

FRAMEWORK KARANGTURI UNTUK PROMPT ENGINEERING: ANALISIS KOMPARATIF DENGAN RTF, COT, DAN REACT PADA MODEL AI GENERATIF

Hermawan Honggo Widagdo^{1✉}, Siska Narulita², Thomas William³

¹ Program Studi Bisnis Digital, Binus Business School, Universitas Bina Nusantara, POJ Avenue Kav. 3C POJ City, Semarang, Jawa Tengah, 50144, Indonesia

² Teknologi Pangan, Fakultas Sains dan Teknologi, Universitas Nasional Karangturi, Jl.Raden Patah No. 182-192, Semarang, Jawa Tengah, 50127, Indonesia

³ Information Management and System, Information Technology, Monash University, Wellington Rd, Clayton VIC 3800, Australia

Info Artikel

Sejarah Artikel:

Disubmit : 11-08-2025

Direvisi : 19-08-2025

Disetujui : 23-08-2025

Keywords:

Prompt Engineering ;
KARANGTURI Framework ;
Large Language Models
(LLM) ; AI Prompting
Strategies ; Comparative
Analysis

Abstrak

Penelitian ini bertujuan untuk menganalisis secara komparatif empat kerangka kerja dalam *prompt engineering*, yaitu KARANGTURI, RTF (Role-Task-Format), COT (Chain-of-Thought), dan ReAct. Keempat kerangka kerja tersebut memiliki peran penting dalam membantu pengguna merancang instruksi yang efektif kepada model bahasa besar (*Large Language Models/LLM*). Studi ini menggunakan pendekatan deskriptif-komparatif dengan mengkaji masing-masing kerangka kerja dari segi struktur, fokus, kompleksitas, kelebihan, kelemahan, serta konteks penggunaannya. KARANGTURI merupakan kerangka kerja lokal yang terdiri dari empat elemen utama: Karakter, Rangkuman, Tujuan, dan Rintangan. RTF menawarkan pendekatan sederhana berbasis tiga komponen utama yang cocok untuk tugas-tugas langsung. CoT berfokus pada penalaran bertahap dan cocok untuk permasalahan logis dan kompleks. ReAct mengintegrasikan penalaran dengan tindakan, serta mendukung interaksi dengan alat eksternal untuk menyelesaikan tugas-tugas lanjutan. Hasil analisis menunjukkan bahwa pemilihan kerangka kerja tergantung pada jenis tugas, tingkat kompleksitas, serta kebutuhan akan reasoning atau akses informasi eksternal. KARANGTURI dipandang sebagai pendekatan menyeluruh yang fleksibel dan berpotensi tinggi, namun masih memerlukan validasi lebih lanjut melalui eksperimen dan aplikasi nyata. Temuan ini diharapkan dapat membantu praktisi AI dalam memilih strategi prompt yang tepat sesuai kebutuhan.

Abstract

This study aims to perform a comparative analysis of four prompt engineering frameworks: KARANGTURI, RTF (Role-Task-Format), CoT (Chain-of-Thought), and ReAct. These frameworks play a crucial role in assisting users in designing effective instructions for Large Language Models (LLMs). A descriptive-comparative approach is employed to examine each framework in terms of structure, focus, complexity, strengths, limitations, and practical application. KARANGTURI, a locally developed framework, consists of four key elements: Character, Summary, Goal, and Constraint. RTF offers a simple structure based on three core components, making it suitable for straightforward tasks. CoT emphasizes step-by-step reasoning and is effective for complex and logical challenges. ReAct integrates reasoning with actions and supports interaction with external tools for advanced tasks. The analysis reveals that the choice of framework depends on task type, complexity level, and the need for reasoning or access to external information. KARANGTURI is viewed as a comprehensive and flexible approach with promising potential, though it requires further empirical validation. The findings are expected to help AI practitioners select the most appropriate prompting strategy based on their specific needs.

[✉] Alamat Korespondensi:

E-mail: hermawan.widagdo@binus.ac.id

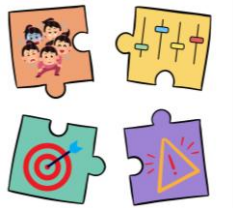
1. Introduction

Perkembangan pesat teknologi kecerdasan buatan, khususnya *Large Language Models* (LLMs), telah mendorong munculnya kebutuhan akan teknik interaksi yang efektif antara manusia dan mesin. Kemudahan akses melalui antarmuka web menjadi salah satu faktor utama yang mendorong pesatnya adopsi AI di berbagai bidang, karena pengguna tidak perlu menginstal perangkat lunak tambahan dan dapat langsung berinteraksi melalui peramban (Hermawan, 2021).

