat memperkuat atau mengubah bias tersebut. Pada persona eksplisit, penyebutan identitas dapat memicu asosiasi tertentu yang pernah muncul dalam data pelatihan. Pada persona implisit, perubahan gaya bahasa dapat membuat model menafsirkan identitas tertentu meskipun tidak disebutkan secara langsung (Tseng dkk. 2024).

Penelitian Zhou (Zhou dkk. 2023) dan Turpin (Turpin dkk. 2023) menunjukkan bahwa LLM sangat sensitif terhadap perubahan formulasi instruksi. Karena persona merupakan bagian dari formulasi tersebut, variasi persona dapat menyebabkan perubahan dalam penjelasan atau penalaran yang dihasilkan model.

Pengaruh ini lebih terlihat pada tugas yang bersifat terbuka, seperti skenario sosial. Pada konteks seperti ini, ruang interpretasi yang lebih luas membuat bias dan persona lebih mudah memengaruhi hasil akhir.

II.5 Evaluasi Penalaran dan Benchmark

Evaluasi terhadap LLM umumnya dilakukan menggunakan benchmark yang dirancang untuk mengukur kemampuan penalaran dan pemahaman model secara lebih

terstruktur. Benchmark membantu memberikan gambaran mengenai performa model pada tugas yang bersifat konsisten dan terukur. Dalam penelitian ini, dua benchmark digunakan untuk menilai bagaimana persona dapat memengaruhi cara model menjawab, yaitu GSM8K dan MMLU-Redux.

II.5.1 GSM8K

GSM8K adalah kumpulan soal matematika tingkat sekolah dasar yang dirancang untuk menguji kemampuan penalaran numerik (Cobbe dkk. 2021). Setiap soal biasanya membutuhkan beberapa langkah pemikiran sederhana, seperti memahami konteks, melakukan perhitungan dasar, dan menarik kesimpulan. Bagi manusia, soal-soal ini relatif mudah, tetapi bagi LLM, benchmark ini menantang karena model harus menyusun langkah penyelesaian yang runtut.

Benchmark ini digunakan dalam penelitian untuk melihat apakah variasi persona dapat memengaruhi cara model membentuk langkah penalaran tersebut. Misalnya, persona tertentu dapat membuat model memberikan penjelasan lebih panjang, sementara persona lain mendorong model untuk menjawab lebih singkat. Dengan demikian, GSM8K memberikan konteks yang jelas untuk mengamati perubahan pada pola penalaran model.

II.5.2 MMLU-Redux

MMLU-Redux merupakan versi kurasi ulang dari benchmark MMLU yang berisi pertanyaan dari berbagai bidang pengetahuan (Edinburgh Dataset Analytics Working Group 2024). Tidak seperti GSM8K yang terfokus pada matematika dasar, MMLU-Redux mencakup berbagai kategori seperti sains, humaniora, dan ilmu sosial. Soal-soal dalam benchmark ini menguji kemampuan model dalam memahami konsep dan memilih jawaban yang paling tepat berdasarkan pengetahuan umum.

Benchmark ini digunakan dalam penelitian untuk melihat bagaimana persona dapat memengaruhi pilihan jawaban model, terutama pada pertanyaan yang memerlukan pemahaman konsep dan penalaran tingkat menengah. Karena format soal bersifat pilihan ganda, MMLU-Redux memberikan lingkungan evaluasi yang lebih terkontrol, sehingga perbedaan respons yang muncul lebih mudah diamati dari sisi persona.

II.5.3 Tantangan Evaluasi Berbasis Persona

Penggunaan benchmark dalam penelitian persona memiliki beberapa tantangan. Tantangan pertama adalah memastikan bahwa perubahan jawaban benar-benar dise-

babkan oleh persona, bukan oleh perbedaan formulasi instruksi. Karena LLM peka terhadap gaya penulisan (Zhou dkk. 2023), evaluasi perlu dilakukan dengan struktur prompt yang konsisten.

Tantangan berikutnya adalah variasi hasil yang terjadi antarpemanggilan model. LLM dapat menghasilkan jawaban berbeda meskipun instruksi yang diberikan sama (Turpin dkk. 2023). Oleh karena itu, proses evaluasi dilakukan secara terotomatisasi dan terstandarisasi agar hasil yang diperoleh lebih dapat dibandingkan.

Harapannya, benchmark GSM8K dan MMLU-Redux memberikan dasar yang jelas untuk melihat bagaimana persona dapat memengaruhi penalaran dan pilihan jawaban model dalam dua konteks yang berbeda, yaitu penalaran numerik dan pengetahuan umum.

II.6 Penelitian Terdahulu dan Kesenjangan Penelitian

Pembahasan mengenai persona dan perilaku LLM telah dibahas dalam beberapa penelitian sebelumnya. Secara umum, penelitian-penelitian tersebut menunjukkan bahwa identitas pengguna—baik yang dinyatakan secara langsung maupun tersirat melalui gaya penulisan—dapat memengaruhi cara model menghasilkan jawaban. Namun, sebagian besar studi masih terbatas pada jenis model atau bentuk persona tertentu sehingga gambaran mengenai pengaruh persona secara lebih luas belum banyak diuraikan.

II.6.1 Ringkasan Literatur Terkait

Gupta (Gupta dkk. 2024) menunjukkan bahwa pemberian persona eksplisit dapat mengubah langkah penalaran model pada tugas yang sama. Temuan ini memperlihatkan bahwa model tidak hanya memproses isi instruksi, tetapi juga memperhatikan informasi identitas yang disisipkan ke dalam prompt.

Tseng (Tseng dkk. 2024) menyoroti persona implisit yang muncul dari pilihan kata dan gaya penulisan. Dalam banyak kasus, model menafsirkan pola bahasa tersebut sebagai sinyal identitas tertentu dan menyesuaikan struktur responsnya. Studi ini memperlihatkan bahwa persona dapat terbentuk bahkan tanpa penyebutan identitas secara langsung.

Turpin (Turpin dkk. 2023) menemukan bahwa LLM dapat menghasilkan urutan penalaran yang berbeda hanya karena perubahan kecil dalam formulasi instruksi. Temuan ini menunjukkan bahwa penalaran model sangat dipengaruhi oleh konteks

linguistik yang diterima saat inferensi.

Dalam konteks bias, Weidinger (Weidinger dkk. 2021) menunjukkan bahwa model dapat meniru pola sosial atau stereotip yang ada dalam data pelatihan. Kondisi ini relevan ketika menilai pengaruh persona karena identitas tertentu dapat memperkuat pola bias yang sudah ada.

II.6.2 Keterbatasan Penelitian Sebelumnya

Meskipun penelitian sebelumnya memberikan kontribusi penting, sebagian besar studi masih memiliki beberapa keterbatasan. Pertama, banyak penelitian hanya menguji sedikit model sehingga belum memberikan gambaran mengenai bagaimana pengaruh persona dapat berbeda antar-LLM. Kedua, jumlah persona yang digunakan umumnya terbatas sehingga variasi efek persona belum terobservasi secara lebih luas. Ketiga, sebagian penelitian hanya menguji sedikit jenis tugas, padahal persona dapat memengaruhi model secara berbeda pada tugas numerik, pengetahuan umum, atau skenario sosial. Selain itu, tidak semua penelitian menggunakan kerangka evaluasi yang terstandarisasi sehingga sulit memastikan bahwa perubahan jawaban benar-benar disebabkan oleh persona.

II.6.3 Posisi dan Kontribusi Penelitian Ini

Penelitian ini disusun untuk mengatasi keterbatasan tersebut. Berbeda dari sebagian studi sebelumnya, penelitian ini menggunakan beberapa model dan beberapa persona untuk melihat bagaimana keduanya memengaruhi penalaran dan respons model. Penelitian ini juga menggunakan dua benchmark yang berbeda—GSM8K dan MMLU-Redux—agar pengaruh persona dapat diamati pada tugas numerik dan pengetahuan umum.

Selain itu, penelitian ini menggunakan *pipeline* evaluasi yang terotomatisasi sehingga setiap model menerima instruksi yang konsisten. Pendekatan ini membantu memastikan bahwa perbedaan yang muncul benar-benar berasal dari persona dan bukan dari variasi struktur prompt.

Dengan demikian, penelitian ini diharapkan dapat memberikan gambaran yang lebih menyeluruh mengenai bagaimana persona memengaruhi perilaku model bahasa, terutama ketika melibatkan beberapa model dan kategori tugas yang berbeda.

BAB III

ANALISIS MASALAH

III.1 Analisis Kondisi Saat Ini

Perkembangan *large language model* (LLM) dalam beberapa tahun terakhir mendorong pemanfaatan model bahasa dalam berbagai aplikasi, mulai dari penjawab pertanyaan, agen percakapan, hingga sistem pendukung pengambilan keputusan (Bommasani, Hudson, Adeli, dkk. 2021). Dengan penggunaan yang semakin luas, muncul kebutuhan untuk memahami bagaimana model merespons variasi identitas dan karakteristik pengguna, bukan hanya variasi instruksi tugas. Hal ini penting karena pada praktiknya, interaksi dengan LLM selalu membawa konteks mengenai siapa penggunanya dan dari gaya komunikasi seperti apa instruksi tersebut disampaikan.

Penelitian mengenai persona pada LLM sejauh ini lebih banyak menempatkan persona pada sisi model. Tseng et al. mengkaji berbagai pendekatan *role-playing* dan *personalization* yang memberikan identitas tertentu kepada model melalui instruksi sistem (Tseng dkk. 2024). Pada pengaturan ini, model diarahkan untuk meniru karakter, gaya bicara, atau peran tertentu, dan evaluasi dilakukan dengan menilai kesesuaian perilaku model terhadap persona tersebut. Fokus semacam ini berbeda dengan skenario ketika persona justru muncul dari sisi pengguna—melalui gaya penulisan, latar belakang yang dinyatakan, atau sinyal sosial lain yang terbawa dalam instruksi.

