



PT SOLUSI
ADMINISTRASI
HUKUM

Prompt Engineering ✨

Muhammad Anwar Fauzi, S.Kom., M.Kom.
Putry Wahyu Setyaningsih, S.Kom., M.Kom.
Ir. Muhammad Yusuf, M.T.
Yulaikha Mar'atullatifah, S.Pd., M.Kom.
Ardy Wicaksono, S.Kom., M.Kom.
Ilman Kadori, S.Kom., M.Kom.
Faridatus Shofiyah, S.Pd.
Dwi Utari Iswavigra, S.T., M.Kom.
Very Dwi Setiawan, S.T., M.Cs.
Deny Prasetyo, S.Pd., M.Kom.
Suyahman, S.Kom., M.Kom.

Prompt Engineering

UU No 28 tahun 2014 tentang Hak Cipta

Fungsi dan sifat hak cipta Pasal 4

Hak Cipta sebagaimana dimaksud dalam Pasal 3 huruf a merupakan hak eksklusif yang terdiri atas hak moral dan hak ekonomi.

Pembatasan Pelindungan Pasal 26

Ketentuan sebagaimana dimaksud dalam Pasal 23, Pasal 24, dan Pasal 25 tidak berlaku terhadap:

- penggunaan kutipan singkat Ciptaan dan/atau produk Hak Terkait untuk pelaporan peristiwa aktual yang ditujukan hanya untuk keperluan penyediaan informasi aktual;

- Penggandaan Ciptaan dan/atau produk Hak Terkait hanya untuk kepentingan penelitian ilmu pengetahuan;

- Penggandaan Ciptaan dan/atau produk Hak Terkait hanya untuk keperluan pengajaran, kecuali pertunjukan dan Fonogram yang telah dilakukan Pengumuman sebagai bahan ajar; dan

- penggunaan untuk kepentingan pendidikan dan pengembangan ilmu pengetahuan yang memungkinkan suatu Ciptaan dan/atau produk Hak Terkait dapat digunakan tanpa izin Pelaku Pertunjukan, Produser Fonogram, atau Lembaga Penyiaran.

Sanksi Pelanggaran Pasal 113

Setiap Orang yang dengan tanpa hak melakukan pelanggaran hak ekonomi sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf i untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 1 (satu) tahun dan/atau pidana denda paling banyak Rp100.000.000 (seratus juta rupiah).

Setiap Orang yang dengan tanpa hak dan/atau tanpa izin Pencipta atau pemegang Hak Cipta melakukan pelanggaran hak ekonomi Pencipta sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf c, huruf d, huruf f, dan/atau huruf h untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 3 (tiga) tahun dan/atau pidana denda paling banyak Rp500.000.000,00 (lima ratus juta rupiah).

Prompt Engineering

Muhammad Anwar Fauzi, S.Kom., M.Kom.
Putry Wahyu Setyaningsih, S.Kom., M.Kom
Ir. Muhammad Yusuf, M.T.
Yulaikha Mar'atullatifah, S.Pd., M.Kom.
Ardy Wicaksono, S.Kom., M.Kom.
Ilman Kadori, S.Kom., M.Kom.
Faridatus Shofiyah, S.Pd.
Dwi Utari Iswavigra, S.T., M.Kom.
Very Dwi Setiawan, S.T., M.Cs.
Deny Prasetyo, S.Pd., M.Kom.
Suyahman, S.Kom., M.Kom.

PT Solusi Administrasi Hukum

Prompt Engineering

Nama Penulis

Muhammad Anwar Fauzi, S.Kom., M.Kom.
Putry Wahyu Setyaningsih, S.Kom., M.Kom.
Ir. Muhammad Yusuf, M.T.
Yulaikha Mar'atullatifah, S.Pd., M.Kom.
Ardy Wicaksono, S.Kom., M.Kom.
Ilman Kadori, S.Kom., M.Kom.
Faridatus Shofiyah, S.Pd.
Dwi Utari Iswavigra, S.T., M.Kom.
Very Dwi Setiawan, S.T., M.Cs.
Deny Prasetyo, S.Pd., M.Kom.
Suyahman, S.Kom., M.Kom.

Desain Cover :

Tim Sah Publisher

Tata Letak :

Tim Sah Publisher

Proofreader :

Tim Sah Publisher

Ukuran :

144 halaman, Uk: 17.5x25 cm

ISBN :

978-634-04-3935-9

Cetakan Pertama :

September 2025

Hak Cipta 2025, Pada Penulis

Isi diluar tanggung jawab percetakan

Copyright © 2025 by Sah Publisher

All Right Reserved

Hak cipta dilindungi undang-undang
Dilarang keras menerjemahkan, memfotokopi, atau
memperbanyak sebagian atau seluruh isi buku ini
tanpa izin tertulis dari Penerbit.

PENERBIT PT SOLUSI ADMINISTRASI HUKUM

Jl. Pedan Karangdowo, Munggun, Karangdowo, Klaten
Telp: +628562160034
Website: www.sah.co.id
publisher.sah.co.id
E-mail: publisher@sah.co.id

KATA PENGANTAR

Perkembangan teknologi kecerdasan buatan, khususnya large language models (LLM), telah menghadirkan tantangan sekaligus peluang yang signifikan bagi dunia akademik, industri, dan masyarakat luas. Dalam konteks ini, prompt engineering menjadi instrumen strategis yang menentukan kualitas interaksi antara manusia dan mesin. Keterampilan dalam merumuskan instruksi yang tepat bukan sekadar aspek teknis, melainkan fondasi epistemologis yang memastikan keluaran sistem kecerdasan buatan selaras dengan kebutuhan pengguna.

Urgensi penguasaan prompt engineering semakin terasa karena kualitas hasil yang dihasilkan LLM secara langsung dipengaruhi oleh struktur, kejelasan, dan konteks instruksi yang diberikan. Dengan demikian, prompt engineering dapat dipandang sebagai keterampilan literasi baru yang tidak hanya relevan dalam bidang teknologi, tetapi juga penting dalam ranah pendidikan, penelitian ilmiah, serta pengambilan keputusan berbasis data. Pengabaian terhadap aspek ini berpotensi menimbulkan bias, kesalahan interpretasi, maupun ketidakefisienan dalam pemanfaatan kecerdasan buatan.

Buku ini disusun untuk memberikan landasan teoretis sekaligus panduan praktis mengenai prompt engineering. Melalui pendekatan sistematis, diharapkan pembaca memperoleh pemahaman yang komprehensif mengenai konsep, prinsip, dan aplikasi prompt engineering sehingga mampu mengoptimalkan pemanfaatan kecerdasan buatan dalam berbagai bidang. Dengan demikian, buku ini diharapkan dapat berkontribusi pada pengembangan literasi digital yang lebih kritis, reflektif, dan adaptif terhadap dinamika teknologi masa kini.

Penulis

DAFTAR ISI

KATA PENGANTAR.....	5
DAFTAR ISI.....	6
BAB I.....	1
DASAR-DASAR LARGE LANGUAGE MODELS (LLM).....	1
1.1 Pengertian Large Language Models (LLM).....	2
1.2 Fungsi dan Mekanisme Kerja LLM.....	3
1.3 Konsep Token, Context Window, dan Inference.....	4
1.4 Monomodal dan Multimodal Models.....	8
1.5 Evolusi Model Bahasa.....	10
1.6 Arsitektur LLM.....	13
1.7 Dataset dan Proses Pelatihan.....	14
1.8 Parameter, Skala, dan Infrastruktur.....	16
1.9 Evaluasi dan Benchmarking.....	17
1.10 Contoh Penggunaan LLM.....	19
BAB II.....	22
RAGAM MODEL LLM POPULER.....	22
2.1 ChatGPT.....	22
GPT-3.5.....	23
GPT-4.....	23
GPT-4o.....	24
GPT-5.....	25
2.2 Gemini.....	28
2.3 Claude.....	29
2.4 Model Alternatif dan Open-Source.....	30
Mistral.....	31
LLaMA (Meta AI).....	31
Command R+ (Cohere).....	31
Grok (xAI).....	32
2.5 Perbandingan Karakteristik dan Kecocokan Penggunaan.....	32
BAB III.....	36
DASAR-DASAR PROMPT ENGINEERING.....	36
3.1 Pengantar Prompt Engineering.....	36
3.2 Konsep Dasar Prompt Engineering.....	37

3.3 Kesalahan Umum dalam Prompt Engineering.....	39
BAB IV.....	41
TEKNIK PROMPTING TINGKAT LANJUT.....	41
4.1 Chain-of-Thought (CoT) Prompting.....	41
4.2 Role Prompting (“Act as...”.....)	43
4.3 Self-asking dan Self-consistency.....	45
4.4 ReAct Prompting (Reasoning + Action).....	46
4.5 Iterative Prompting & Refinement.....	48
4.6 Step-back Prompting.....	50
4.7 Tree-of-Thought (ToT) Prompting.....	52
4.8 Automatic Prompt Engineering (APE).....	53
4.9 Multimodal Prompting.....	55
4.10 Prompt Chaining (Multi-step Prompting).....	56
4.11 Contrastive Prompting.....	58
4.12 Hypothetical Document Embeddings (HyDE).....	59
4.13 Context Distillation Prompting.....	61
4.14 Persona/Character Prompting.....	62
4.15 Multi-Agent Prompting.....	64
BAB V.....	66
PROMPT UNTUK TEKS DAN BAHASA.....	66
5.1 Prompt untuk Penulisan Kreatif.....	66
5.2 Prompt untuk Analisis Teks.....	68
5.3 Prompt untuk Coding.....	69
5.4 Prompt untuk Pendidikan dan Soal Interaktif.....	73
5.5 Prompt untuk Analisis Data & Visualisasi.....	77
BAB VI.....	79
PROMPT MULTIMODAL.....	79
6.1. Prompt Gambar (Image Generation).....	80
Prompt Teks AI.....	80
Petunjuk Gambar AI.....	82
6.2 Prompt Video.....	83
Teks ke Video.....	83
Gambar ke Video.....	84
6.3 Prompt Musik dan Audio.....	86
Cara Kerja Musik AI.....	86