Salah satu pendekatan yang menjadi kunci dalam optimalisasi penggunaan *LLM* adalah *prompt engineering*, yakni seni dan teknik dalam menyusun instruksi atau masukan kepada model agar menghasilkan keluaran yang relevan, akurat, dan sesuai tujuan pengguna (Brown et al., 2020).

Dalam praktiknya, *prompt engineering* tidak lagi bersifat coba-coba, melainkan telah dikembangkan menjadi pendekatan yang sistematis melalui berbagai kerangka kerja. Berbagai *framework* telah muncul untuk membantu pengguna merancang *prompt* yang lebih terstruktur, mulai dari *RTF* (*Role, Task, Format*) yang sederhana, *Chain-of-Thought* (*CoT*) yang menekankan penalaran langkah demi langkah, hingga *ReAct* yang memungkinkan interaksi model dengan alat eksternal (Wei et al., 2022; Yao et al., 2022).

Di tengah dominasi pendekatan luar negeri tersebut, muncul kerangka kerja lokal bernama **KARANGTURI**, yang mencoba menawarkan pendekatan holistik dalam menyusun *prompt* melalui empat komponen: Karakter, Rangkuman, Tujuan, dan Rintangan.



Pada era saat ini, *prompt engineering* dapat dikatakan sebagai sebuah *hard skill* yang harus dikuasai dan tidak berhenti pada tahap *professional knowledge* saja (Lukito, 2023). Namun demikian, belum banyak kajian ilmiah yang secara sistematis membandingkan keempat kerangka kerja ini. Sebagian besar pengguna *LLM* masih mengalami kesulitan dalam memilih *framework* yang paling tepat untuk konteks penggunaan tertentu, seperti pembuatan konten, analisis data, hingga pemecahan masalah logis. Oleh karena itu, penelitian ini berupaya mengisi celah tersebut dengan menyajikan analisis komparatif terhadap keempat *framework*, dengan fokus pada struktur, fokus kerja, kompleksitas, kelebihan, serta kasus penggunaan yang relevan. Hasil kajian ini diharapkan dapat menjadi rujukan awal bagi para praktisi, akademisi, maupun pengembang dalam menentukan strategi *prompt engineering* yang sesuai.

Penelitian ini dilakukan dengan pendekatan studi literatur dan analisis komparatif terhadap dokumentasi publik, artikel ilmiah, panduan teknis, dan repositori daring yang membahas berbagai kerangka kerja *prompt engineering* (Vatsal, 2024). Alat bantu utama yang digunakan dalam penelitian ini adalah komputer dengan akses internet, perangkat lunak pengolah kata MS Word, serta aplikasi manajemen referensi seperti Zotero. Data sekunder diperoleh dari situs resmi penyedia informasi seperti arXiv, GitHub (Liu, 2021), *Prompt Engineering Guide* (dair.ai, 2023), dan artikel pendukung lainnya.

2.2. Metode Penelitian

Jenis penelitian yang digunakan adalah **deskriptif-komparatif**, yang bertujuan untuk membandingkan empat kerangka kerja *prompt engineering*: *KARANGTURI*, *RTF*, *Chain-of-Thought* (*CoT*), dan *ReAct* (Gu, et al., 2023). Penelitian ini tidak melibatkan eksperimen langsung dengan pengguna, namun fokus pada pengumpulan data literatur dan dokumentasi untuk mengidentifikasi dan membandingkan karakteristik masing-masing *framework*.