Di luar skenario *role-playing*, beberapa penelitian menunjukkan bahwa penyisipan persona eksplisit dapat memengaruhi penalaran model, termasuk pada soal penalaran formal yang tidak melibatkan konteks sosial. Gupta et al. menemukan bahwa identitas pengguna yang disebutkan dalam instruksi dapat mengubah cara model menyusun langkah penalaran dan memilih jawaban (Gupta dkk. 2024). Temuan ini menunjukkan bahwa persona tidak hanya memengaruhi pemilihan kosakata atau gaya respons, tetapi juga struktur penalaran yang digunakan model.

Selain itu, penalaran LLM terbukti sensitif terhadap variasi kecil pada formulasi instruksi. Turpin et al. memperlihatkan bahwa perubahan ringan pada *prompt* dapat menghasilkan rantai penalaran yang berbeda untuk pertanyaan yang sama (Turpin dkk. 2023). Sensitivitas terhadap framing juga ditunjukkan oleh Zhou et al., yang menemukan bahwa cara instruksi disusun dapat memengaruhi isi maupun gaya jawaban model (Zhou dkk. 2023). Kombinasi sifat ini membuat analisis persona menjadi lebih menantang, karena persona, framing, dan gaya penulisan sering hadir secara bersamaan dalam sebuah instruksi, sehingga pengaruh masing-masing sulit dipisahkan.

Lapisan kompleksitas lain muncul dari isu bias. Weidinger et al. menunjukkan bahwa LLM dapat mencerminkan pola bias sosial yang terdapat pada data pelatihan (Weidinger dkk. 2021). Ketika atribut sosial tertentu—seperti profesi, gender, atau latar budaya—muncul dalam instruksi, respons model berpotensi dipengaruhi oleh bias representasional maupun inferensial. Dalam konteks persona, hal ini berarti bahwa variasi respons tidak selalu mencerminkan perubahan kemampuan penalaran, tetapi dapat berasal dari bias yang telah terinternalisasi di dalam model.

Sementara itu, penelitian yang menempatkan persona pada sisi pengguna masih terbatas. Pendekatan pemodelan pengguna, seperti *user language model*, mulai dikembangkan untuk mempelajari variasi bahasa berdasarkan karakteristik pengguna (Naous, Roziere, dkk. 2025). Namun, kajian yang secara sistematis menilai pengaruh *user persona*—baik eksplisit maupun implisit—terhadap penalaran dan kualitas jawaban pada berbagai jenis tugas masih belum banyak dilakukan.

Dari sisi teknis, banyak studi persona masih mengandalkan eksekusi manual atau semiotomatis ketika menjalankan eksperimen. Naous et al. menyoroti pentingnya mekanisme evaluasi yang terstruktur, termasuk pengelolaan konfigurasi, pencatatan hasil, dan konsistensi skenario pengujian (Naous, Roziere, dkk. 2025). Tanpa kerangka evaluasi yang terdokumentasi dengan baik, eksperimen yang melibatkan banyak model, banyak persona, dan berbagai jenis tugas menjadi sulit direplikasi.

Berdasarkan kondisi tersebut, masalah-masalah utama yang melatarbelakangi penelitian ini dirangkum pada Tabel III.1.

Masalah M-01 berkaitan dengan kecenderungan penelitian sebelumnya yang lebih banyak menempatkan persona pada sisi model. Tseng et al. membahas bagaimana persona digunakan untuk mengubah gaya dan peran model melalui instruksi sistem (Tseng dkk. 2024). Pendekatan ini berbeda dengan skenario ketika identitas

Tabel III.1 Daftar masalah penelitian terkait *user persona* pada LLM

Kode	Uraian Masalah	Dampak terhadap Penelitian
M-01	Persona pada LLM umumnya diterapkan pada sisi model, bukan pada sisi pengguna.	Belum ada pemahaman sistematis mengenai bagaimana *user persona* eksplisit maupun implisit memengaruhi penalaran dan kualitas jawaban pada berbagai tugas.
M-02	Efek persona sulit dipisahkan dari efek framing dan gaya penulisan *prompt*.	Perubahan performa atau pola penalaran dapat berasal dari variasi formulasi instruksi, bukan semata akibat perubahan *user persona*, sehingga interpretasi hasil menjadi tidak pasti.
M-03	LLM membawa bias sosial yang terinternalisasi dari data pelatihan.	Ketika identitas pengguna memuat atribut sosial tertentu, respons model berpotensi mencerminkan bias representasional maupun inferensial, sehingga perbedaan jawaban bisa terkait dengan bias yang sudah ada pada model.
M-04	Cakupan model dan tugas pada studi terdahulu masih terbatas.	Analisis sensitivitas terhadap persona sering kali hanya mencakup sedikit model atau jenis tugas, sehingga belum memberikan gambaran yang cukup luas mengenai variasi perilaku LLM di berbagai konteks.

pengguna—baik eksplisit maupun implisit—menjadi bagian dari konteks interaksi. Akibatnya, pengaruh *user persona* terhadap penalaran dan kualitas jawaban belum banyak dikaji secara sistematis.

Masalah M-02 muncul karena struktur penalaran LLM sangat sensitif terhadap variasi kecil dalam formulasi instruksi. Turpin et al. menunjukkan bahwa perubahan ringan dalam *prompt* dapat menghasilkan rantai penalaran yang berbeda meskipun pertanyaannya sama (Turpin dkk. 2023). Zhou et al. juga memperlihatkan bahwa framing dan gaya penulisan instruksi dapat memengaruhi isi dan gaya jawaban (Zhou dkk. 2023). Untuk itu, penelitian yang menilai pengaruh persona perlu dirancang sedemikian rupa agar dapat membedakan pengaruh persona dari pengaruh framing.

Masalah M-03 berhubungan dengan bias sosial yang sudah tertanam di dalam model. Weidinger et al. menunjukkan bahwa LLM dapat mereproduksi dan memperkuat bias yang terdapat pada data pelatihan (Weidinger dkk. 2021). Ketika *user persona* memuat atribut sosial tertentu, respons model dapat dipengaruhi oleh bias tersebut. Hal ini membuat interpretasi hasil menjadi lebih rumit karena variasi jawaban bisa berasal dari interaksi antara persona dan bias model.

Masalah M-04 menyoroti keterbatasan cakupan model dan tugas pada penelitian persona sebelumnya. Banyak studi hanya menguji sedikit model atau fokus pada

satu jenis tugas, sehingga belum memberikan gambaran yang lebih luas mengenai bagaimana variasi *user persona* memengaruhi perilaku model pada berbagai kategori tugas (Gupta dkk. 2024; Tseng dkk. 2024). Kondisi ini membuka peluang untuk merancang eksperimen dengan cakupan multi model dan multi persona.

III.2 Analisis Kebutuhan

III.2.1 Identifikasi Masalah Pengguna

Pengguna dalam penelitian ini adalah peneliti yang ingin mengevaluasi perilaku model bahasa di bawah variasi *user persona*. Berdasarkan kondisi yang telah dibahas sebelumnya, beberapa kebutuhan dasar dapat diidentifikasi sebagai berikut.

1. Belum tersedia cara yang terstruktur untuk merumuskan *user persona* eksplisit maupun implisit pada sisi pengguna. Literatur yang ada umumnya berfokus pada persona di sisi model, sehingga peneliti perlu menyusun sendiri definisi persona yang diperlukan dalam eksperimen.
2. Perbedaan respons model dapat dipengaruhi oleh variasi kecil pada formulasi pertanyaan. Hal ini menyulitkan proses analisis, karena tidak selalu jelas apakah perubahan jawaban disebabkan oleh persona atau oleh perbedaan cara instruksi disampaikan.
3. Eksperimen yang melibatkan lebih dari satu model dan beberapa kategori tugas membutuhkan prosedur yang memungkinkan skenario yang sama dijalankan kembali dan hasilnya dicatat secara konsisten, sehingga perbandingan antar kondisi dapat dilakukan secara sistematis.

Identifikasi ini menjadi dasar penyusunan kebutuhan fungsional dan nonfungsional penelitian.

III.2.2 Kebutuhan Fungsional

Kebutuhan fungsional merujuk pada kemampuan yang perlu tersedia agar eksperimen dapat berjalan sesuai tujuan. Kebutuhan tersebut ditampilkan pada Tabel III.2.

KF-01 berfungsi memastikan bahwa definisi persona disusun secara terstandar. KF-02 memungkinkan skenario eksperimen diterapkan secara konsisten pada beberapa kondisi. KF-03 menyediakan dasar pencatatan yang mendukung proses analisis baik secara kuantitatif maupun kualitatif.

Tabel III.2 Kebutuhan fungsional penelitian

Kode	Uraian kebutuhan fungsional	Terkait masalah
KF-01	Mekanisme untuk mendefinisikan *user persona* eksplisit dan implisit dalam bentuk skenario teks yang seragam, sehingga persona dapat dirancang secara konsisten dan digunakan kembali.	M-01
KF-02	Mekanisme untuk menjalankan pertanyaan yang sama pada beberapa persona dan beberapa model, serta mencatat respons berikut informasi persona, model, dan jenis tugas.	M-02, M-04
KF-03	Format pencatatan hasil yang mendukung penilaian sederhana seperti benar–salah dan indikasi bias, sehingga keluaran model dapat dianalisis lebih lanjut tanpa perlakuan tambahan yang kompleks.	M-03, M-04

III.2.3 Kebutuhan Nonfungsional

Kebutuhan nonfungsional berkaitan dengan kualitas pelaksanaan eksperimen dan sifat teknis dari kerangka kerja yang digunakan. Daftar kebutuhan nonfungsional ditunjukkan pada Tabel III.3.