Interpretasi AI.....	87
Generasi Musik.....	88
Prompt Musik Teknologi dan Suno AI.....	89
5 Elemen Kunci dari Prompt Musik AI yang Efektif.....	90
BAB VII.....	93
EVALUASI DAN OPTIMASI PROMPT.....	93
7.1 Dasar - Dasar Prompt.....	93
7.2 Minimal Kata VS Prompt Panjang.....	95
Prompt Singkat: Cepat dan Efisien.....	95
Prompt Panjang: Detail dan Terarah.....	96
Aturan Praktis Panjang Prompt.....	97
Studi Kasus: Pembuatan Gambar.....	97
7.3 Membangun Prompt Yang Efektif.....	98
Tujuan.....	99
Konteks.....	99
Gaya.....	99
Format.....	100
7.4 Optimasi Prompt.....	100
7.5 Masa Depan Prompt.....	102
BAB VIII.....	104
APLIKASI PROMPT ENGINEERING DI DUNIA NYATA.....	104
8.1 Prompt dalam Bisnis dan Automasi Kantor.....	104
Penyusunan Dokumen Otomatis.....	105
Asisten Virtual Kantor.....	105
Analisis Pasar dan Pelanggan.....	105
Automasi Komunikasi Bisnis.....	105
8.2 Prompt untuk Industri Kreatif dan Media.....	106
Penulisan Kreatif.....	106
Desain dan Produksi Konten Visual.....	107
Kampanye Media Sosial.....	107
Kolaborasi Kreatif Manusia–AI.....	107
8.3 Prompt untuk Legal, Finansial, dan Pemerintahan.....	108
Bidang Hukum.....	109
Bidang Finansial.....	109
Sektor Pemerintahan.....	109

Aspek Etika & Hukum.....	109
8.4 Prompt dalam Dunia Pendidikan dan Riset.....	110
Pembelajaran Personal.....	111
Pembuatan Soal dan Simulasi Ujian.....	111
Penulisan Akademik.....	111
Eksperimen Riset.....	111
8.5 Studi Kasus Prompt Project dan Workflows.....	112
Studi Kasus Bisnis.....	113
Studi Kasus Industri Kreatif.....	113
Studi Kasus Legal & Pemerintahan.....	113
Studi Kasus Pendidikan.....	113
BAB IX.....	115
ETIKA, KEAMANAN, DAN RISIKO.....	115
9.1 Tantangan Etika dalam Penggunaan LLM.....	116
Bias Dan Diskriminasi Dalam Output.....	116
Framing Dan Manipulasi Prompt.....	116
Halusinasi (Fabricated Information).....	117
Tanggung Jawab Pengguna Vs Pengembang.....	117
9.2 Keamanan Interaksi dengan Model AI.....	118
Prompt Injection dan Prompt Leaks.....	118
Data Sensitif dan Privasi.....	118
Risiko Penggunaan dalam Sektor Kritis.....	119
Kontrol terhadap Konten Berbahaya.....	119
9.3 Hak Cipta, Kepemilikan, dan Lisensi.....	119
Status hukum output AI.....	120
Panduan resmi penyedia LLM.....	120
Contoh Kasus Hukum.....	121
Rekomendasi Penggunaan LLM.....	121
9.4 Risiko Sosial dan Dampak Luas.....	121
Ketergantungan Berlebihan pada AI.....	121
Penyebaran Disinformasi dan Deepfake.....	122
Etika Distribusi dan Akses.....	122
Dampak Lingkungan.....	122
9.5 Prinsip-Prinsip Desain Prompt yang Etis dan Aman.....	123
Transparansi.....	123

Akuntabilitas.....	123
Keadilan.....	123
Privasi.....	123
Ketaatan Hukum.....	124
9.6 Rekomendasi.....	124
Bagi Pengguna Umum.....	124
Bagi Peneliti dan Pendidik.....	124
Bagi Bisnis dan Industri.....	125
Bagi Pemerintah dan Regulator.....	125
BAB X.....	126
MASA DEPAN PROMPT ENGINEERING.....	126
10.1 Ke Depan, Ada Beberapa Arah Perkembangan Prompt Engineering:	127
10.2 Masa Depan Prompt Engineering: Dari Karier hingga Vibe Coding..	128
10.2.1 Prompt Engineering sebagai Karier Baru.....	129
10.2.2 Dari Coding ke “Vibe Coding”.....	130
10.2.3 Evolusi Hubungan Manusia dan Mesin.....	131
10.2.4 Tantangan Etis dan Sosial.....	132
10.2.5 Masa Depan yang Kolaboratif.....	133
DAFTAR PUSTAKA.....	135

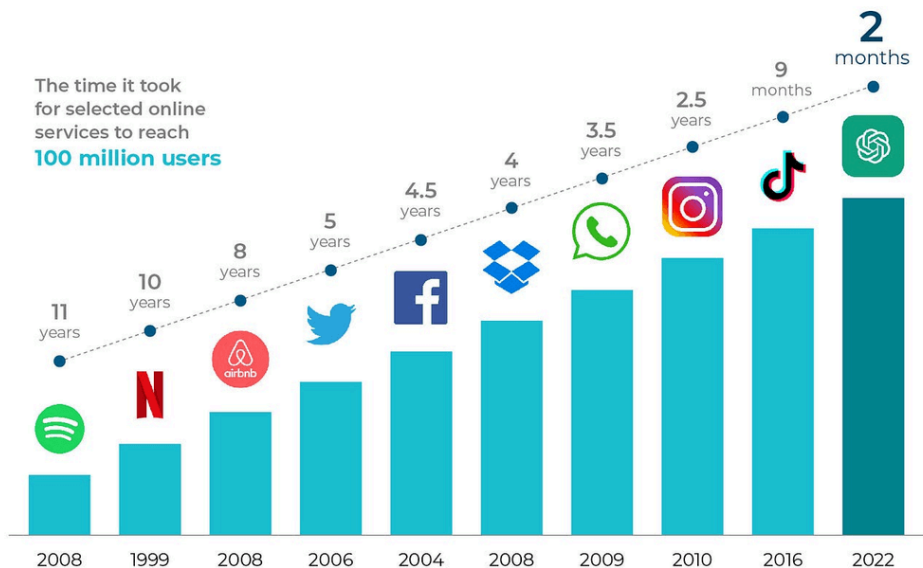
BAB I

DASAR-DASAR LARGE LANGUAGE MODELS (LLM)

Perkembangan kecerdasan buatan dalam dekade terakhir telah membawa perubahan mendasar pada cara manusia berinteraksi dengan teknologi. Salah satu lompatan paling signifikan adalah lahirnya model bahasa berskala besar yang mampu memahami sekaligus menghasilkan teks menyerupai manusia. Bahkan ChatGPT bisa mendapatkan 100 juta pengguna pertama hanya dalam waktu 2 bulan. Kehadiran teknologi ini tidak hanya mengubah ranah penelitian akademik, tetapi juga membuka peluang baru dalam bidang pendidikan, hukum, bisnis, dan komunikasi sehari-hari. Untuk memahami bagaimana model ini bekerja, diperlukan pemahaman yang kokoh mengenai konsep dasar yang melandasinya



Chat-GPT sprints to 100 million users



Waktu yang Dibutuhkan Perusahaan Teknologi untuk Mendapatkan 100 Juta Pengguna Pertama (Netscribes)

1.1 Pengertian Large Language Models (LLM)



Mengenal LLM

Large Language Models (LLM) adalah sistem kecerdasan buatan yang dirancang untuk memahami dan menghasilkan bahasa manusia secara mendalam. Model ini dilatih pada jumlah data teks yang sangat besar, mencakup buku, artikel, dokumen, hingga konten dari internet, sehingga mampu mengenali pola, struktur, dan konteks bahasa dengan tingkat keakuratan yang luar biasa. Disebut “large” karena model ini memiliki jumlah parameter yang sangat besar, mulai dari miliaran hingga triliunan. Bahkan GPT-5 memiliki parameter 5-10 Triliun. Parameter tersebut berfungsi sebagai bobot dalam jaringan saraf tiruan yang mengatur bagaimana informasi diolah, disimpan, dan diproyeksikan kembali dalam bentuk keluaran. Skala parameter yang sangat besar inilah yang memungkinkan LLM meniru kompleksitas bahasa manusia dengan tingkat keakuratan dan keluwesan yang tinggi.