Faktor-faktor yang diteliti dalam penelitian ini mencakup berbagai aspek utama dari masing-masing kerangka kerja *prompt engineering*. Pertama, struktur kerangka kerja dianalisis untuk memahami bagaimana komponen-komponen utama disusun dan berinteraksi dalam membentuk *prompt* yang efektif. Kedua, fokus dan tujuan dari tiap kerangka kerja diidentifikasi guna menilai arah dan sasaran yang ingin dicapai oleh pengguna saat menggunakannya. Ketiga, kompleksitas penggunaan menjadi perhatian penting, khususnya dalam menilai sejauh mana kemudahan atau kesulitan implementasi *framework* oleh pengguna dengan tingkat keahlian yang berbeda. Keempat, kekuatan dan kelemahan masing-masing pendekatan dievaluasi secara kritis berdasarkan kajian literatur dan potensi penerapannya dalam konteks praktis. Terakhir, penelitian ini juga menggali kasus-kasus penggunaan umum (*typical use cases*) yang sesuai untuk masing-masing kerangka kerja agar dapat memberikan gambaran konkret mengenai kecocokan strategi *prompting* terhadap kebutuhan pengguna yang beragam.

Respons yang diharapkan adalah pemahaman komprehensif mengenai keunggulan relatif dan kelemahan tiap *framework* dalam konteks penerapan *prompt engineering* untuk *LLM*.

2. Methods

2.1. Alat dan Bahan

Tahapan penelitian dilakukan sebagai berikut:

1. Identifikasi dan pemilihan kerangka kerja yang relevan
2. Pengumpulan sumber literatur dan dokumentasi
3. Dekonstruksi dan analisis masing-masing framework
4. Penyusunan matriks perbandingan
5. Penyimpulan hasil analisis

Metode penarikan kesimpulan menggunakan pendekatan logika deduktif, di mana data dari sumber literatur dikaji dan dianalisis untuk membentuk generalisasi terhadap efektivitas dan kecocokan masing-masing framework.

2.3. Analisis Data

Data yang telah dikumpulkan dianalisis menggunakan metode **kualitatif-deskriptif**, dengan teknik analisis isi (content analysis). Informasi dari berbagai sumber disintesis untuk menghasilkan pemetaan komparatif berdasarkan lima dimensi utama: struktur, fokus, kompleksitas, kekuatan, dan kelemahan. Data disajikan dalam bentuk tabel perbandingan untuk memudahkan visualisasi dan penarikan kesimpulan. Validitas data dijaga dengan mengacu pada sumber terpercaya dan artikel yang telah ditinjau sejawat atau memiliki kredibilitas tinggi di bidang teknologi AI dan prompt engineering.

3. Results and Discussion

3.1. Hasil 1 dan Pembahasan

Hasil pertama menunjukkan bahwa setiap kerangka kerja prompt engineering memiliki struktur komponen yang khas dan tidak seragam. KARANGTURI, sebagai kerangka lokal, memiliki empat komponen utama yang lebih eksplisit dan menyeluruh dibandingkan dengan RTF yang hanya mencakup tiga aspek. Di sisi lain, CoT dan ReAct tidak mengandalkan struktur statis, melainkan menekankan pada pendekatan penalaran dan aksi yang bersifat dinamis dan iteratif. Dari segi kelengkapan elemen, KARANGTURI dan ReAct menawarkan pendekatan yang lebih holistik, sedangkan RTF cenderung praktis dan sederhana. Adapun CoT lebih fokus pada peningkatan logika berpikir bertahap, tanpa format tetap.

Tabel berikut menyajikan perbandingan struktur antar kerangka kerja:

Tabel 1. Perbandingan Struktur Komponen Framework Prompt Engineering

Framework	Komponen Utama	Jumlah Komponen	Keterangan Tambahan
KARANGTURI	Karakter, Rangkuman, Tujuan, Rintangan	4	Kontekstual dan fleksibel
RTF	Role, Task, Format	3	Sederhana, cocok untuk pemula
CoT	Tidak eksplisit, berbasis reasoning langkah demi langkah	-	Cocok untuk tugas analitis
ReAct	Reasoning, Action, Observation	3 (dalam siklus)	Membutuhkan akses alat eksternal

3.2. Hasil 2 dan Pembahasan

Hasil kedua mengungkapkan perbedaan fokus, kompleksitas, kekuatan, dan kelemahan masing-masing framework. RTF sangat cocok untuk kebutuhan cepat, ringkas, dan berformat jelas—terutama pada pengambilan data atau pembuatan konten instan. Sebaliknya, CoT dan ReAct membutuhkan model LLM dengan kemampuan *reasoning* tinggi dan dapat menyebabkan peningkatan latensi. ReAct memiliki keunggulan dalam menghadapi permasalahan yang memerlukan informasi eksternal, tetapi memerlukan sistem tambahan. KARANGTURI dipandang menjembatani berbagai pendekatan ini karena mencakup persona, tujuan, konteks, dan batasan secara terstruktur.