Tabel III.3 Kebutuhan nonfungsional penelitian

Kode	Jenis kebutuhan	Uraian kebutuhan
KNF-01	Reproducibility	Seluruh rangkaian eksperimen dapat dijalankan ulang melalui skrip atau konfigurasi yang terdokumentasi, sehingga model, persona, dan tugas dapat diuji kembali dalam kondisi yang serupa.
KNF-02	Simplicity	Pelaksanaan eksperimen dapat dilakukan dengan langkah-langkah yang langsung dan tidak memerlukan infrastruktur tambahan di luar pemanggilan API atau prosedur serupa.
KNF-03	Extensibility	Struktur eksperimen memungkinkan penambahan model atau persona baru tanpa perubahan besar pada kerangka yang sudah ada, sehingga penelitian dapat dikembangkan lebih lanjut sesuai kebutuhan.

KNF-01 memastikan eksperimen dapat diujikan kembali dalam kondisi yang sama. KNF-02 menekankan agar implementasi tidak menimbulkan kompleksitas teknis yang tidak diperlukan. KNF-03 memberi fleksibilitas untuk memperluas cakupan

eksperimen pada tahap selanjutnya.

III.3 Analisis Pemilihan Solusi

Bagian ini membahas alternatif pendekatan yang dapat digunakan untuk melaksanakan eksperimen *multi model* dan *multi persona*, kemudian menjelaskan dasar pemilihan solusi yang digunakan dalam penelitian. Analisis dilakukan dengan mempertimbangkan kebutuhan representasi *user persona*, konsistensi eksekusi lintas model dan tugas, kemudahan pencatatan hasil, serta tingkat kerumitan implementasi.

III.3.1 Alternatif Solusi

Berdasarkan kebutuhan yang telah dirumuskan pada Subbagian 3.2, beberapa alternatif solusi yang dapat dipertimbangkan adalah sebagai berikut.

1. Evaluasi manual melalui antarmuka percakapan.
   Interaksi dengan *large language model* dilakukan langsung melalui antarmuka percakapan yang disediakan oleh penyedia layanan. *User persona* disisipkan ke dalam instruksi, pertanyaan dijalankan satu per satu, dan hasil dicatat secara manual. Alternatif ini mudah digunakan pada tahap awal, tetapi tidak efisien ketika jumlah kombinasi skenario menjadi besar. Prosesnya rentan terhadap variasi formulasi instruksi dan bergantung pada ketelitian pencatatan, sehingga menyulitkan replikasi dengan kondisi yang sama.
2. Skrip eksperimen semi terotomatisasi berbasis konfigurasi.
   Pada alternatif ini, daftar persona, model, dan kumpulan tugas (misalnya GSM8K dan MMLU-Redux) disimpan dalam berkas konfigurasi yang terstruktur. Skrip eksperimen membaca konfigurasi tersebut, menyusun *prompt* untuk setiap kombinasi skenario, memanggil model melalui API, lalu menyimpan keluaran beserta metadata ke dalam berkas JSON. Tahap analisis kemudian mengolah JSON menjadi keluaran yang lebih ringkas, seperti CSV, untuk perhitungan metrik dan evaluasi lanjutan. Pendekatan ini memerlukan penulisan skrip, tetapi memberikan struktur yang rapi dan mendukung eksekusi dalam jumlah besar.
3. Kerangka evaluasi umum yang dapat digunakan kembali.
   Alternatif ini merupakan perluasan dari pendekatan kedua dengan membangun kerangka evaluasi yang lebih lengkap, misalnya berupa pustaka atau layanan khusus. Fitur yang disediakan dapat mencakup penjadwalan eksekusi, pengelolaan versi konfigurasi, penilaian otomatis, hingga visualisasi hasil. Pendekatan ini cenderung lebih fleksibel untuk penggunaan jangka panjang,

tetapi memerlukan usaha perancangan dan implementasi yang cukup besar untuk konteks tugas akhir.

III.3.2 Analisis Penentuan Solusi

Ketiga alternatif dibandingkan berdasarkan beberapa kriteria, yaitu kemampuan merepresentasikan skenario eksperimen secara terstruktur, konsistensi eksekusi, dukungan pencatatan metadata, keterulangan (*reproducibility*), tingkat kerumitan implementasi, serta kemudahan menambahkan model atau persona baru. Ringkasan perbandingan ditunjukkan pada Tabel III.4.

Tabel III.4 Perbandingan alternatif solusi

Kriteria	Evaluasi Manual	Skrip Semi-Otomatis	Kerangka Evaluasi Umum
Representasi *user persona* dan skenario terstruktur	Rendah	Tinggi	Tinggi
Konsistensi eksekusi lintas model dan tugas	Rendah	Tinggi	Tinggi
Pencatatan hasil dan metadata	Rendah	Tinggi	Tinggi
Keterulangan eksperimen	Rendah	Tinggi	Tinggi
Kerumitan implementasi dan pemeliharaan	Rendah	Sedang	Tinggi
Kemudahan penambahan model atau persona baru	Rendah	Tinggi	Tinggi

Pendekatan evaluasi manual mudah digunakan, tetapi tidak memenuhi kebutuhan eksperimen dengan banyak kombinasi model dan persona. Keterbatasan terutama terlihat pada konsistensi eksekusi, dokumentasi hasil, serta kesulitan mengulang percobaan dengan kondisi identik.

Pendekatan kerangka evaluasi umum menyediakan fleksibilitas yang lebih luas, tetapi memerlukan usaha perancangan dan implementasi yang cukup besar. Beban tersebut dapat mengalihkan fokus dari tujuan utama penelitian.

Pendekatan skrip eksperimen semi terotomatisasi berbasis konfigurasi menawarkan keseimbangan yang paling sesuai. Representasi model, persona, dan tugas dapat diatur dalam direktori konfigurasi, sedangkan proses eksekusi dan analisis dijalankan melalui skrip yang konsisten. Seluruh keluaran disimpan dalam format terstruktur sehingga mudah dianalisis kembali. Struktur seperti ini mendukung keterulangan eksperimen dan perluasan skenario tanpa memerlukan pembangunan kerangka yang kompleks.

Berdasarkan pertimbangan tersebut, penelitian ini menggunakan pendekatan skrip eksperimen semi terotomatisasi berbasis konfigurasi sebagai solusi utama dalam

melaksanakan eksperimen *multi model* dan *multi persona*.

BAB IV

DESAIN KONSEP SOLUSI

IV.1 Desain Konseptual Eksperimen

Bagian ini menjelaskan landasan perancangan eksperimen yang digunakan dalam penelitian. Desain ini disusun untuk melihat bagaimana dua bentuk persona, yaitu persona eksplisit dan persona implisit, memengaruhi hasil keluaran pada beberapa kategori tugas penalaran dan beberapa sistem yang berbeda. Penyusunan bagian ini dimaksudkan untuk memastikan bahwa setiap variasi yang muncul dapat ditelusuri kembali pada kondisi persona yang digunakan, bukan pada perbedaan situasi pengujian atau susunan instruksi.

IV.1.1 Tujuan Perancangan Eksperimen

Perancangan eksperimen dilakukan untuk menyediakan kerangka yang memungkinkan perbandingan persona secara terarah. Dua bentuk persona digunakan karena mewakili dua pola interaksi yang umum terjadi, yaitu ketika identitas pengguna dinyatakan secara langsung serta ketika identitas tersebut tersirat melalui cara bertutur. Kerangka ini juga dirancang agar dapat digunakan untuk membandingkan respons dari beberapa sistem secara konsisten pada jenis tugas yang sama.

IV.1.2 Komponen Utama Eksperimen

Eksperimen yang dilakukan mengombinasikan tiga komponen utama, yaitu persona, sistem, dan tugas penalaran. Persona mencakup bentuk eksplisit dan implisit, yang masing-masing memberikan konteks pengguna dengan kedalaman dan cara penyampaian yang berbeda. Komponen sistem terdiri atas beberapa model yang tersedia melalui layanan API sehingga memungkinkan analisis lintas arsitektur. Tugas penalaran yang digunakan mencakup penalaran numerik dan penalaran lintas topik untuk melihat bagaimana bentuk persona memengaruhi keluaran pada sifat

tugas yang berbeda.

IV.1.3 Prinsip Pengendalian Variabel

Untuk menjaga kesetaraan pengujian, seluruh instruksi disampaikan menggunakan susunan yang seragam pada setiap kombinasi persona, sistem, dan tugas. Dengan demikian, unsur yang bervariasi hanyalah bentuk persona. Pendekatan ini dilakukan agar hasil yang diperoleh dapat dibandingkan secara langsung tanpa dipengaruhi oleh variasi lain di luar persona.

IV.1.4 Ruang Konfigurasi

Ruang eksperimen dibentuk berdasarkan kombinasi antara persona, sistem, dan tugas penalaran. Setiap elemen didefinisikan melalui berkas konfigurasi sehingga struktur ruang eksperimen terdokumentasi dengan jelas dan dapat diperluas apabila diperlukan. Dengan adanya pengaturan ini, seluruh kondisi yang diuji dapat ditelusuri kembali dan dianalisis berdasarkan konfigurasi yang digunakan.

IV.1.5 Keterkaitan dengan Pelaksanaan Eksperimen

Desain konseptual ini menjadi dasar bagi alur pelaksanaan yang dibahas pada bagian berikutnya. Dengan pemisahan antara tahap perancangan dan tahap pelaksanaan, eksperimen dapat dijalankan secara teratur, dan seluruh hasil yang diperoleh dapat dianalisis kembali pada bab selanjutnya.