Tabel Perbandingan Parameter LLM GPT-4 dengan GPT-5 (felloai.com)

Atribut	GPT-4	GPT-5 (Perkiraan)
Jumlah Parameter	~1 Triliun (1.000 miliar)	5–10 Triliun (5.000–10.000 miliar)
Ukuran Dataset	~10 Triliun token	40–100 Triliun token
Perangkat Pelatihan	25.000 GPU A100	250.000–500.000 GPU H100
Biaya Pelatihan	~100 Juta USD	1,25–2,5 Miliar USD

LLM memungkinkan mesin untuk menanggapi pertanyaan, menulis esai, menyusun kode, memahami emosi dalam teks, bahkan berinteraksi secara natural seperti manusia, semua melalui proses pemrosesan bahasa alami (Natural Language Processing, NLP).

1.2 Fungsi dan Mekanisme Kerja LLM

Large Language Model (LLM) tidak “memahami” bahasa sebagaimana manusia melakukannya. Model ini bekerja berdasarkan pola statistik yang dipelajari dari miliaran hingga triliunan kata yang ada di dataset pelatihannya. LLM bekerja dengan memprediksi token berikutnya dalam sebuah urutan teks berdasarkan token-token sebelumnya. Token adalah satuan teks yang dapat berupa kata, sub-kata, atau simbol tertentu, bergantung pada metode tokenisasi yang digunakan. Melalui prediksi berulang-ulang, model dapat menyusun kalimat, paragraf, bahkan teks panjang yang koheren. Mekanisme ini pada dasarnya berbasis probabilistik, di mana model menghitung kemungkinan munculnya token tertentu dalam konteks yang sedang diproses. Semakin besar dan terlatih model tersebut, semakin baik kemampuannya dalam menghasilkan teks yang menyerupai bahasa alami manusia. Dengan memanfaatkan pola tersebut, LLM dapat memprediksi kata, frasa, atau kalimat berikutnya dengan akurasi yang sangat tinggi.

Secara garis besar, fungsi utama LLM meliputi:

- Pemahaman konteks: Mengidentifikasi struktur dan makna input teks.
- Generasi bahasa alami: Menghasilkan kalimat yang kohesif dan relevan.
- Penyelesaian tugas spesifik: Seperti menjawab pertanyaan, meringkas teks, menerjemahkan bahasa, hingga menulis kode program.

Perkembangan LLM berakar dari berbagai pendekatan dalam Natural Language Processing (NLP). Pada tahap awal, NLP mengandalkan model statistik seperti n-gram, yang bekerja dengan menghitung frekuensi kemunculan kata dalam sebuah korpus. Walaupun sederhana, pendekatan ini hanya efektif pada konteks terbatas dan tidak mampu menangkap relasi kata dalam rentang panjang.

Kemudian, generasi berikutnya diperkuat dengan hadirnya jaringan saraf berulang (Recurrent Neural Network atau RNN) dan variannya, seperti Long Short-Term Memory (LSTM). RNN dan LSTM dapat memproses data berurutan

dan mempertahankan memori jangka pendek, tetapi tetap terbatas ketika menangani teks panjang akibat masalah vanishing gradient.

Terobosan besar muncul dengan diperkenalkannya arsitektur transformer pada tahun 2017. Transformer menggunakan mekanisme self-attention yang memungkinkan model memproses seluruh urutan input secara paralel dan menangkap relasi antarkata pada jarak jauh. Inovasi ini menjadi fondasi utama bagi LLM modern.

Selain arsitektur, perkembangan LLM juga dipengaruhi oleh konsep scaling law. Prinsip ini menyatakan bahwa kualitas model akan meningkat secara konsisten seiring dengan bertambahnya jumlah parameter, volume data pelatihan, dan kapasitas komputasi. Dengan kata lain, semakin besar skala model dan data yang digunakan, semakin baik pula performa yang dihasilkan. Prinsip scaling law inilah yang mendorong pengembangan model raksasa seperti GPT-3, GPT-4, Gemini, atau LLaMA, yang mampu menunjukkan kemampuan bahasa yang mendekati manusia.

Berbeda dengan model AI tradisional yang biasanya dirancang untuk tugas tertentu, LLM bersifat sebagai general-purpose foundation model. Model tradisional, misalnya, hanya dilatih untuk klasifikasi sentimen atau penerjemahan bahasa. Sebaliknya, LLM dapat digunakan untuk berbagai tugas dengan sedikit atau tanpa penyesuaian, seperti menjawab pertanyaan, meringkas teks, menerjemahkan, hingga menghasilkan kode program. Sifat general-purpose ini menjadikan LLM sebagai pondasi yang fleksibel bagi beragam aplikasi kecerdasan buatan modern, sekaligus menandai pergeseran besar dalam paradigma pemrosesan bahasa alami.

1.3 Konsep Token, Context Window, dan Inference

Saat memahami cara kerja Large Language Models (LLM), terdapat tiga konsep mendasar yang tidak dapat dipisahkan, yaitu token, context window, dan inference. Ketiga elemen ini membentuk fondasi operasional LLM dalam menerima masukan, memproses informasi, dan menghasilkan keluaran. Token berfungsi sebagai unit dasar teks yang dapat dibaca oleh model, context window menentukan sejauh mana model mampu mengingat dan menghubungkan informasi dalam satu kali pemrosesan, sedangkan inference menjelaskan bagaimana model menghasilkan prediksi atau keluaran berdasarkan informasi yang telah dipelajari.

Token

Dalam Large Language Models (LLM), bahasa tidak diproses secara langsung dalam bentuk kata atau kalimat, melainkan diuraikan ke dalam unit terkecil yang disebut token. Token dapat berupa kata utuh, suku kata, atau potongan sub-kata, tergantung pada metode tokenisasi yang digunakan. Proses tokenisasi bertujuan untuk mengubah teks menjadi representasi yang dapat dipahami model. Beberapa metode tokenisasi populer antara lain sebagai berikut.

- WordPiece: memecah kata menjadi sub-unit jika kata tersebut jarang muncul.
- Byte Pair Encoding (BPE): menggabungkan pasangan karakter atau sub-kata yang sering muncul.
- SentencePiece: metode berbasis statistik yang fleksibel dan sering digunakan pada model modern.

Proses Kerja Tokenisasi (Tokenization) adalah sebagai berikut.

- Teks masukan dipecah menjadi unit-unit kecil yang disebut token.
- Token bisa berupa kata, suku kata, bahkan potongan huruf, tergantung model dan tokenizer yang digunakan.
- Contoh: Kata “*menghasilkan*” akan dipecah menjadi [“meng”, “hasil”, “kan”].
- Tokenisasi ini penting agar model dapat menangani kosakata yang sangat luas dengan efisien.

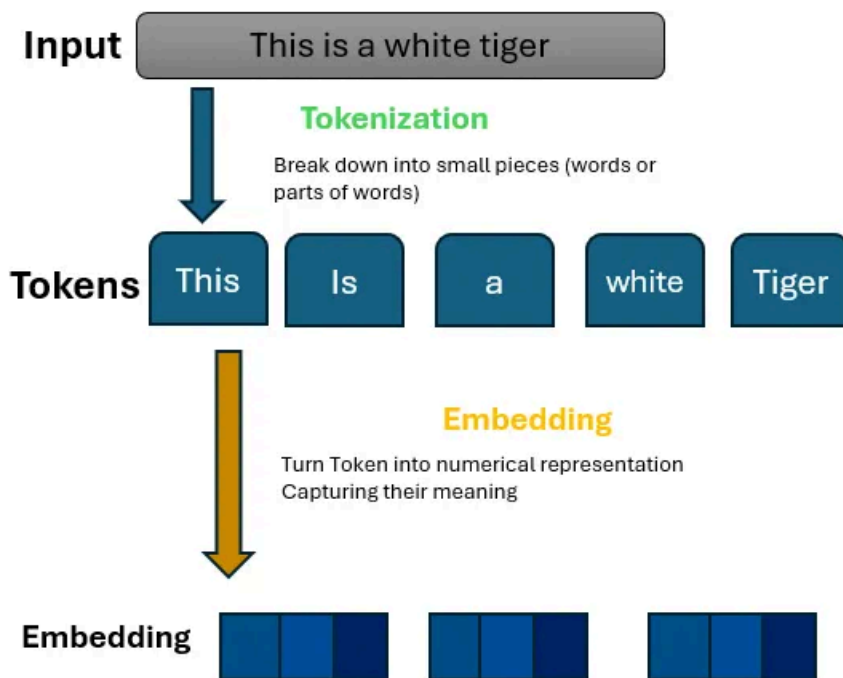
Dengan tokenisasi, sebuah kata panjang seperti “menghasilkan” dapat diuraikan menjadi beberapa token, misalnya [“meng”, “hasil”, “kan”]. Hal ini membuat model tidak harus menghafal seluruh bentuk kata dalam bahasa, melainkan cukup memahami bagian-bagiannya.

Embedding

Setelah teks diubah menjadi token, setiap token kemudian dipetakan ke dalam bentuk numerik berdimensi tinggi yang disebut embedding. Embedding merupakan representasi matematis yang memungkinkan model menangkap makna semantik dari token dan menghubungkannya dengan token lain dalam suatu ruang vektor.