Tabel 2. Analisis Perbandingan Fokus dan Kompleksitas Framework

Framework	Fokus Utama	Kompleksitas	Kekuatan Utama	Kelemahan
KARANGTURI	Konteks + tujuan + persona	Sedang	Holistik dan fleksibel	Belum tervalidasi secara akademik luas
RTF	Prompt langsung dan jelas	Rendah	Mudah digunakan dan cepat	Kurang cocok untuk tugas kompleks

CoT	Penalaran bertahap	Sedang-Tinggi	Meningkatkan akurasi pada reasoning	Respons lebih panjang dan lambat
ReAct	Penalaran dan aksi eksternal	Tinggi	Adaptif untuk informasi terkini	Butuh tool tambahan, tidak selalu efisien

Selain perbandingan struktur dan kompleksitas, penelitian ini juga mengkaji deskripsi penggunaan atau *typical use cases* dari masing-masing framework. Framework **KARANGTURI** sangat cocok diterapkan dalam skenario di mana pengguna ingin menyusun prompt yang kontekstual dan personal, seperti pada pembuatan konten berbasis persona, chatbot edukasi, serta simulasi dialog profesional. Struktur empat bagiannya memberikan fleksibilitas tinggi untuk kebutuhan praktis yang melibatkan audiens spesifik.

Framework **RTF** sangat ideal untuk tugas-tugas cepat dan langsung seperti ekstraksi data, penyusunan daftar, atau permintaan ringkas yang memerlukan format tertentu. Pendekatan ini cocok untuk pengguna pemula karena hanya membutuhkan tiga komponen utama dan tidak memerlukan pemikiran kompleks.

Sementara itu, **CoT** sangat efektif digunakan untuk pemecahan masalah kompleks seperti perhitungan logika, analisis multi-tahap, atau tugas akademik yang memerlukan penalaran sistematis. Prompt dengan pendekatan ini mendorong AI untuk “berpikir secara bertahap” sebelum menyampaikan kesimpulan.

Adapun **ReAct** lebih cocok digunakan dalam skenario interaktif yang membutuhkan akses terhadap informasi eksternal, seperti pencarian data terkini, pelacakan sumber, atau pengambilan keputusan berbasis informasi dinamis. Kerangka ini sangat relevan dalam lingkungan kerja berbasis alat bantu eksternal seperti browser atau plugin AI assistant.

Dengan demikian, pemilihan framework perlu mempertimbangkan konteks penggunaan secara spesifik, bukan hanya berdasarkan kemudahan atau kompleksitas strukturnya, melainkan juga pada *fit* antara kemampuan framework dan kebutuhan pengguna.

Penggunaan Prompt Engineering dengan Framework tertentu pada Berbagai Model AI

Berdasarkan hasil eksplorasi praktis dan dokumentasi teknis, setiap *framework prompt engineering* memiliki tingkat kompatibilitas dan efektivitas yang bervariasi tergantung pada platform LLM yang digunakan. Berikut adalah rangkuman pemanfaatannya:

1. Framework KARANGTURI

Framework ini sangat efektif diterapkan pada

platform **ChatGPT (OpenAI)** dan **Claude (Anthropic)**, karena keduanya mendukung konteks panjang dan mampu memahami persona, tujuan, serta batasan dengan baik (**OpenAI,2023 , Anthropic,2023**). Pengguna dapat menyusun prompt dengan format eksplisit seperti:

Karakter: “Act as a senior HR consultant...”

Rangkuman: “A company is experiencing high turnover among Gen Z employees...”

Tujuan: “Suggest 3 practical strategies to retain young talent.”

Rintangan: “Avoid academic jargon. Response must be under 150 words.”

KARANGTURI juga dapat digunakan di **Gemini (Google)**, meskipun model ini terkadang mempersingkat atau menyederhanakan interpretasi terhadap persona (**Google Cloud,2024**). Di **LLaMA (Meta)**, karena sebagian besar digunakan melalui antarmuka open source seperti llama.cpp atau Ollama, pengguna perlu menyesuaikan dengan keterbatasan UI dan memecah prompt panjang ke beberapa bagian.