IV.2 Arsitektur *Evaluation Pipeline* dan Alur Pelaksanaan Eksperimen

Bagian ini menjelaskan bagaimana rancangan konseptual pada Subbab sebelumnya direalisasikan dalam bentuk arsitektur *evaluation pipeline* yang terotomatisasi, serta bagaimana pipeline tersebut menjalankan alur eksperimen dari pemuatan *specification* hingga diperolehnya keluaran akhir. Pendekatan ini dirancang agar proses evaluasi berjalan secara otomatis, konsisten, dan dapat direproduksi, sehingga setiap kombinasi persona, model, dan *benchmark task* diuji dalam kondisi yang setara dan bebas dari variasi yang tidak diperlukan.

Pipeline bekerja sebagai rangkaian komponen yang saling berinteraksi: mulai dari pemuatan data, konstruksi instruksi, pengiriman permintaan ke model, hingga pencatatan *telemetry*. Seluruh proses tersebut membentuk satu alur terintegrasi yang mampu menangani jumlah evaluasi besar secara stabil.

IV.2.1 Arsitektur Alur Kerja Sistem

Secara garis besar, *evaluation pipeline* terbagi ke dalam empat komponen utama yang membentuk satu siklus pemrosesan berulang untuk setiap kombinasi persona dan butir soal. Keempat komponen tersebut adalah sebagai berikut.

1. *Configuration initialization and validation*.
   Tahap ini memuat seluruh konfigurasi sistem, definisi persona, dan *benchmark dataset* ke dalam memori. Struktur data yang dibaca dari berkas *specification* (persona, model, dan *task*) divalidasi untuk memastikan bahwa setiap persona memiliki *system instruction* yang lengkap dan setiap butir tugas memiliki pasangan pertanyaan dan jawaban acuan. Validasi awal ini penting untuk mencegah kesalahan format yang dapat menghentikan proses pada tahap berikutnya.
2. *Prompt construction engine*.
   Pada tahap ini, sistem membentuk dua jenis pesan utama: *system message* yang berisi identitas dan karakter persona, serta *user message* yang memuat pertanyaan dari *benchmark*. Penyusunan instruksi dilakukan menggunakan pola yang seragam untuk seluruh iterasi, sehingga setiap model menerima bentuk stimulus yang konsisten. Pendekatan ini menghilangkan variasi yang berasal dari perbedaan penulisan instruksi manual, sehingga perubahan keluaran dapat dikaitkan pada persona, bukan pada redaksi *prompt*.
3. *Execution manager*.
   Komponen ini mengatur pengiriman permintaan ke model-model bahasa melalui *API interface*. Untuk mengatasi volume permintaan yang besar, *execution manager* menggunakan pendekatan eksekusi asinkron berbasis *I/O concurrency*. Permintaan disusun dalam *task queue* dan dieksekusi dalam kelompok sesuai batas *rate limit* dari penyedia layanan model. Strategi ini mempercepat proses pengujian tanpa melampaui kapasitas layanan.
4. *Telemetry logger*.
   Komponen terakhir bertanggung jawab menyimpan seluruh respons model dalam format terstruktur, termasuk keluaran teks, jawaban akhir yang diekstraksi, jumlah token yang digunakan, serta *latency* inferensi. Data ini menjadi dasar analisis performa pada Bab V, baik dari sisi akurasi maupun beban komputasi.

Dengan pembagian tersebut, pipeline dapat beroperasi secara modular, namun tetap terpadu dalam satu alur pemrosesan yang deterministik.

IV.2.2 Algoritma Orkestrasi dan Konkurensi

Eksperimen dalam penelitian ini melibatkan ribuan kombinasi persona–model–pertanyaan yang menghasilkan volume permintaan API dalam jumlah besar. Eksekusi secara sekuensial tidak praktis karena setiap permintaan memiliki latensi yang bervariasi, sementara penyedia model menerapkan batas *rate limit* yang ketat. Untuk mengatasi hal tersebut, pipeline menggunakan pendekatan eksekusi asinkron berbasis *I/O concurrency*.

Pendekatan ini memungkinkan banyak permintaan dieksekusi secara paralel (hingga batas tertentu), sehingga waktu total dapat ditekan dari kompleksitas $O(N)$ menjadi mendekati $O(N/C)$, dengan C adalah kapasitas konkurensi maksimum. Pipeline membangun sebuah *task queue* yang berisi seluruh pasangan persona–soal, kemudian memprosesnya dalam kelompok (*batch*) sesuai kapasitas konkurensi. Ketika satu batch sedang diproses, sistem dapat menyiapkan batch berikutnya tanpa menunggu seluruh permintaan selesai.

Selain meningkatkan efisiensi waktu, mekanisme ini juga menyediakan ketahanan terhadap kesalahan. Jika terjadi galat seperti *timeout*, *connection reset*, atau `429 Too Many Requests`, pipeline tidak menghentikan seluruh proses. Tugas yang gagal akan dicatat dan dijalankan ulang menggunakan strategi *exponential backoff*, sehingga stabilitas eksekusi jangka panjang tetap terjaga.

Algoritma 4.1 berikut mendefinisikan prosedur eksekusi paralel secara formal.

```
Algoritma 4.1: Prosedur Eksekusi Eksperimen Paralel

Input : Himpunan Persona P, Himpunan Tugas T, Batas Konkurensi C
Output: Himpunan Log L

Function RunExperiment(P, T):
  1. Inisialisasi Antrean Tugas Q <- Kosong
  2. Untuk setiap p dalam P lakukan:
       Untuk setiap t dalam T lakukan:
         Prompt <- ConstructPrompt(p.instruction, t.question)
         Enqueue(Q, Prompt)

  3. Inisialisasi Semaphore S dengan kapasitas C
```

```
4. While Q tidak kosong lakukan secara Asinkron:
     Batch <- DequeueBatch(Q, C)
     Untuk setiap item i dalam Batch lakukan secara Paralel:
       Acquire(S)
       Coba:
         Respons <- AsyncCallAPI(i.prompt, i.config)
         Metadata <- ExtractTelemetry(Respons)
         SaveLog(Respons, Metadata)
         Tambahkan ke L
       Tangkap Galat:
         LogGalat(i)
         RetryWithBackoff(i)
       Akhirnya:
         Release(S)

5. Return L
```

Melalui orkestrasi ini, pipeline mencapai dua tujuan sekaligus: (1) efisiensi waktu eksekusi yang optimal berkat pemrosesan paralel, dan (2) ketahanan proses melalui penanganan galat adaptif.

IV.2.3 Mekanisme Injeksi Konteks Persona

Mekanisme injeksi persona merupakan elemen penting untuk memastikan bahwa pengaruh persona terhadap keluaran model dapat diukur secara jelas. Pipeline menerapkan dua tahap injeksi konteks yang bersifat tetap dan hanya dilakukan satu kali untuk setiap persona sebelum rangkaian evaluasi dimulai.

Tahap pertama adalah *persona context initialization*. Pada tahap ini, sistem menyusun *system message* yang merangkum identitas dan karakter persona, baik dalam bentuk eksplisit maupun implisit sebagaimana didefinisikan pada Subbab IV.1. Pesan ini berfungsi membangun *cognitive framing* awal pada model sehingga konteks persona tertanam sebelum tugas utama diberikan.

Tahap kedua adalah *persona warm-up message*. Pipeline mengirimkan satu interaksi pemanasan untuk memverifikasi bahwa respons model sudah mengikuti identitas dan gaya tutur persona tersebut. Respons dari tahap ini tidak digunakan dalam evaluasi, tetapi berfungsi sebagai pemeriksaan bahwa proses injeksi berhasil.

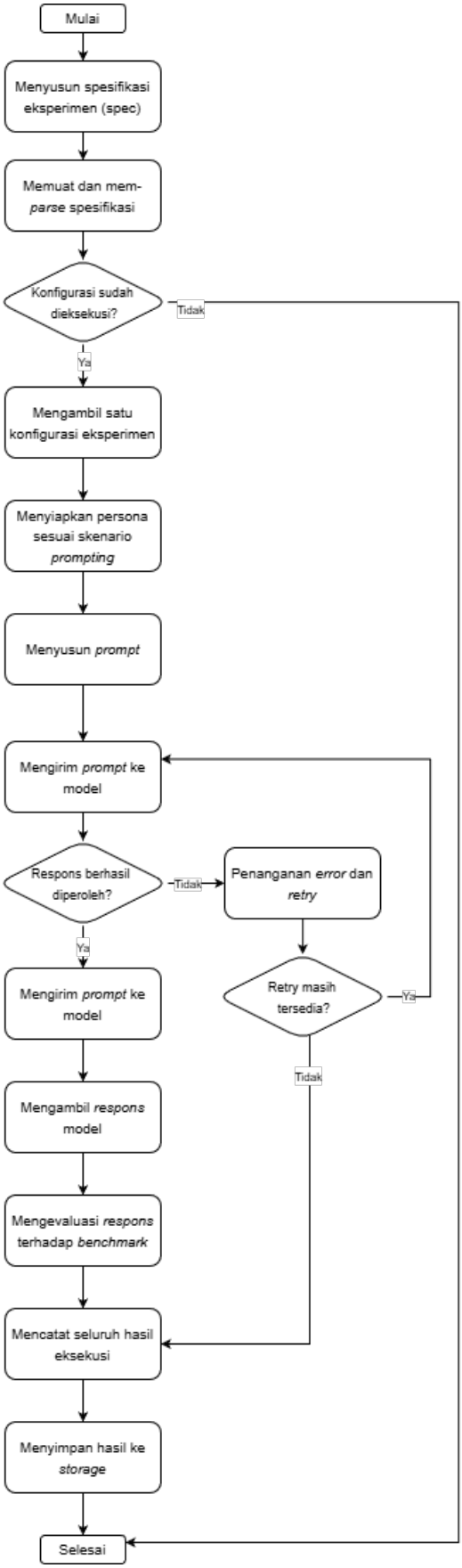
Setelah kedua tahap ini selesai, pipeline tidak lagi mengulangi injeksi persona untuk setiap pertanyaan. Identitas yang telah ditanamkan pada awal percakapan tetap digunakan selama seluruh rangkaian pengujian. Model kemudian langsung memproses seluruh soal pada GSM8K dan MMLU-Redux dalam kondisi persona yang sama. Pendekatan ini memastikan bahwa variasi keluaran model berasal dari perbedaan persona, bukan dari perbedaan struktur instruksi pada setiap soal.