Fungsi embedding sangat penting karena menjembatani teks manusia dengan sistem matematis yang dapat dipahami mesin. Setelah teks dipecah menjadi unit-unit kecil yang disebut token melalui proses tokenisasi, setiap token diubah menjadi vektor berdimensi tinggi yang dikenal sebagai embedding. Representasi vektor ini memungkinkan model memetakan makna semantik

sebuah token ke dalam ruang matematis, di mana hubungan antar kata dapat diukur berdasarkan jarak atau kedekatannya. Token yang memiliki makna serupa akan ditempatkan berdekatan dalam ruang vektor, sementara token dengan makna berbeda akan ditempatkan lebih jauh. Sebagai contoh, kata “bank” dalam konteks “bank uang” akan berada lebih dekat dengan kata “money”, sedangkan dalam konteks “bank sungai” akan lebih dekat dengan kata “river”. Perbedaan ini muncul karena embedding dilatih untuk menangkap hubungan semantik dan konteks penggunaan kata, sehingga menjadi dasar bagi kemampuan LLM memahami sinonim, asosiasi makna, serta nuansa bahasa dalam berbagai domain.



Proses Tokenization Menjadi Embedding

Setelah token direpresentasikan dalam bentuk embedding, tahap berikutnya adalah pemrosesan melalui mekanisme perhatian atau attention. Mekanisme ini, khususnya self-attention, memungkinkan model memahami keterkaitan antar-token di seluruh bagian kalimat atau paragraf secara bersamaan. Dengan demikian, model tidak hanya melihat hubungan kata yang berdekatan, tetapi juga mampu menangkap relasi kata yang jauh terpisah. Misalnya, dalam kalimat

“Anjing itu berlari cepat karena ingin mengejar bola,” kata “bola” memiliki hubungan yang lebih kuat dengan kata “mengejar” dibandingkan dengan kata lainnya. Melalui self-attention, model dapat memberikan bobot yang lebih besar pada hubungan semacam ini sehingga menghasilkan representasi konteks yang lebih akurat.

Tahap akhir dari proses kerja LLM adalah prediksi keluaran atau next-token prediction. Berdasarkan representasi konteks dan bobot probabilitas yang telah dihitung selama proses attention, model memprediksi token yang paling mungkin muncul setelah urutan token sebelumnya. Proses ini dilakukan secara berulang, di mana setiap token baru yang diprediksi ditambahkan ke dalam urutan untuk kembali diproses, hingga terbentuk kalimat atau teks yang utuh. Karena prediksi ini bersifat probabilistik, keluaran LLM sering kali terasa alami dan bervariasi, meskipun pada kenyataannya model tidak benar-benar “memahami” bahasa seperti manusia. Model hanya mengandalkan pola dan distribusi probabilitas yang dipelajari dari data pelatihan, tetapi kemampuan ini sudah cukup untuk menghasilkan teks dengan kualitas yang menyerupai hasil karya manusia.

Context Window

Context window adalah jumlah maksimum token yang dapat diproses oleh model dalam satu kali masukan. Batasan ini berfungsi seperti “memori jangka pendek” bagi LLM, karena menentukan seberapa jauh model dapat mengingat dan mempertahankan konteks.

Ukuran context window bervariasi antar model. GPT-3, misalnya, memiliki kapasitas sekitar 2.048 token, sementara GPT-4 memperluas hingga 32.000 token. Model terbaru seperti GPT-4o bahkan mampu menangani hingga 128.000 token dalam sekali input. Semakin panjang context window, semakin baik model dalam menjaga koherensi pada teks panjang, misalnya ketika menganalisis dokumen hukum, riset ilmiah, atau percakapan multi-babak.

Namun, jika input melebihi kapasitas jendela konteks, sebagian teks akan dipotong (truncated), sehingga informasi penting bisa hilang. Inilah salah satu keterbatasan fundamental LLM yang mendorong penelitian baru dalam memperluas atau mengoptimalkan memori model.

Inference

Inference adalah tahap di mana model menghasilkan keluaran berdasarkan input yang telah diproses. Pada tahap ini, model menggunakan parameter yang telah dilatih untuk menghitung distribusi probabilitas token berikutnya, lalu memilih token yang paling sesuai untuk membentuk teks lanjutan.

Proses inference dapat dilakukan dengan berbagai metode decoding:

- Greedy decoding, yang selalu memilih token dengan probabilitas tertinggi, sederhana namun sering menghasilkan keluaran monoton.
- Beam search, yang mengeksplorasi beberapa jalur kemungkinan sekaligus untuk menemukan hasil yang paling optimal.
- Top-k sampling, yang membatasi pilihan hanya pada k token dengan probabilitas tertinggi.
- Nucleus sampling (top-p), yang memilih dari distribusi kumulatif hingga ambang probabilitas tertentu, menghasilkan teks lebih alami dan variatif.

Karena inference pada LLM besar membutuhkan sumber daya komputasi yang tinggi, digunakan berbagai teknik optimasi, seperti caching (menyimpan hasil perhitungan sebelumnya), quantization (mengurangi presisi angka untuk mempercepat proses), serta distillation (melatih model lebih kecil agar meniru perilaku model besar).

Walaupun inference menghasilkan teks yang tampak alami, penting dipahami bahwa LLM tidak benar-benar “mengerti” bahasa seperti manusia. Model hanya melakukan prediksi probabilistik berdasarkan pola dalam data pelatihan. Hal ini menjelaskan mengapa terkadang muncul keluaran yang tidak akurat atau fenomena hallucination.

1.4 Monomodal dan Multimodal Models

Large Language Models (LLM) pada dasarnya dikembangkan untuk memproses bahasa alami dalam bentuk teks. Model yang hanya menerima satu jenis input ini disebut monomodal. Monomodal LLM bekerja secara eksklusif pada data teks, sehingga seluruh masukan dan keluaran berbentuk rangkaian kata atau kalimat. Contoh model monomodal adalah GPT-3 dan LLaMA generasi awal. Kedua model ini menunjukkan performa luar biasa dalam pemrosesan teks, seperti menjawab pertanyaan, meringkas dokumen, dan menghasilkan tulisan kreatif. Akan tetapi, keterbatasannya terletak pada kemampuan yang hanya berpusat pada satu modality, yaitu teks, sehingga tidak dapat langsung memahami informasi visual, audio, atau multimodal yang semakin relevan dalam kehidupan sehari-hari.

Berbeda dengan itu, model multimodal dirancang untuk menerima dan memproses lebih dari satu jenis input. Multimodal LLM mampu menggabungkan teks dengan gambar, suara, atau bahkan video dalam satu sistem pemrosesan (fusion). Hal ini menjadikannya lebih fleksibel dan lebih dekat dengan cara manusia memahami dunia, karena manusia secara alami mengintegrasikan

berbagai indera untuk membentuk pemahaman yang utuh. Contoh dari model multimodal kontemporer adalah GPT-4o, Gemini, dan Claude Vision, yang dapat menganalisis teks sekaligus menjawab pertanyaan berbasis gambar atau mendeskripsikan konten visual yang diberikan pengguna.



LLM Multimodal Mampu Memproses Teks Dengan Gambar, Suara, Atau Bahkan Video Dalam Satu Sistem Pemrosesan (Dataiku)

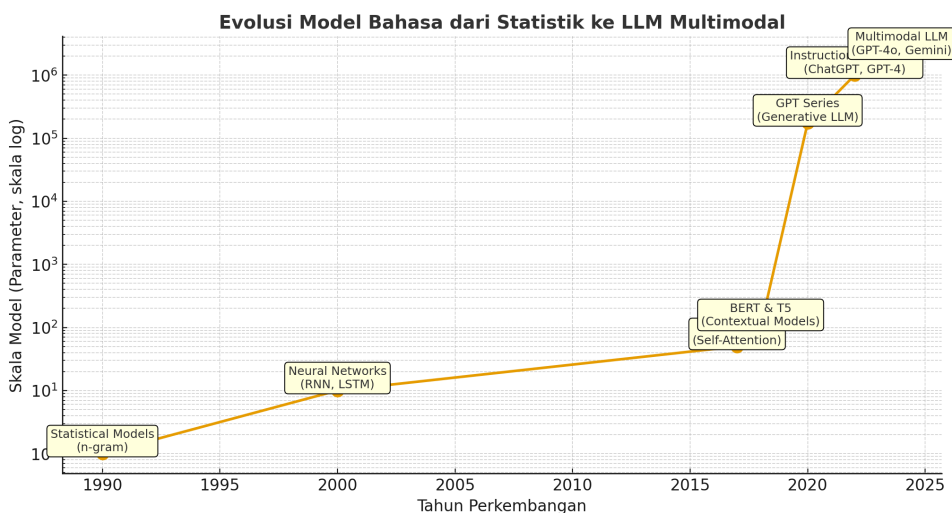
Secara teknis, arsitektur multimodal menggunakan konsep penggabungan representasi atau fusion pada tahap embedding. Setiap modality, baik teks, gambar, maupun audio, diubah terlebih dahulu ke dalam bentuk embedding yang sesuai dengan karakteristik data masing-masing. Representasi ini kemudian digabungkan dalam ruang vektor yang sama sehingga model dapat memahami keterkaitan antar-modality. Misalnya, dalam sebuah sistem text-to-image, embedding teks yang berisi deskripsi seperti “seekor kucing hitam di atas kursi” dipadukan dengan embedding visual yang memungkinkan model menghasilkan gambar sesuai deskripsi tersebut. Proses fusion inilah yang membuat multimodal model mampu menghubungkan kata dengan objek visual, suara dengan teks, atau bahkan gerakan dalam video dengan narasi yang relevan.