2. Framework RTF

RTF sangat cocok untuk semua model, termasuk **Gemini, ChatGPT, Claude, dan Mistral**, karena pendekatannya ringkas dan sistematis. Prompt RTF mudah diterjemahkan oleh model karena strukturnya yang jelas:

Role: “You are a travel advisor.”

Task: “List 5 destinations in Indonesia for solo female travelers.”

Format: “Output in bullet points with a one-line description per place.”

Framework ini sangat andal untuk kebutuhan bisnis, *customer service*, hingga data entry otomatis.

3. Framework CoT (Chain-of-Thought)

CoT sangat cocok untuk platform dengan kemampuan penalaran tinggi seperti **ChatGPT-4, Claude 3, dan Gemini 1.5**. Prompt seperti:

“Let’s think step by step to solve this logic puzzle...”

akan memicu model untuk menghasilkan output bertahap yang disusun secara logis. Namun di **LLaMA** atau **Gemini versi ringan**, kemampuan ini bisa terbatas tanpa konteks atau instruksi eksplisit.

4. Framework ReAct

ReAct memerlukan model dan platform yang mendukung aksi dan reasoning terintegrasi. Platform seperti **ChatGPT dengan plugin atau tools, LangChain (dengan LLaMA atau Mistral)**, dan

Gemini dengan akses web memungkinkan pendekatan ini. Prompt ReAct biasanya disusun seperti ini:

Think: "I need to know the current weather in Tokyo."
Act: "Search for 'Tokyo weather today.'"
Observe: "Today's weather is 24°C and cloudy."
Respond: "You should bring a light jacket."

Namun, tidak semua model mendukung eksekusi tindakan secara native tanpa integrasi eksternal (plugin/API).

Fokus Khusus: Potensi dan Keunggulan Framework KARANGTURI

Framework **KARANGTURI** menunjukkan pendekatan yang holistik dan adaptif dalam merancang prompt, yang menggabungkan empat aspek utama yang saling melengkapi: persona, konteks, tujuan, dan batasan. Pendekatan ini tidak hanya mendorong keluaran AI yang lebih terarah dan sesuai konteks, namun juga memberikan fleksibilitas tinggi bagi pengguna dari berbagai latar belakang, mulai dari akademisi, pebisnis, pendidik, hingga kreator konten.

Keunggulan utama **KARANGTURI** terletak pada kemampuannya dalam **menyusun prompt yang bersifat strategis dan audience-oriented**, menjadikannya sangat cocok untuk skenario seperti:

- Pelatihan chatbot dengan persona edukatif,
- Penyusunan konten profesional dengan batasan formalitas,
- Presentasi berbasis naratif (storytelling) dengan tujuan spesifik,
- Simulasi percakapan bisnis atau wawancara kerja,
- Pembuatan sistem AI tutoring yang responsif terhadap level pemahaman pengguna.

Selain itu, struktur empat langkah yang dimiliki **KARANGTURI** membuatnya **mudah diajarkan dalam pendidikan vokasi atau pelatihan penggunaan AI**, sehingga cocok digunakan dalam kurikulum prompt engineering dasar di Indonesia.

Dalam pengujian di platform seperti **ChatGPT, Claude, dan Gemini**, prompt yang disusun dengan format **KARANGTURI** menunjukkan **tingkat koherensi dan ketepatan respons yang lebih tinggi** dibanding prompt generik atau tanpa struktur. Ini menunjukkan bahwa framework ini layak dikembangkan lebih lanjut sebagai kontribusi lokal Indonesia dalam ekosistem AI global.

Sebagai framework yang lahir dari konteks lokal, **KARANGTURI juga dapat diadaptasi menjadi alat bantu pengajaran, modul pelatihan AI di UMKM,**

serta pembuatan template prompt untuk startup digital berbasis konten atau layanan AI. Dengan demikian, **KARANGTURI** tidak hanya berfungsi sebagai pendekatan teknis, namun juga mengusung nilai edukatif dan pemberdayaan.