IV.2.4 Alur Operasional Pelaksanaan Eksperimen

Secara operasional, alur pelaksanaan eksperimen mengikuti rangkaian langkah yang digambarkan pada Gambar IV.1. Diagram tersebut menunjukkan hubungan antara pembentukan *configuration*, penyusunan *instruction*, eksekusi *task*, dan pencatatan hasil dalam satu siklus pipeline.

Pelaksanaan eksperimen dilakukan melalui langkah-langkah berikut.

1. Memuat *specification*.
   System membaca berkas *specification* yang memuat daftar persona, daftar model, daftar *task*, serta aturan eksekusi. Informasi tersebut diproses menjadi dasar pembentukan himpunan *configuration* yang akan dievaluasi.
2. Membentuk *configuration* lengkap.
   Seluruh kombinasi persona, model, dan *task* dibentuk sebagai unit eksekusi dan dicatat untuk dijalankan selama eksperimen. Setiap *configuration* menyimpan identitas persona, model, dan penanda butir soal yang terkait.
3. Memilih satu *configuration*.
   System mengambil satu *configuration* dari antrean tugas pada setiap siklus dan menjadwalkannya untuk dieksekusi hingga seluruh kombinasi selesai diproses.
4. Menerapkan *persona*.
   Persona yang sesuai dengan *configuration* tersebut diterapkan terlebih dahulu agar *task* diproses dalam konteks pengguna yang telah ditetapkan. Pada beberapa kondisi, digunakan satu interaksi *warmup* untuk memastikan bahwa respons awal model sudah mengikuti karakter persona sebelum rangkaian *task* utama dikirimkan.
5. Menyusun *instruction* untuk *task*.
   Instruction dirumuskan dengan susunan yang seragam oleh *prompt construction engine*, sehingga perbedaan hasil dapat dikaitkan pada variasi persona dan model, bukan pada perbedaan redaksi atau struktur penyampaian.
6. Mengirim *instruction* kepada model.



Gambar IV.1 Diagram alur pelaksanaan eksperimen

Instruction yang telah lengkap dikirimkan kepada model melalui *execution manager* untuk memperoleh *response* yang digunakan dalam tahap analisis.

7. Penanganan kegagalan.
   Jika *response* tidak diperoleh atau terjadi gangguan sementara, *instruction* dijadwalkan ulang menggunakan jeda adaptif hingga *response* valid diterima. Mekanisme ini memastikan seluruh *configuration* menghasilkan keluaran yang dapat digunakan.
8. Mencatat hasil *response*.
   Response yang diterima disimpan oleh *telemetry logger* dalam berkas penyimpanan bersama informasi pendukung lainnya, seperti jumlah token dan *latency*, untuk keperluan analisis.
9. Melanjutkan ke *configuration* berikutnya.
   Setelah satu *configuration* selesai, *system* beralih ke *configuration* berikutnya hingga seluruh ruang eksperimen selesai dievaluasi.

Dengan arsitektur dan alur operasional ini, eksperimen dapat dijalankan secara teratur, terukur, dan setiap hasil yang dihasilkan dapat ditelusuri kembali berdasarkan *configuration* yang digunakan.

IV.3 Integrasi Komponen Eksperimen

Bagian ini menjelaskan komponen-komponen yang digunakan dalam eksperimen, yang terdiri atas *benchmark* penalaran, himpunan model, struktur persona, ruang *configuration*, serta contoh mekanisme injeksi persona. Seluruh komponen tersebut didefinisikan melalui berkas *specification* sehingga dapat digunakan secara konsisten pada seluruh tahapan eksperimen.

IV.3.1 Benchmark Penalaran

Eksperimen menggunakan dua *benchmark* yang mewakili dua bentuk kemampuan penalaran.

Benchmark pertama adalah *GSM8K*, yang berisi soal cerita matematika tingkat sekolah menengah. *Benchmark* ini menilai kemampuan sistem dalam melakukan penalaran numerik bertahap. Setiap soal memiliki jawaban numerik yang jelas sehingga pemeriksaan hasil dapat dilakukan secara deterministik (Cobbe dkk. 2021).

Benchmark kedua adalah *MMLU-Redux*, versi terkurasi dari MMLU yang memperbaiki ketidakkonsistenan format dan pilihan jawaban. *Benchmark* ini digunakan untuk menilai penalaran lintas topik dalam format pilihan ganda, meliputi bidang sa-

ins, matematika, humaniora, dan ilmu sosial (Edinburgh Dataset Analytics Working Group 2024).

Penggunaan kedua *benchmark* tersebut memberikan cakupan dua bentuk penalaran yang berbeda, yaitu penalaran numerik prosedural dan penalaran konseptual deklaratif.

IV.3.2 Himpunan Model

Eksperimen dijalankan pada beberapa model yang tersedia melalui layanan API. Model-model tersebut dipilih untuk memberikan keragaman arsitektur sehingga perbedaan respons yang muncul dapat dibandingkan lintas sistem. Model yang digunakan meliputi:

1. Model komersial GPT-5 Mini, Claude 4.5 Haiku, Gemini 2.5 Flash, Llama 3.3 Nemotron Super 49B V1.5, Google Gemma 3n 4B, dan DeepSeek V3.2
2. Model publik Grok 4.1 Fast, NVIDIA Nemotron-nano-12B-v2-VL, dan Bert Nebulon Alpha.

Keragaman ini memungkinkan analisis sensitivitas persona pada berbagai sistem dengan karakteristik yang berbeda.

IV.3.3 Struktur Persona

Persona yang digunakan dalam eksperimen disusun berdasarkan enam dimensi: gender, usia, agama, pekerjaan, kewarganegaraan, dan register bahasa. Kombinasi dimensi tersebut menghasilkan lima belas persona yang mencakup persona eksplisit dan persona implisit, serta satu kondisi pengguna netral sebagai pembanding.

Tabel IV.1 menyajikan daftar lengkap persona yang digunakan.

Tabel IV.1 Daftar persona pada kondisi eksperimen

ID	Persona	Mode	Gender	Age Group	Religion	Occupation	Nationality / Register
P1	Implicit male baseline	Implicit	Male	-	-	-	Neutral
P2	Implicit female baseline	Implicit	Female	-	-	-	Neutral
P3	Neutral user	Neutral	-	-	-	-	Neutral
P4	Indonesian Muslim young woman	Explicit	Female	Young adult	Muslim	Healthcare worker	Indonesian / Semi-formal
P5	Indonesian Muslim young man	Implicit	Male	Young adult	Muslim	Healthcare worker	Indonesian / Semi-formal
P6	American middle-aged male	Explicit	Male	Middle-aged	Christian	Engineer	American / Formal
P7	American middle-aged female	Implicit	Female	Middle-aged	Christian	Engineer	American / Formal
P8	Indonesian Gen-Z female	Explicit	Female	Gen-Z	-	Student	Indonesian / Casual-slang
P9	Indonesian Gen-Z male	Implicit	Male	Gen-Z	-	Student	Indonesian / Casual-slang
P10	Middle Eastern young adult male	Explicit	Male	Young adult	Muslim	Engineer	Middle Eastern Arabic / Formal
P11	Middle Eastern young adult female	Implicit	Female	Young adult	Muslim	Student	Middle Eastern Arabic / Formal
P12	American atheist young male	Explicit	Male	Young adult	Atheist	Student	American / Formal
P13	American atheist young female	Implicit	Female	Young adult	Atheist	Student	American / Formal
P14	Indonesian female healthcare worker	Explicit	Female	Young adult	Muslim	Healthcare worker	Indonesian / Semi-formal
P15	Indonesian male healthcare worker	Implicit	Male	Young adult	Muslim	Healthcare worker	Indonesian / Semi-formal

IV.3.4 Ruang *Configuration*

Kombinasi lima belas persona dan sembilan model membentuk seratus tiga puluh lima *configuration*. Setiap *configuration* merepresentasikan satu pasangan persona dan model yang kemudian diuji pada himpunan *task* yang sama. Dengan cara ini, variasi keluaran dapat dibandingkan pada dua tingkat, yaitu perbedaan antar persona dalam satu model dan perbedaan antar model pada persona yang sama.

Untuk menjaga keteraturan proses, setiap *configuration* melewati urutan eksekusi yang tetap. Urutan tersebut meliputi penerapan persona pada awal percakapan, penyiapan konteks interaksi, pelaksanaan *benchmark* pada himpunan soal yang telah ditetapkan, serta pencatatan hasil dan informasi pendukung. Pola yang berulang ini memudahkan penelusuran kembali setiap hasil ke persona, model, dan *task* yang digunakan.

IV.3.5 Contoh Mekanisme Injeksi Persona

Persona diterapkan melalui *system message* yang dikirim sebelum *task* utama diberikan. Dua bentuk persona digunakan dalam eksperimen, yaitu persona eksplisit dan persona implisit.

Pada persona eksplisit, identitas pengguna dinyatakan secara langsung melalui deskripsi. Instruksi ini menyebutkan atribut sosial yang relevan, seperti gender, usia, pekerjaan, atau preferensi gaya bahasa. Contoh yang digunakan dalam eksperimen adalah sebagai berikut.