Kemampuan multimodal membuka jalan bagi berbagai aplikasi praktis. Chatbot berbasis suara, misalnya, memungkinkan interaksi manusia dengan mesin dalam bentuk percakapan yang lebih alami karena menggabungkan pengenalan suara, pemahaman teks, dan sintesis suara. Model multimodal juga mendukung teknologi text-to-image yang mampu menghasilkan ilustrasi atau desain grafis hanya dari deskripsi verbal. Selain itu, analisis video berbantuan multimodal memungkinkan sistem mendeteksi peristiwa tertentu, mengidentifikasi objek, atau menganalisis interaksi kompleks dalam rekaman visual. Dengan kemampuan ini, multimodal LLM menjadi fondasi bagi inovasi di bidang kesehatan, pendidikan, hiburan, hingga keamanan.

Namun, di balik potensinya yang besar, pengembangan multimodal model menghadapi sejumlah tantangan. Salah satu persoalan utama adalah keselarasan antar-modality. Tidak selalu mudah bagi model untuk memastikan bahwa representasi visual, teks, dan audio benar-benar sesuai dan saling mendukung. Masalah lain adalah tingginya biaya pelatihan. Multimodal memerlukan dataset yang jauh lebih besar dan beragam, serta infrastruktur komputasi yang lebih kuat dibandingkan dengan model monomodal. Selain itu, evaluasi performa multimodal juga lebih kompleks. Jika pada model teks evaluasi bisa menggunakan metrik seperti perplexity atau BLEU, maka pada multimodal perlu ada pendekatan khusus untuk menilai sejauh mana teks sesuai dengan gambar atau bagaimana model memahami hubungan antara suara dan teks.

1.5 Evolusi Model Bahasa

Perkembangan model bahasa merupakan perjalanan panjang yang dipengaruhi oleh kemajuan teori linguistik, metode statistik, dan kecerdasan buatan. Evolusi ini menunjukkan bagaimana pemrosesan bahasa alami (Natural Language Processing/NLP) bertransformasi dari model sederhana berbasis frekuensi kata hingga menjadi model raksasa dengan miliaran parameter yang mampu menirukan bahasa manusia dengan sangat realistis.



Perkembangan LLM dari Waktu ke Waktu

Tahap paling awal dalam evolusi model bahasa ditandai oleh pendekatan statistik. Model seperti n-gram digunakan untuk menghitung probabilitas kemunculan suatu kata berdasarkan sejumlah kata sebelumnya. Misalnya, model bigram memprediksi kata berikutnya hanya dengan melihat satu kata sebelumnya, sedangkan trigram menggunakan dua kata sebelumnya. Meskipun sederhana, pendekatan ini menghadapi keterbatasan signifikan, terutama dalam mengatasi data sparsity (kelangkaan data) dan ketidakmampuannya menangkap relasi jangka panjang dalam teks.

Kemajuan berikutnya hadir melalui penggunaan jaringan saraf tiruan (neural networks). Pada awal 2000-an, feedforward neural language models mulai diperkenalkan dengan representasi kata berbasis vektor yang dikenal sebagai word embeddings. Representasi ini memungkinkan kata-kata dengan makna serupa memiliki kedekatan matematis dalam ruang vektor. Namun, model ini masih terbatas dalam menangani teks panjang. Perkembangan lebih lanjut muncul dengan hadirnya Recurrent Neural Networks (RNN) dan variannya, Long Short-Term Memory (LSTM) serta Gated Recurrent Unit (GRU). RNN mampu memproses urutan kata secara berurutan sehingga dapat menangkap konteks yang lebih panjang dibandingkan n-gram. LSTM kemudian memperbaiki kelemahan RNN dalam menghadapi vanishing gradient problem, sehingga lebih efektif dalam mengingat informasi jangka panjang. Walaupun begitu, model berbasis RNN tetap memiliki keterbatasan dalam skalabilitas, terutama ketika dihadapkan pada teks yang sangat panjang dan kompleks.

Terobosan besar berikutnya terjadi dengan diperkenalkannya arsitektur transformer oleh Vaswani et al. (2017) melalui makalah “Attention is All You Need.” Transformer memperkenalkan mekanisme self-attention yang memungkinkan model melihat keseluruhan urutan teks secara paralel, bukan hanya secara berurutan seperti RNN. Hal ini menjadikan transformer jauh lebih efisien dalam pelatihan sekaligus lebih kuat dalam menangkap relasi antar kata yang berjauhan. Inovasi ini kemudian menjadi dasar dari hampir seluruh LLM modern.

Berdasarkan arsitektur transformer, lahirlah model-model besar yang semakin mengubah lanskap NLP. Salah satunya adalah BERT (Bidirectional Encoder Representations from Transformers) yang diperkenalkan oleh Google pada tahun 2018. BERT membawa konsep bidirectional context, artinya model memahami sebuah kata dengan mempertimbangkan konteks sebelum dan sesudahnya secara bersamaan. Pendekatan ini menjadikan BERT sangat efektif dalam berbagai tugas NLP seperti klasifikasi teks, ekstraksi entitas, dan tanya-jawab berbasis teks. Bersamaan dengan itu, model lain seperti T5

(Text-to-Text Transfer Transformer) memperluas pendekatan dengan menyatukan berbagai tugas NLP dalam kerangka text-to-text, sehingga setiap masalah dipandang sebagai proses mengubah teks masukan menjadi teks keluaran.

Gelombang berikutnya datang dengan munculnya model generatif skala besar. Generative Pre-trained Transformer (GPT) yang dikembangkan oleh OpenAI menjadi pelopor dalam kategori ini. GPT-1 memperkenalkan konsep pretraining pada data besar kemudian fine-tuning untuk tugas spesifik. GPT-2 memperluas kapasitas dengan ratusan juta parameter, menghasilkan keluaran teks yang lebih koheren, meskipun awalnya tidak dirilis sepenuhnya karena kekhawatiran penyalahgunaan. Lompatan besar terjadi dengan GPT-3, yang memiliki 175 miliar parameter dan menunjukkan kemampuan luar biasa dalam menulis esai, menjawab pertanyaan, hingga membuat kode program hanya dengan instruksi berupa prompt sederhana.

Selanjutnya, evolusi model bahasa memasuki era instruction tuning dan reinforcement learning from human feedback (RLHF). GPT-3.5 dan ChatGPT dilatih untuk lebih patuh pada instruksi manusia, sehingga interaksi menjadi lebih natural dan bermanfaat. GPT-4 kemudian memperluas kemampuan ini dengan konteks yang lebih panjang, pemahaman yang lebih halus, serta kemampuan multimodal. Model lain seperti Claude, Gemini, dan LLaMA 2 menandai lahirnya ekosistem model open-source maupun komersial yang semakin kompetitif.

Tren terkini menunjukkan bahwa evolusi model bahasa tidak hanya mengarah pada ukuran yang semakin besar, tetapi juga pada efisiensi dan diversifikasi fungsi. Muncul konsep scaling law yang menyatakan bahwa performa model meningkat seiring pertambahan parameter, data, dan komputasi. Namun, penelitian terbaru juga berfokus pada model yang lebih ringan, hemat energi, dan dapat dijalankan di perangkat dengan keterbatasan sumber daya, misalnya melalui quantization, distillation, atau low-rank adaptation.

Dengan demikian, perjalanan evolusi model bahasa mencerminkan kombinasi antara pencarian arsitektur yang lebih canggih, peningkatan skala parameter, dan penyesuaian dengan kebutuhan praktis. Dari n-gram sederhana hingga LLM multimodal modern, setiap tahap perkembangan membawa dampak besar bagi cara manusia dan mesin berinteraksi melalui bahasa. Masa depan model bahasa kemungkinan akan ditandai dengan keseimbangan antara kekuatan generatif, kemampuan multimodal, dan efisiensi yang memungkinkan penerapannya dalam skala yang lebih luas.