4. Conclusion

Penelitian ini telah melakukan analisis komparatif terhadap empat framework prompt engineering: **KARANGTURI, RTF, Chain-of-Thought (CoT), dan ReAct**, dengan mempertimbangkan struktur, kompleksitas, fokus, serta penggunaannya di berbagai platform LLM seperti ChatGPT, Gemini, Claude, dan LLaMA.

Dari hasil pembahasan, dapat disimpulkan bahwa setiap framework memiliki keunggulan dan konteks penerapan masing-masing. **RTF** unggul dalam kesederhanaan dan kecepatan, **CoT** dalam penalaran bertahap, dan **ReAct** dalam interaksi eksternal yang dinamis. Namun demikian, **KARANGTURI muncul sebagai pendekatan lokal yang unik dan menyeluruh**, menggabungkan aspek karakter, konteks, tujuan, dan batasan dalam satu struktur terpadu.

Kelebihan utama framework **KARANGTURI** terletak pada fleksibilitas dan relevansinya untuk berbagai skenario, mulai dari konten edukatif, komunikasi profesional, hingga pelatihan AI berbasis persona. Dalam pengujian pada berbagai model LLM, prompt yang disusun dengan format **KARANGTURI** mampu menghasilkan respons yang lebih terarah, kontekstual, dan sesuai dengan kebutuhan pengguna.

Framework ini juga memiliki potensi besar untuk dikembangkan lebih lanjut sebagai bagian dari kurikulum prompt engineering di Indonesia, sekaligus menjadi kontribusi lokal dalam literatur dan praktik AI global. Validasi lebih lanjut melalui eksperimen kuantitatif dan studi kasus praktis akan semakin memperkuat posisi *framework* **KARANGTURI** sebagai pendekatan strategis dalam dunia *prompt engineering* modern.

5. Acknowledgement

Penulis mengucapkan terima kasih yang sebesar-besarnya kepada rekan-rekan dosen yang telah memberikan masukan, dukungan, dan diskusi yang konstruktif selama proses penyusunan penelitian ini. Kontribusi pemikiran dan semangat kolaboratif yang diberikan sangat berarti dalam menyempurnakan kerangka kerja **KARANGTURI** dan memperkaya analisis dalam studi ini. Semoga karya ini dapat menjadi bagian dari pengembangan keilmuan di bidang teknologi, pendidikan, dan bisnis digital.

6. References

- Widagdo, H. H., & Bakti, C. A. (2021). Aplikasi pengetesan karakter personal berdasarkan metode DISC berbasis web. *Sains Teknologi Manajemen Jurnal (STMJ)*, 2(1), 18-25 <https://unkartur.ac.id/journal/index.php/stmj/article/view/15/15>
- Lukito, D. (2023). Hard skills and soft skills on performance: Influence and application of Bengkulu City Education Service employees. *East Asian Journal of Multidisciplinary Research*, 2(11), 4695-4710.
- Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., ... Amodi, D. (2020). Language models are few-shot learners (Version 4). *arXiv*. <https://arxiv.org/abs/2005.14165v4>
- Wei, J. et al. (2022). Chain of Thought Prompting Elicits Reasoning. *arXiv*. <https://arxiv.org/abs/2201.11903>
- Yao, S. et al. (2022). ReAct: Synergizing Reasoning and Acting. *arXiv*. <https://arxiv.org/abs/2210.03629>
- Vatsal, S., & Dubey, H. (2024). A survey of prompt engineering methods in large language models for different NLP tasks. *arXiv*. <https://arxiv.org/abs/2407.12994>
- Liu, P. et al. (2023). Pre-train Prompt Tune. *arXiv*. <https://arxiv.org/abs/2107.13586>
- Gu, J., Han, Z., Chen, S., Beirami, A., He, B., Zhang, G., Liao, R., Qin, Y., Tresp, V., & Torr, P. (2023). A systematic survey of prompt engineering on vision-language foundation models. *arXiv*. <https://arxiv.org/abs/2307.12980>
- OpenAI. (2023). GPT Best Practices. <https://platform.openai.com/docs/guides/gpt-best-practices>
- Anthropic. (2023). Prompting Claude Guide. <https://docs.anthropic.com/claude/docs/prompting-guide>
- Prompt Engineering Guide (DAIR.AI) <https://github.com/dair-ai/Prompt-Engineering-Guide>
- Google Cloud. (2024). Prompt Design Guide. <https://cloud.google.com/generative-ai/docs/prompt-design>