> *"Your user is an Indonesian Gen-Z male who works as a junior engineer. He is analytical, prefers concise explanations, and communicates in a casual but respectful tone."*

Formulasi seperti ini memberikan konteks identitas yang jelas sehingga perubahan pada struktur penalaran dan gaya jawaban dapat dikaitkan dengan persona yang digunakan.

Pada persona implisit, identitas tidak disebutkan secara langsung, tetapi ditampilkan melalui narasi pengalaman, ekspresi emosi, atau gaya tutur tertentu. Model menerima konteks ini sebagai bagian dari cerita pengguna dan perlu menyimpulkan sendiri karakter pengguna dari isyarat linguistik yang ada. Contoh yang digunakan dalam eksperimen adalah sebagai berikut.

> *"Lately I have been feeling a strange mix of emotional exhaustion and pressure to appear composed, especially when my skin starts acting up unexpectedly. Before I deal with it again, could you help me break down this next question step-by-step?"*

Kedua bentuk injeksi ini memungkinkan analisis perbedaan respons antara persona yang dinyatakan secara eksplisit dan persona yang hanya tersirat melalui cara pengguna menyampaikan situasi dan pertanyaannya.

IV.4 Perancangan Data dan Struktur Berkas

Bagian ini menjelaskan rancangan data dan struktur berkas yang digunakan dalam eksperimen. Perancangan ini diperlukan agar keluaran dari setiap *configuration* dapat dicatat secara teratur, ditelusuri kembali, dan dianalisis pada tahap berikutnya. Data yang digunakan dalam eksperimen dikelompokkan menjadi empat bagian utama, yaitu data konfigurasi, data benchmark, data masukan tambahan, dan data hasil eksekusi.

Data konfigurasi disimpan di dalam direktori `config`. Direktori ini memuat berkas *specification* yang menjadi dasar pembentukan ruang eksperimen, termasuk berkas `model.keys.json` yang berisi daftar model yang tersedia melalui layanan API, serta berkas lain yang memuat daftar *persona*, daftar *task*, dan parameter eksekusi. Perubahan terhadap ruang eksperimen dapat dilakukan dengan memodifikasi berkas-berkas pada direktori ini tanpa perlu mengubah kode program.

Data benchmark disimpan di dalam direktori `data`. Direktori ini berisi *dataset* yang digunakan dalam eksperimen, termasuk materi *GSM8K* dan *MMLU-Redux* dalam bentuk mentah maupun bentuk yang telah dinormalisasi untuk keperluan pemrosesan. Dengan pemisahan ini, sumber data utama yang digunakan pipeline terdokumentasi secara jelas.

Direktori `input` digunakan untuk menyimpan data pendukung yang tidak berasal dari benchmark utama tetapi dibutuhkan selama eksperimen, seperti kumpulan soal yang dihasilkan ulang, daftar pertanyaan tambahan, atau berkas uji lain yang disiapkan secara terpisah dari *dataset* utama. Pemisahan antara `data` dan `input` menjaga agar data asli dan data turunan tidak tercampur, serta memudahkan pelacakan asal setiap *task* yang dieksekusi.

Dokumen pendukung, seperti catatan desain, skema eksperimen, dan dokumenta-

si penggunaan pipeline, disimpan pada direktori `docs`. Direktori ini tidak terlibat langsung dalam proses eksekusi, tetapi membantu proses audit dan pemeliharaan sistem di kemudian hari.

Hasil eksperimen disimpan di dalam direktori `results`. Direktori ini memuat berkas *JSON* yang mencatat *response* lengkap untuk setiap *configuration*, termasuk *instruction* yang digunakan, jawaban model, serta metadata yang dihasilkan selama eksekusi. Ringkasan hasil disimpan dalam bentuk *CSV* untuk mempermudah proses analisis, misalnya perbandingan jawaban akhir, tingkat akurasi, jumlah token, atau latensi jika informasi tersebut disediakan oleh layanan model.

Seluruh kode program ditempatkan dalam direktori `src`. Direktori ini berisi modul yang memuat *specification*, menyusun *instruction*, menjalankan *task* untuk setiap *configuration*, serta mencatat hasil eksekusi ke dalam `results`. Dengan pemisahan antara kode dan data, eksperimen dapat dijalankan kembali dengan pengaturan yang sama atau diperluas dengan *specification* baru tanpa mengubah struktur direktori lainnya.

Dengan struktur direktori ini, setiap *response* yang dihasilkan dapat ditelusuri kembali melalui *persona*, model, dan *task* yang digunakan. Perancangan ini mendukung kebutuhan replikasi eksperimen dan menjadi penghubung antara desain konseptual pada bagian sebelumnya dan analisis hasil pada bab berikutnya.

IV.5 Penanganan Gangguan dan Pemulihan *Execution Flow*

Proses eksekusi melibatkan sejumlah besar kombinasi persona, model, dan *task* sehingga rentan terhadap berbagai bentuk gangguan, baik yang bersumber dari layanan model maupun dari kondisi jaringan. Bagian ini menjelaskan mekanisme yang digunakan untuk menjaga agar alur eksekusi tetap berlanjut meskipun terjadi hambatan, serta memastikan bahwa hasil yang diperoleh tetap dapat ditelusuri dan dianalisis tanpa kehilangan konsistensi.

Penanganan gangguan dilakukan dalam dua bentuk utama.

1. *Transient error handling*
   Sistem mendeteksi gangguan sementara seperti *timeout*, penolakan layanan, atau pemutusan koneksi. Apabila gangguan terjadi, *instruction* dikirim ulang menggunakan jeda adaptif. Mekanisme ini mencegah penghentian proses secara keseluruhan dan memastikan setiap *configuration* tetap menghasilkan keluaran yang dapat dianalisis.

2. *Execution flow recovery*

   Untuk menjaga keberlanjutan proses, sistem mencatat status terakhir setelah setiap respons diterima. Apabila eksekusi terhenti sebelum seluruh *configuration* selesai diproses, pipeline dapat dilanjutkan dari posisi terakhir tanpa mengulang bagian yang telah berhasil. Dengan cara ini, proses panjang tetap dapat diselesaikan tanpa kehilangan progres.

Kedua mekanisme ini bekerja bersamaan untuk memastikan bahwa alur eksekusi tetap stabil pada skala besar. Pendekatan ini memungkinkan seluruh rangkaian eksperimen diselesaikan meskipun terdapat hambatan teknis, sehingga hasil yang diperoleh tetap dapat dipertanggungjawabkan dalam tahap analisis pada bab berikutnya.

IV.6 Implementasi Keluaran Pipeline

Bagian ini menyajikan bentuk keluaran yang dihasilkan oleh *evaluation pipeline* setelah seluruh tahapan pemrosesan dijalankan. Keluaran ini berfungsi sebagai artefak utama yang digunakan dalam analisis pada Bab V. Seluruh hasil disimpan dalam direktori `results` dalam format terstruktur sehingga dapat ditelusuri kembali ke *persona*, model, dan *task* yang digunakan.

IV.6.1 Contoh Struktur Log Inferensi

Pipeline mencatat setiap interaksi dengan model dalam bentuk berkas `JSON`. Log ini memuat identitas konfigurasi yang dieksekusi, jawaban model, serta telemetri penggunaan token. Cuplikan berikut memperlihatkan struktur log untuk model yang tidak menyediakan *reasoning trace*.

```
{
  "run": {
    "model_id": "example-model",
    "question_id": "gsm8k_00001",
    "persona": "implicit_male"
  },
  "response": {
    "choices": [
      {
        "message": {
          "content": "Let's break down the problem..."
        }
      }
```

```
    ],
    "usage": {
      "prompt_tokens": 211,
      "completion_tokens": 197,
      "total_tokens": 408
    }
  },
  "meta": {
    "latency_ms": 842,
    "timestamp": "2025-01-18T12:44:10Z"
  }
}
```


Struktur tersebut menunjukkan bahwa pipeline tidak hanya merekam jawaban, tetapi juga metadata komputasional yang diperlukan dalam analisis efisiensi.

IV.6.2 Contoh Struktur Log dengan Reasoning Trace

Beberapa model menyediakan tambahan berupa *reasoning trace*. Bagian penalaran ini disimpan terpisah dari jawaban akhir dan dicatat sebagai bagian dari log. Cuplikan berikut menunjukkan contoh berkas log yang memuat *reasoning trace*.


```
{
  "run": {
    "model_id": "example-model-reason",
    "question_id": "gsm8k_00003",
    "persona": "explicit_genz_female"
  },
  "response": {
    "choices": [
      {
        "message": {
          "content": "Final answer: 70000",
          "reasoning": "First compute the purchase cost..."
        }
      }
    ],
    "usage": {
      "completion_tokens": 867,
```

```
      "reasoning_tokens": 485,
      "total_tokens": 1352
    }
  },
  "meta": {
    "latency_ms": 2134,
    "timestamp": "2025-01-18T12:52:41Z"
  }
}
```


Log ini memungkinkan analisis lebih dalam mengenai gaya penalaran dan perubahan struktur argumen yang mungkin disebabkan oleh persona tertentu.

IV.6.3 Ringkasan Hasil Eksperimen

Pipeline juga menghasilkan ringkasan performa dalam bentuk tabel yang menggabungkan metrik akurasi dan penggunaan token untuk setiap pasangan persona–model. Berkas ini disimpan dalam format `CSV` untuk memudahkan analisis lanjutan. Tabel berikut merupakan contoh ringkasan hasil yang dihasilkan.