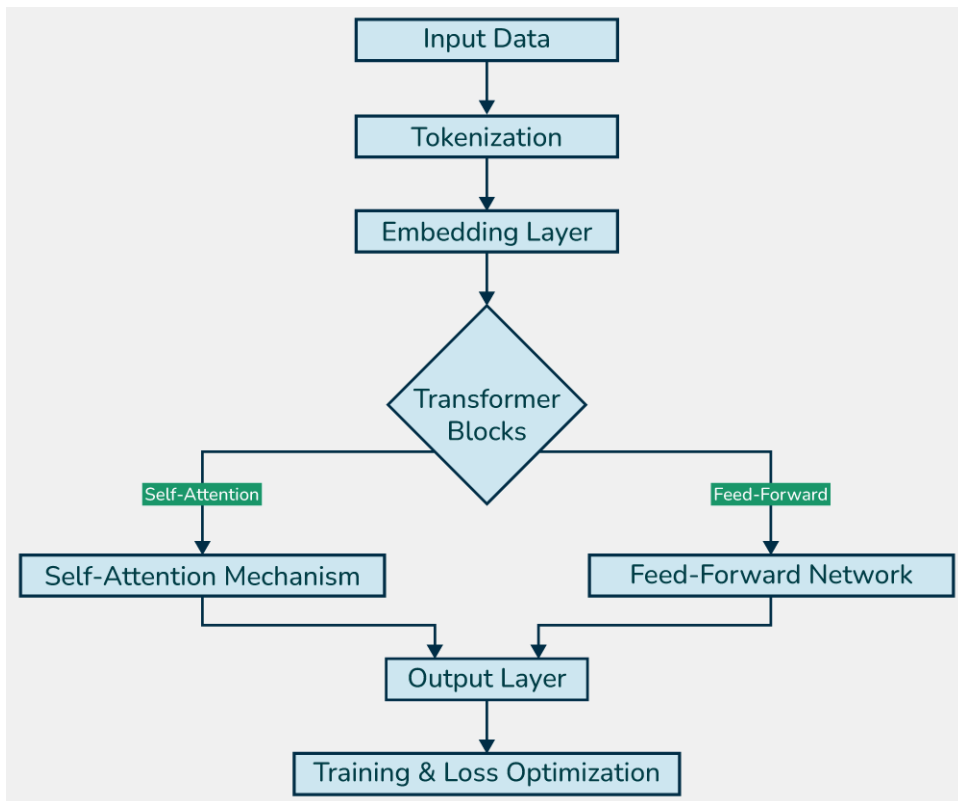
1.6 Arsitektur LLM

Large Language Models (LLM) dibangun di atas arsitektur yang kompleks, namun pada dasarnya mengikuti pola umum yang terdiri dari tahap masukan, representasi, pemrosesan melalui blok transformer, dan keluaran. Arsitektur ini dirancang untuk memungkinkan model memahami konteks bahasa dalam jumlah besar secara paralel dan efisien.

Tahap pertama dimulai dari input data, yaitu teks dalam bahasa alami yang dimasukkan ke dalam sistem. Teks ini tidak dapat diproses secara langsung oleh model, sehingga dilakukan tokenisasi, yaitu pemecahan teks ke dalam unit-unit kecil yang disebut token. Setelah itu, setiap token dipetakan menjadi embedding, yaitu representasi numerik berdimensi tinggi yang memuat makna semantik token tersebut. Embedding inilah yang menjadi titik awal agar teks manusia dapat dipahami dalam bentuk matematis oleh mesin.

Embedding yang telah terbentuk kemudian diproses dalam blok transformer, yang merupakan inti dari arsitektur LLM modern. Blok ini terdiri dari dua komponen utama, yaitu mekanisme self-attention dan feed-forward neural network. Mekanisme self-attention memungkinkan model untuk mengidentifikasi hubungan antar-token di seluruh bagian kalimat, bahkan jika token tersebut berjauhan. Dengan cara ini, model dapat memahami konteks secara menyeluruh dan menentukan kata mana yang paling relevan dalam suatu urutan. Sementara itu, jaringan feed-forward berfungsi mengolah representasi token setelah melalui self-attention, memperkaya pemahaman model terhadap struktur bahasa.

Hasil pemrosesan dari blok transformer kemudian dikirim ke output layer, yang bertugas menghasilkan prediksi berupa token berikutnya. Proses ini diulang secara berkelanjutan untuk menyusun teks yang lebih panjang. Agar model semakin akurat, dilakukan pula tahap training dan optimasi fungsi loss, di mana model belajar dengan menyesuaikan parameter berdasarkan perbedaan antara prediksi dan data yang sebenarnya. Secara visual, arsitektur LLM dapat digambarkan dalam diagram berikut.



Arsitektur LLM

Diagram di atas menunjukkan alur mulai dari masukan data, tokenisasi, embedding, hingga masuk ke blok transformer. Dari blok ini, aliran informasi bercabang ke mekanisme self-attention dan feed-forward network, yang hasilnya kemudian digabungkan kembali sebelum diteruskan ke lapisan keluaran dan proses pelatihan. Representasi ini membantu menjelaskan bagaimana LLM dapat memproses bahasa alami secara sistematis dengan memanfaatkan kekuatan attention mechanism.

1.7 Dataset dan Proses Pelatihan

Dataset merupakan fondasi utama dalam pengembangan Large Language Models (LLM). Tanpa data dalam jumlah besar dan beragam, model tidak akan mampu menangkap keragaman, kompleksitas, maupun nuansa bahasa alami. Proses pelatihan LLM melibatkan miliaran hingga triliunan token yang dikumpulkan dari berbagai sumber teks, sehingga model memperoleh

pemahaman luas tentang struktur bahasa, pengetahuan dunia, hingga pola interaksi manusia.

Sumber dataset yang digunakan untuk pelatihan LLM biasanya mencakup korpus teks berskala besar seperti Common Crawl, Wikipedia, buku digital, artikel berita, forum diskusi, media sosial, hingga kode pemrograman dari repositori publik. Setiap sumber memberikan kontribusi berbeda: Wikipedia menyediakan informasi faktual terstruktur, buku memberi kedalaman linguistik, sementara forum atau media sosial menambah variasi bahasa sehari-hari. Dengan kombinasi ini, model diharapkan mampu menggeneralisasi berbagai gaya bahasa, topik, dan konteks.

Namun, penggunaan dataset berskala masif ini juga menghadirkan tantangan. Pertama, data yang digunakan sering kali mengandung bias, baik dalam bentuk bias gender, etnis, maupun perspektif politik, yang secara tidak langsung dapat diwarisi model. Kedua, terdapat risiko masuknya konten berbahaya atau beracun (toxic content) seperti ujaran kebencian, informasi salah, atau materi eksplisit. Ketiga, masalah privasi juga menjadi sorotan, karena model dapat tanpa sengaja menyimpan informasi sensitif jika data pelatihan tidak disaring dengan benar. Oleh sebab itu, kurasi data menjadi tahap krusial yang melibatkan pembersihan (data cleaning), penyaringan konten, serta pengendalian kualitas sebelum digunakan untuk pelatihan.

Proses pelatihan LLM secara umum dibagi menjadi dua tahap utama, yaitu pretraining dan fine-tuning. Pada tahap pretraining, model dilatih menggunakan dataset umum dalam jumlah sangat besar tanpa fokus pada tugas tertentu. Tujuannya adalah agar model menguasai pola dasar bahasa, hubungan semantik antar kata, dan struktur kalimat. Misalnya, model belajar bahwa kata “matahari” sering berhubungan dengan “terbit” atau “cahaya,” meskipun tidak diberi instruksi eksplisit.

Setelah pretraining, model biasanya masih bersifat umum dan belum sepenuhnya sesuai dengan kebutuhan praktis. Oleh karena itu, dilakukan tahap fine-tuning. Fine-tuning menyesuaikan model dengan dataset yang lebih kecil dan spesifik untuk tugas tertentu. Ada berbagai pendekatan fine-tuning, mulai dari full fine-tuning di mana seluruh parameter model diperbarui, hingga parameter-efficient fine-tuning (PEFT) seperti LoRA (Low-Rank Adaptation), adapter layers, atau prompt-tuning, yang hanya memperbarui sebagian kecil parameter. Dengan cara ini, model dapat beradaptasi lebih cepat dengan kebutuhan domain tertentu, misalnya hukum, kesehatan, atau bisnis, tanpa biaya komputasi sebesar pretraining.

Selain itu, perkembangan terbaru juga memperkenalkan instruction tuning dan reinforcement learning from human feedback (RLHF) sebagai bagian dari proses pelatihan. Instruction tuning melibatkan pemberian contoh bagaimana model seharusnya merespons instruksi manusia, sehingga interaksi menjadi lebih intuitif. Sementara itu, RLHF menggunakan umpan balik dari penilai manusia untuk menyesuaikan keluaran model, dengan tujuan meningkatkan kesesuaian jawaban terhadap preferensi pengguna.

Tahap terakhir dalam pelatihan LLM adalah evaluasi dan optimasi loss. Model terus diperbaiki dengan menyesuaikan bobot parameternya agar perbedaan antara prediksi dan data referensi semakin kecil. Proses ini dilakukan secara iteratif menggunakan infrastruktur komputasi besar, seperti kluster GPU atau TPU, yang memungkinkan pembelajaran paralel dalam skala masif.

1.8 Parameter, Skala, dan Infrastruktur

Parameter merupakan inti dari sebuah Large Language Model (LLM). Dalam jaringan saraf tiruan, parameter adalah bobot numerik yang digunakan untuk menghitung keluaran berdasarkan masukan. Setiap parameter berfungsi sebagai pengatur hubungan antara neuron dalam lapisan jaringan, sehingga jumlah dan kualitas parameter sangat menentukan kemampuan model. Semakin banyak parameter yang dimiliki, semakin kompleks representasi yang dapat dipelajari, dan semakin kaya pula pola bahasa yang bisa ditangkap.

Skala parameter pada LLM berkembang secara dramatis dalam beberapa tahun terakhir. Model awal berbasis recurrent neural networks hanya memiliki jutaan parameter, sementara BERT yang diluncurkan pada 2018 telah mencapai ratusan juta parameter. GPT-2, yang diperkenalkan pada 2019, memiliki sekitar 1,5 miliar parameter, dan GPT-3 melompat jauh hingga 175 miliar parameter. Generasi terbaru, seperti GPT-4, Gemini, dan Claude, bahkan diperkirakan memiliki ratusan miliar hingga triliunan parameter. Lonjakan skala ini bukan sekadar angka, tetapi mencerminkan penerapan prinsip scaling law yang menunjukkan bahwa performa model cenderung meningkat seiring dengan bertambahnya jumlah parameter, volume data pelatihan, dan kapasitas komputasi.