Tabel IV.2 Contoh Ringkasan Hasil Eksperimen GSM8K untuk Seluruh Model dan Persona

Model	**Persona**	**Total Q**	**Correct**	**Accuracy (%)**	**Total Tokens**
Bert Nebulon Alpha	man_implicit	610	593	97.21	285250
Bert Nebulon Alpha	woman_implicit	641	627	97.26	335208
Grok 4.1 Fast	man_implicit	1315	1242	94.45	1325229
Grok 4.1 Fast	woman_implicit	1316	1254	95.36	1422736
Nvidia Nemotron 12B v2 VL	man_implicit	1305	1224	93.79	1156049
Nvidia Nemotron 12B v2 VL	woman_implicit	1306	1230	94.18	1184521

BAB V

RENCANA SELANJUTNYA

V.1 Rencana Implementasi dan Estimasi Biaya

Rencana implementasi pada tahap berikutnya adalah menjalankan kembali *evaluation pipeline* yang telah dijelaskan pada Bab IV dengan cakupan penuh, yang meliputi sembilan model bahasa, dua *benchmark* penalaran (GSM8K dan MMLU-Redux), serta lima belas *user persona* (implisit, eksplisit, dan netral). Bagian ini merumuskan langkah implementasi teknis, asumsi kebutuhan token, serta estimasi biaya penggunaan API berdasarkan harga resmi masing-masing model pada platform OpenRouter

Estimasi dilakukan menggunakan kurs konstan 1 USD = Rp16.000.

V.1.1 Rencana Implementasi Eksperimen

Pelaksanaan eksperimen direncanakan mengikuti enam langkah utama berikut.

1. Persiapan aset data.
   Sistem memuat berkas definisi lima belas persona, korpus GSM8K (split *test*), MMLU-Redux (20 subjek), kredensial API, serta konfigurasi model. Struktur direktori dan modul pemrosesan mengikuti rancangan pada Subbab IV.4.
2. Inisialisasi dan *warm-up* persona.
   Setiap model menerima satu pesan awal untuk menanamkan konteks persona sebelum mengerjakan soal pertama. Tahap ini juga berfungsi sebagai *sanity check* untuk memastikan bahwa model mengikuti identitas dan gaya bahasa persona secara konsisten.
3. Eksekusi eksperimen utama.
   Setiap kombinasi model–persona menjalankan seluruh soal GSM8K dan MMLU-Redux menggunakan mekanisme injeksi pesan berbasis peran: persona pada *system message* dan soal pada *user message*. Setiap respons diharuskan me-

nyertakan penalaran langkah demi langkah.

4. Pencatatan log granular.
   Seluruh respons disimpan sebagai berkas JSON yang memuat isi *prompt*, jawaban mentah, *token usage*, serta *latency*. Format ini memastikan bahwa setiap respons dapat ditelusuri kembali ke konfigurasi yang digunakan.
5. Agregasi dan validasi hasil.
   Log yang terkumpul diubah menjadi berkas CSV agregat yang berisi akurasi, rata-rata latensi, serta total konsumsi token. Validasi tambahan dilakukan melalui pemeriksaan pola jawaban dan konsistensi jumlah entri.
6. Penanganan kegagalan.
   Kegagalan akibat *timeout* atau batas *rate limit* ditangani menggunakan mekanisme *retry* dengan *exponential backoff*, sebagaimana dijelaskan pada Bab IV. Dengan demikian, kegagalan sebagian tidak menghentikan keseluruhan eksperimen.

V.1.2 Himpunan Model dan Skenario Eksekusi

Eksperimen ini menggunakan sembilan model dengan rincian sebagai berikut.

1. Enam model berbayar (via OpenRouter):
   (a) openai/gpt-5-mini
   (b) anthropic/claude-haiku-4.5
   (c) google/gemini-2.5-flash
   (d) deepseek/deepseek-v3.2
   (e) nvidia/llama-3.3-nemotron-super-49b-v1.5
   (f) google/gemma-3n-e4b-it
2. Tiga model yang pada saat perancangan tersedia sebagai *free-tier*:
   (a) xai/grok-4.1-fast
   (b) nvidia/nemotron-nano-12b-v2-vl
   (c) openrouter/bert-nebulon-alpha

Seluruh sembilan model dijalankan pada konfigurasi penuh: dua *benchmark* dan lima belas persona. Namun, estimasi biaya hanya dihitung untuk enam model berbayar.

V.1.3 Asumsi Jumlah Soal dan Kebutuhan Token

Kebutuhan token dihitung berdasarkan dua sumber utama: GSM8K (1319 soal) dan MMLU-Redux (2000 soal). Pada kedua *benchmark*, model diarahkan untuk memberikan penalaran lengkap sebelum jawaban akhir, sehingga konsumsi token per

soal diharapkan berada pada kisaran yang relatif tinggi.

1. GSM8K.
   Total token per persona per model diestimasikan sebagai:

$$T_{\text{GSM8K}} \approx 1319 \times 1200 = 1{,}582{,}800 \text{ token}.$$

2. MMLU-Redux.
   Total token per persona per model diestimasikan sebagai:

$$T_{\text{MMLU}} \approx 2000 \times 1200 = 2{,}400{,}000 \text{ token}.$$

Total token inti per persona diperoleh dari penjumlahan keduanya:

$$T_{\text{base, persona}} = 1{,}582{,}800 + 2{,}400{,}000 = 3{,}982{,}800.$$

Untuk mengakomodasi *warm-up* dan *retry*, digunakan faktor overhead 20%:

$$T_{\text{persona}} \approx 1.2 \times 3{,}982{,}800 = 4{,}779{,}360.$$

Sehingga total token per model untuk 15 persona adalah:

$$T_{\text{model}} \approx 15 \times 4{,}779{,}360 = 71{,}690{,}400 \approx 71{,}7 \times 10^6.$$

Komposisi token diasumsikan:

$$T_{\text{in}} = 0.4T_{\text{model}}, \qquad T_{\text{out}} = 0.6T_{\text{model}}.$$

V.1.4 Estimasi Biaya per Model

Harga token per model mengacu pada dokumentasi OpenRouter(*OpenAI GPT-5 Mini Pricing*; *Anthropic Claude Haiku 4.5 Pricing*; *Google Gemini 2.5 Flash Pricing*; *DeepSeek V3.2 Pricing*; *NVIDIA Llama 3.3 Nemotron Super 49B V1.5 Pricing*; *Google Gemma 3n 4B Pricing*). Biaya untuk model ke-m dihitung dengan rumus:

$$\text{cost}_m = p_{\text{in},m} \times \frac{T_{\text{in}}}{10^6} + p_{\text{out},m} \times \frac{T_{\text{out}}}{10^6},$$

dengan $p_{\text{in},m}$ dan $p_{\text{out},m}$ adalah harga per satu juta token untuk *input* dan *output*.

Estimasi berikut menggunakan kurs Rp 16.000 per USD dan total token $T_{\text{model}} \approx 71{,}7 \times 10^6$.

Tabel V.1 Estimasi biaya enam model berbayar untuk konfigurasi penuh 15 persona

Model	**Total Token T_{model}**	**Biaya (USD)**	**Biaya (Rp)**
openai/gpt-5-mini	$\approx 71{,}7 \times 10^6$	93.20	$\approx 1{,}491{,}000$
anthropic/claude-haiku-4.5	$\approx 71{,}7 \times 10^6$	243.75	$\approx 3{,}900{,}000$
google/gemini-2.5-flash	$\approx 71{,}7 \times 10^6$	116.14	$\approx 1{,}858{,}000$
deepseek/deepseek-v3.2	$\approx 71{,}7 \times 10^6$	24.95	$\approx 399{,}000$
nvidia/llama-3.3-nemotron-super-49b-v1.5	$\approx 71{,}7 \times 10^6$	20.07	$\approx 321{,}000$
google/gemma-3n-e4b-it	$\approx 71{,}7 \times 10^6$	2.29	$\approx 37{,}000$
Total enam model berbayar	–	**500.40**	$\approx$ **8,006,000**

Tiga model lain yang tersedia sebagai *free-tier* (grok-4.1-fast, nemotron-nano-12b-v2-vl, dan bert-nebulon-alpha) diperkirakan mengonsumsi token serupa tetapi tidak menimbulkan biaya finansial langsung. Status *free-tier* tersebut tetap harus diverifikasi kembali sebelum eksperimen akhir dijalankan.

Dengan demikian, estimasi total biaya finansial untuk menjalankan seluruh eksperimen multi-model, multi-persona, dan dua *benchmark* penalaran adalah sekitar 500,40 USD atau kurang lebih 8 juta rupiah. Angka ini bersifat konservatif karena telah memasukkan biaya *warm-up* dan *retry*, sehingga realisasi biaya dapat lebih rendah apabila konsumsi token aktual per soal ternyata lebih kecil dari asumsi yang digunakan dalam perhitungan ini.

V.2 Desain Pengujian dan Evaluasi

Desain pengujian pada tahap berikut disusun untuk memastikan bahwa seluruh hasil eksperimen dapat diverifikasi, divalidasi, dan direplikasi. Struktur pengujian memanfaatkan artefak log granular, telemetry penggunaan token, serta pemeriksaan konsistensi yang telah ditanamkan dalam pipeline pada Bab IV.

1. Verifikasi konsistensi eksekusi.
   Verifikasi dilakukan untuk memastikan bahwa setiap model menerima stimulus yang identik pada setiap soal dan persona, sehingga variasi respons dapat dikaitkan langsung dengan perbedaan persona atau arsitektur model.
   (i) Konsistensi konstruksi prompt.
   Pemeriksaan dilakukan untuk memastikan bahwa struktur persona pada *system message* dan isi soal pada *user message* identik pada seluruh eksekusi.

Setiap variasi kecil seperti pergeseran tanda baca atau perubahan format dapat mengubah jalur penalaran model, sehingga pemeriksaan dilakukan secara programatik pada log JSON.

(ii) Kesesuaian urutan eksekusi.

Pemeriksaan dilakukan dengan mencocokkan indeks interaksi, nomor soal, dan urutan persona pada seluruh berkas log untuk memastikan bahwa sistem menjalankan eksperimen sesuai konfigurasi yang direncanakan.

(iii) Keberhasilan tahap warm-up.

Tahap warm-up diverifikasi dengan menilai apakah respons awal model mengikuti identitas dan gaya bahasa persona. Kegagalan tahap ini dicatat sebagai anomali dan disertai eksekusi ulang sebelum proses utama dimulai.

2. Validasi keluaran model.

   Validasi keluaran bertujuan memastikan bahwa jawaban model berada dalam format yang sesuai untuk dievaluasi. Pendekatan validasi dibedakan untuk GSM8K dan MMLU-Redux.

   (i) Validasi GSM8K.

   Model harus memberikan jawaban numerik akhir yang dapat diekstraksi secara deterministik. Selain itu, respons harus mencakup penalaran langkah demi langkah sebelum menyatakan jawaban akhir.

   (ii) Validasi MMLU-Redux.

   Model harus memberikan pilihan jawaban dalam format A, B, C, atau D. Meskipun merupakan soal pilihan ganda, model tetap diminta menjelaskan penalaran sebelum memilih opsi, sehingga respons memiliki struktur yang konsisten.

   (iii) Pemeriksaan konsistensi format respons.

   Pemeriksaan mencakup panjang respons, struktur teks, keberadaan penalaran, serta keterbacaan sehingga setiap respons dapat diproses ulang tanpa kesalahan parsing.

3. Evaluasi kuantitatif.

   Evaluasi kuantitatif dilakukan untuk mengukur dampak persona terhadap performa model pada dua benchmark.

   (i) Akurasi jawaban.

   Akurasi dihitung dengan membandingkan jawaban akhir yang diekstraksi terhadap ground truth. Penghitungan dilakukan pada tabel agregasi hasil.

   (ii) Konsumsi token.

   Evaluasi melibatkan token input, token output, dan token penalaran sebagai indikator beban komputasi dan kecenderungan verbosity model di bawah per-

sona tertentu.

(iii) Latensi eksekusi.

Latensi diambil dari metadata waktu pada log JSON untuk menilai stabilitas waktu respons model ketika menangani beban besar dan variasi persona.

V.3 Analisis Risiko dan Mitigasi

Pelaksanaan eksperimen pada lingkungan multi-model dan multi-persona menimbulkan sejumlah risiko metodologis dan operasional yang perlu dikelola secara sistematis agar integritas penelitian tetap terjaga. Risiko-risiko tersebut mencakup aspek reliabilitas penggunaan API, kestabilan keluaran model, konsistensi proses penalaran, menjaga ketepatan pesan sepanjang percakapan, serta akurasi proses agregasi data. Selain itu, penelitian sebelumnya menunjukkan bahwa konsistensi LLM dapat menurun pada evaluasi berskala besar dan bahwa penalaran model dapat terpengaruh oleh faktor-faktor non-linguistik yang tidak terkontrol. Berdasarkan temuan tersebut, bagian ini menguraikan tiga kategori risiko utama serta strategi mitigasinya.

1. Risiko kegagalan pemanggilan API.

   Risiko ini mencakup galat seperti *timeout*, gangguan koneksi, dan pembatasan layanan (rate limit). Kegagalan ini berpotensi menyebabkan hilangnya sebagian data atau ketidaksinkronan indeks percobaan.

   (i) Mitigasi dilakukan melalui mekanisme *retry* adaptif berbasis *exponential backoff*, sesuai praktik standar pada sistem terdistribusi.

   (ii) Seluruh kegagalan direkam dalam log terpisah untuk memastikan keterlacakan sehingga perbaikan atau pengulangan dapat dilakukan secara selektif.

   (iii) Tingkat konkurensi dijalankan secara otomatis ketika sistem mendeteksi peningkatan laju galat, guna menjaga stabilitas kapasitas layanan.

2. Risiko lonjakan konsumsi token.

   LLM sering menghasilkan keluaran yang lebih panjang daripada yang diinstruksikan, terutama ketika diminta memberikan penalaran langkah demi langkah. Fenomena ini berdampak langsung pada biaya dan durasi eksperimen.

   (i) Sistem membatasi panjang keluaran dengan parameter *maximum completion length* untuk mencegah respons berlebihan.

   (ii) Validasi awal dijalankan secara berkala untuk memantau rata-rata konsumsi token per soal.

   (iii) Persona yang terbukti memicu keluaran terlalu panjang dilakukan penyesuaian instruksi secara minimal untuk mengendalikan panjang teks tanpa

mengubah maksud identitas sosial.

3. Risiko penyimpanan dan konsistensi log.
   Volume log yang besar berpotensi menimbulkan risiko korupsi berkas dan ketidakcocokan antara indeks model, persona, dan soal.
   (i) Setiap respons disimpan dalam format terstruktur (JSON) dengan skema tetap.
   (ii) Proses agregasi mengadopsi pemeriksaan konsistensi silang antara jumlah entri dan indeks soal.
   (iii) Mekanisme *checkpointing* diterapkan untuk menghindari kehilangan data apabila eksekusi terhenti di tengah proses.

DAFTAR PUSTAKA

Anthropic Claude Haiku 4.5 Pricing. https://openrouter.ai/anthropic/claude-haiku-4.5. Diakses 2025.

Bommasani, Rishi, Drew A. Hudson, Ehsan Adeli, dkk. 2021. "On the Opportunities and Risks of Foundation Models". *arXiv preprint arXiv:2108.07258,* https://arxiv.org/abs/2108.07258.

Cobbe, Karl, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, dkk. 2021. "Training Verifiers to Solve Math Word Problems". *arXiv preprint arXiv:2110.14168,* https://arxiv.org/abs/2110.14168.

DeepSeek V3.2 Pricing. https://openrouter.ai/deepseek/deepseek-v3.2. Diakses 2025.

Edinburgh Dataset Analytics Working Group. 2024. *MMLU-Redux 2.0 Dataset.* https://huggingface.co/datasets/edinburgh-dawg/mmlu-redux-2.0. Versi kurasi ulang MMLU dengan 57 subjek dan 100 butir soal per subjek.

Gema, Aryo Pradipta, Joshua Ong Jun Leang, Giwon Hong, Alessio Devoto, dkk. 2024. "Are We Done with MMLU?" *arXiv preprint arXiv:2406.04127,* https://arxiv.org/abs/2406.04127.

Google Gemini 2.5 Flash Pricing. https://openrouter.ai/google/gemini-2.5-flash. Diakses 2025.

Google Gemma 3n 4B Pricing. https://openrouter.ai/google/gemma-3n-e4b-it. Diakses 2025.

Gupta, Shashank, Vaishnavi Shrivastava, Ameet Deshpande, Ashwin Kalyan, Peter Clark, Ashish Sabharwal, dan Tushar Khot. 2024. "Bias Runs Deep: Implicit Reasoning Biases in Persona-Assigned Language Models". Dalam *Proceedings of the Twelfth International Conference on Learning Representations.* https://openreview.net/forum?id=kGteeZ18Ir.

Hendrycks, Dan, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, dan Jacob Steinhardt. 2021. "Measuring Massive Multitask Language Understanding". *International Conference on Learning Representations,* https://arxiv.org/abs/2009.03300.

Liang, P., R. Bommasani, dkk. 2023. "Holistic Evaluation of Language Models". *arXiv preprint arXiv:2211.09110,* https://arxiv.org/abs/2211.09110.

Naous, Tarek, Baptiste Roziere, dkk. 2025. "Training and Evaluating User Language Models". *arXiv preprint arXiv:2510.06552,* https://arxiv.org/abs/2510.06552.

NVIDIA Llama 3.3 Nemotron Super 49B V1.5 Pricing. https://openrouter.ai/nvidia/llama-3.3-nemotron-super-49b-v1.5. Diakses 2025.

OpenAI GPT-5 Mini Pricing. https://openrouter.ai/openai/gpt-5-mini. Diakses 2025.

Sap, Maarten, Hannah Rashkin, Derek Chen, Ronan Le Bras, dan Yejin Choi. 2019. "SocialIQA: Commonsense Reasoning about Social Interactions". Dalam *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing.* Hong Kong, China. https://arxiv.org/abs/1904.09728.

Tseng, Yu-Min, Yu-Chao Huang, Teng-Yun Hsiao, Wei-Lin Chen, Chao-Wei Huang, Yu Meng, dan Yun-Nung Chen. 2024. "Two Tales of Persona in LLMs: A Survey of Role-Playing and Personalization". Dalam *Findings of the Association for Computational Linguistics: EMNLP 2024,* 16612–16631. https://aclanthology.org/2024.findings-emnlp.969.

Turpin, Miles, dkk. 2023. "Language Models Don't Always Say What They Think: Unfaithful Explanations in Chain-of-Thought Reasoning". *arXiv preprint arXiv:2305.04388,* https://arxiv.org/abs/2305.04388.

Weidinger, Laura, John Mellor, Maribeth Rauh, Christopher Griffin, Iason Gabriel, Jonathan Uesato, Po-Sen Huang, Zachary Kenton, Tom B. Brown, dkk. 2021. "Ethical and Social Risks of Harm from Language Models". *arXiv preprint arXiv:2112.04359,* https://arxiv.org/abs/2112.04359.

Zhao, Yanhao, Eric Wallace, Shi Feng, Mohit Singh, dan Matt Gardner. 2021. "Calibrate Before Use: Improving Few-Shot Performance of Language Models". Dalam *Proceedings of the International Conference on Machine Learning,* 12697–12706.

Zhou, Luozhi, dkk. 2023. "Large Language Models Are Sensitive to Prompt Framing". *arXiv preprint arXiv:2310.05400,* https://arxiv.org/abs/2310.05400.