Namun, besarnya jumlah parameter juga menghadirkan tantangan besar dalam hal infrastruktur. Melatih LLM membutuhkan sumber daya komputasi yang sangat besar, sering kali melibatkan ribuan unit pemrosesan grafis (GPU) atau tensor processing units (TPU) yang bekerja secara paralel. Infrastruktur ini didukung oleh sistem distributed training, di mana beban komputasi dibagi ke banyak perangkat keras sekaligus melalui teknik seperti data parallelism

(membagi batch data ke banyak mesin) dan model parallelism (membagi parameter model ke banyak perangkat). Tanpa teknik distribusi semacam ini, pelatihan LLM modern tidak akan mungkin dilakukan.

Selain itu, kebutuhan penyimpanan dan memori juga menjadi faktor krusial. Model dengan ratusan miliar parameter dapat memakan ruang puluhan hingga ratusan gigabyte hanya untuk bobotnya saja, belum termasuk data pelatihan dan cache selama proses optimasi. Hal ini memunculkan inovasi baru seperti mixed precision training, di mana perhitungan dilakukan dengan presisi lebih rendah (misalnya 16-bit alih-alih 32-bit) untuk menghemat memori dan mempercepat komputasi tanpa mengorbankan akurasi secara signifikan.

Dari sisi energi, pelatihan LLM berbiaya sangat tinggi. Satu kali pelatihan model besar dapat memakan listrik yang setara dengan konsumsi ribuan rumah tangga selama setahun. Inilah yang menimbulkan kekhawatiran terkait keberlanjutan (sustainability) dalam pengembangan AI. Sebagai respons, muncul inisiatif menuju Green AI, yaitu upaya mengembangkan metode pelatihan yang lebih efisien energi, seperti quantization, distillation, serta desain arsitektur yang lebih hemat komputasi.

Tidak kalah penting, infrastruktur LLM juga mencakup sistem perangkat lunak. Framework deep learning seperti TensorFlow, PyTorch, dan JAX menyediakan dasar pemrograman untuk membangun dan melatih model dalam skala besar. Sementara itu, manajemen kluster komputasi biasanya memanfaatkan platform komputasi awan (cloud computing) dari penyedia seperti Google Cloud, Amazon Web Services, dan Microsoft Azure, yang memungkinkan pelatihan dalam skala global.

1.9 Evaluasi dan Benchmarking

Evaluasi dan benchmarking merupakan tahap penting dalam siklus pengembangan Large Language Models (LLM). Sebuah model dengan miliaran parameter dan data pelatihan dalam skala raksasa tidak dapat langsung dinilai hanya dari proses pelatihannya; diperlukan ukuran yang objektif dan sistematis untuk mengukur kualitas keluaran yang dihasilkan. Evaluasi inilah yang memastikan apakah model benar-benar dapat memahami bahasa, memberikan jawaban yang relevan, serta berperilaku sesuai dengan kebutuhan pengguna.

Pada tahap awal, evaluasi model bahasa banyak mengandalkan metrik tradisional seperti perplexity. Perplexity mengukur seberapa baik model memprediksi distribusi probabilitas kata berikutnya dalam sebuah teks. Semakin rendah nilai perplexity, semakin baik kemampuan model dalam memperkirakan

urutan kata yang benar. Selain itu, metrik berbasis perbandingan teks seperti BLEU (Bilingual Evaluation Understudy), ROUGE (Recall-Oriented Understudy for Gisting Evaluation), dan METEOR banyak digunakan untuk menilai kualitas terjemahan mesin dan ringkasan teks. Meskipun berguna, metrik-metrik ini memiliki keterbatasan karena hanya menilai kesamaan permukaan antara keluaran model dan teks referensi, tanpa memperhatikan variasi semantik yang sah.

Seiring dengan berkembangnya LLM, muncul kebutuhan untuk menguji kemampuan yang lebih kompleks, seperti penalaran (reasoning), pengetahuan faktual, pemrograman, dan pemahaman lintas domain. Untuk itu, dikembangkan berbagai benchmark modern yang lebih luas cakupannya. Beberapa benchmark penting antara lain:

- GLUE (General Language Understanding Evaluation) dan SuperGLUE, yang mengukur performa model pada berbagai tugas pemahaman bahasa, seperti klasifikasi teks, inference, dan tanya-jawab.
- MMLU (Massive Multitask Language Understanding), yang menguji kemampuan model pada lebih dari 50 bidang pengetahuan, mulai dari sejarah, hukum, hingga sains. Benchmark ini sering dijadikan standar untuk mengukur kecerdasan umum model.
- BIG-bench (Beyond the Imitation Game Benchmark), yang dirancang untuk menilai kemampuan model pada ratusan tugas yang mencakup kreativitas, penalaran logis, serta pemahaman dunia nyata.
- HumanEval, yang digunakan untuk menilai kemampuan pemrograman model dengan menguji seberapa baik ia menulis kode yang benar berdasarkan instruksi teks.

Selain benchmark berbasis teks, model multimodal memerlukan pendekatan evaluasi yang lebih kompleks. Misalnya, image captioning dievaluasi dengan metrik seperti CIDEr dan SPICE, sementara sistem text-to-image membutuhkan penilaian manusia untuk menilai kesesuaian antara deskripsi teks dan kualitas gambar yang dihasilkan. Hal ini menunjukkan bahwa evaluasi LLM semakin multidimensi seiring dengan meluasnya fungsi model.

Namun, evaluasi otomatis memiliki keterbatasan. Model dapat memperoleh skor tinggi pada benchmark tertentu, tetapi tetap gagal memberikan jawaban yang berguna atau aman dalam praktik nyata. Karena itu, evaluasi berbasis manusia (human evaluation) tetap penting. Dalam metode ini, penilai manusia diminta menilai keluaran model berdasarkan aspek relevansi, koherensi,

keakuratan, kreativitas, hingga keamanan. Evaluasi manusia juga digunakan dalam proses reinforcement learning from human feedback (RLHF), di mana preferensi pengguna dijadikan acuan untuk meningkatkan perilaku model.

Aspek lain yang semakin mendapat perhatian adalah evaluasi terkait etika dan keamanan. Model tidak hanya diukur dari kemampuan kognitifnya, tetapi juga dari kemampuannya menghindari bias diskriminatif, menyaring konten berbahaya, serta memberikan jawaban yang sesuai norma sosial. Benchmark baru mulai dikembangkan untuk mengukur aspek-aspek ini, misalnya pengujian terhadap bias gender, fairness antar kelompok, atau potensi penyalahgunaan.

1.10 Contoh Penggunaan LLM

Large Language Models (LLM) seperti GPT, BERT, dan lainnya telah mengubah banyak aspek kehidupan dan industri. Berikut adalah contoh aplikasi nyata LLM yang sudah diterapkan secara luas:

1. Asisten Virtual dan Chatbot

- LLM digunakan untuk mengembangkan asisten virtual yang dapat memahami dan merespons pertanyaan manusia secara alami.
- Contoh: ChatGPT, Siri, Alexa, Google Assistant yang membantu menjawab pertanyaan, mengatur jadwal, atau memberikan rekomendasi.

2. Otomatisasi Penulisan dan Pembuatan Konten

- Membantu penulis membuat draft artikel, esai, laporan, dan konten kreatif lainnya.
- Digunakan untuk pembuatan konten marketing, deskripsi produk, atau bahkan novel dan puisi.

3. Terjemahan Bahasa Otomatis

- LLM memungkinkan terjemahan bahasa secara real-time dengan kualitas yang semakin baik.
- Aplikasi seperti Google Translate dan DeepL memanfaatkan teknologi ini untuk komunikasi lintas bahasa.

4. Analisis Sentimen dan Pemahaman Teks

- Memproses ulasan pelanggan, opini di media sosial, atau feedback untuk membantu perusahaan memahami sikap dan kebutuhan pasar.
- Digunakan dalam riset pasar, layanan pelanggan, dan monitoring merek.

5. Penyaringan dan Klasifikasi Email

- Membantu mengatur email masuk ke dalam kategori seperti penting, promosi, spam, dll.
- Memudahkan pengguna dalam mengelola komunikasi yang banyak.

6. Pembelajaran dan Pendidikan Adaptif

- LLM dapat menjadi tutor virtual yang memberikan penjelasan, latihan soal, dan feedback secara personal.
- Diterapkan dalam aplikasi pembelajaran online dan edukasi AI.

7. Pengembangan Software dan Dokumentasi

- Membantu programmer dengan memberikan contoh kode, menjelaskan fungsi, atau menulis dokumentasi teknis.
- Contoh: GitHub Copilot yang membantu dalam penulisan kode.

BAB II

RAGAM MODEL LLM POPULER

Perkembangan Large Language Models (LLM) telah menghasilkan berbagai varian yang dirancang oleh perusahaan maupun komunitas berbeda. Setiap model memiliki karakteristik, keunggulan, dan keterbatasan tersendiri. Sebagian model dikembangkan dengan fokus pada percakapan interaktif, sementara yang lain lebih dioptimalkan untuk pemrosesan data skala besar, integrasi dengan ekosistem tertentu, atau fleksibilitas dalam penyesuaian penggunaan.

Pemahaman mengenai keragaman model ini sangat penting agar pemilihan dan penerapan LLM sesuai dengan tujuan yang diharapkan. Dalam praktiknya, model tertentu lebih tepat digunakan untuk penulisan kreatif, model lain unggul dalam analisis data atau penyusunan kode, sementara sebagian lagi difokuskan pada integrasi sistem dan layanan digital.

2.1 ChatGPT

ChatGPT merupakan produk utama dari OpenAI yang berbasis pada model GPT (Generative Pre-trained Transformer), sebuah arsitektur neural network yang dilatih menggunakan miliaran token dari berbagai sumber internet. Sejak versi pertamanya, GPT telah menunjukkan kemajuan pesat, baik dalam hal kemampuan berbahasa maupun penerapan praktisnya.



OpenAI Perusahaan Pengembang ChatGPT

Ketiga model GPT yang telah disebutkan (GPT-3.5, GPT-4, dan GPT-4o) merupakan evolusi dari pendekatan transformer-based language model yang dikembangkan oleh OpenAI. Masing-masing model dirancang untuk mengatasi keterbatasan versi sebelumnya, baik dari sisi pemahaman konteks, efisiensi pengolahan data, hingga integrasi multimodal.

GPT-3.5

Diluncurkan pada tahun 2022, GPT-3.5 merupakan versi transisi yang menggabungkan kekuatan dari GPT-3 dengan peningkatan akurasi, efisiensi, dan kecepatan respon. Model ini banyak digunakan dalam tugas-tugas harian seperti penulisan email, penyusunan ringkasan, hingga tanya jawab berbasis teks secara ringan (OpenAI, 2023).

GPT-3.5 hadir sebagai jembatan antara GPT-3 dan GPT-4, dengan fokus utama pada peningkatan stabilitas, efisiensi pemrosesan, dan kecepatan respon. Meskipun secara arsitektur masih berada dalam garis besar GPT-3, versi ini memperlihatkan optimalisasi besar dalam hal fine-tuning, yang membuat interaksinya terasa lebih natural dan relevan. GPT-3.5 mampu menjawab pertanyaan dengan struktur yang lebih jelas, menghindari repetisi, dan memiliki tingkat koherensi yang lebih baik dalam percakapan panjang.

Penggunaannya menjadi populer dalam banyak aplikasi praktis, mulai dari chatbot layanan pelanggan, penulisan otomatis konten blog ringan, hingga asisten produktivitas pribadi seperti peringkasan email atau dokumen. Keunggulan GPT-3.5 adalah pada kemampuannya mengimbangi kebutuhan performa dan biaya dapat memberikan hasil yang memuaskan dengan beban komputasi yang relatif lebih ringan dibanding GPT-4. Namun, keterbatasannya mulai terasa ketika dihadapkan pada tugas yang memerlukan pemahaman logika tingkat tinggi atau pertanyaan kompleks, di mana model ini kadang memberikan jawaban yang terlalu umum atau bahkan tidak tepat.

GPT-4

GPT-4 adalah model yang jauh lebih kuat dibandingkan pendahulunya. OpenAI (2023) menyatakan bahwa GPT-4 memiliki kemampuan penalaran yang lebih baik, memahami konteks lebih dalam, serta lebih tahan terhadap prompt ambigu atau manipulatif. Dalam studi benchmark seperti MMLU (Massive Multitask Language Understanding), GPT-4 mampu melampaui performa manusia pada banyak kategori akademik (OpenAI, 2023).

Diluncurkan pada tahun 2023, GPT-4 merupakan lompatan besar dalam perkembangan model bahasa AI. Model ini dikembangkan dengan ukuran parameter yang lebih besar dan pendekatan pelatihan yang lebih canggih, termasuk strategi reinforcement learning from human feedback (RLHF) yang ditingkatkan. GPT-4 menampilkan kemampuan penalaran logis, pemahaman konteks yang mendalam, dan ketahanan terhadap prompt yang ambigu—hal yang sering menjadi tantangan pada model sebelumnya.

Salah satu pencapaian signifikan dari GPT-4 adalah kinerjanya dalam uji Massive Multitask Language Understanding (MMLU), di mana model ini berhasil mengungguli skor rata-rata manusia dalam menjawab soal dari berbagai mata pelajaran, mulai dari matematika, biologi, hukum, hingga sejarah. GPT-4 juga memperkenalkan fitur pengendalian gaya dan nada tulisan, membuatnya ideal digunakan dalam penulisan konten profesional, penyusunan laporan kompleks, simulasi akademik, hingga pembuatan kode pemrograman.

GPT-4o

GPT-4o, yang dirilis pada Mei 2024, adalah model multimodal pertama dari OpenAI yang mendukung input dan output dalam bentuk teks, suara, dan gambar secara real-time. Model ini dirancang agar responsif, ekspresif, dan lebih "hidup" dalam interaksi percakapan, sekaligus lebih efisien secara komputasional (OpenAI, 2024). Kemampuan ini membuat GPT-4o sangat ideal untuk aplikasi virtual assistant, customer service berbasis AI, serta alat bantu pembelajaran interaktif. GPT-4o atau “Omni”, yang dirilis pada Mei 2024, menjadi tonggak revolusioner dalam dunia AI karena untuk pertama kalinya OpenAI menghadirkan model yang benar-benar multimodal secara real-time. Tidak hanya teks, GPT-4o mampu menerima dan merespons input dalam bentuk suara, gambar, bahkan ekspresi wajah (dalam beberapa integrasi), serta memberikan output dengan intonasi suara yang ekspresif dan natural. Kemampuan ini menempatkannya sebagai AI yang “hidup” dalam interaksi, tidak sekadar menjawab, tetapi juga merasakan nuansa komunikasi manusia. GPT-4o dirancang untuk menjadi responsif dan ringan dari sisi komputasi, menjadikannya ideal untuk integrasi pada perangkat mobile, aplikasi customer service, virtual assistant interaktif, dan ruang kelas digital. Dengan kecepatan respons setara real-time dan kemampuan interpretasi visual (misalnya membaca grafik, dokumen, atau menilai ekspresi gambar), GPT-4o membuka potensi baru dalam pembelajaran imersif, pelayanan kesehatan digital, serta sistem navigasi dan bantuan pengguna disabilitas.

GPT-5

Perkembangan kecerdasan buatan generatif mencapai tonggak baru dengan hadirnya GPT-5, model yang diperkenalkan OpenAI pada Agustus 2025. GPT-5 diposisikan sebagai penerus GPT-4o dengan misi menyatukan kecepatan respons, kedalaman penalaran, serta kemampuan multimodal dalam satu sistem terpadu. Dengan model ini, percakapan tidak lagi sebatas tanya jawab singkat, tetapi lebih menyerupai dialog yang mampu menyesuaikan kedalaman jawaban dengan kebutuhan pengguna.

Arsitektur GPT-5 dilengkapi dengan mekanisme router dinamis yang secara otomatis memilih gaya respons: cepat dan ringkas untuk pertanyaan sehari-hari, atau analitis dan mendalam untuk masalah yang kompleks. Hal ini membuat interaksi menjadi lebih adaptif tanpa memerlukan pengaturan manual. Model ini juga hadir dalam beberapa varian, mulai dari standar, mini, nano, hingga versi “thinking” yang ditujukan untuk kasus yang membutuhkan penalaran lebih mendalam. Kemampuan memori diperluas dengan jendela konteks berukuran sangat besar, sehingga model mampu membaca dan memahami dokumen panjang maupun percakapan berlapis tanpa kehilangan konsistensi.

Keunggulan GPT-5 tampak pada beragam domain. Dalam matematika, penulisan, pengkodean, analisis hukum, maupun riset medis, model ini menghasilkan jawaban yang lebih akurat, logis, dan konsisten dibanding pendahulunya. Uji akademik menunjukkan bahwa GPT-5 dapat menjawab soal klinis dengan tingkat akurasi mendekati, bahkan dalam beberapa kasus melampaui, kinerja profesional manusia. Model ini juga lebih tahan terhadap kesalahan informasi atau halusinasi. OpenAI menambahkan mekanisme penyelesaian aman yang membuat GPT-5 lebih berhati-hati dalam menangani pertanyaan sensitif, sehingga hasilnya lebih etis dan dapat dipertanggungjawabkan.

Integrasi dengan berbagai alat produktivitas menjadi aspek lain yang memperkuat kegunaannya. GPT-5 terhubung dengan ekosistem Copilot di Microsoft Office, email, dan kalender, menjadikannya asisten kerja yang praktis. Pengguna juga dapat membentuk persona khusus seperti pengajar, analis, atau pendengar. Dengan cara ini, jawaban yang dihasilkan tidak hanya relevan dari sisi isi, tetapi juga konsisten dalam gaya komunikasi.

Meski banyak dipuji, GPT-5 juga menuai kritik. Beberapa pengamat menilai peningkatan yang ditawarkan bersifat evolusi bertahap, bukan lompatan revolusioner. Router otomatis yang menjadi ciri utama dinilai belum sepenuhnya transparan, sehingga sebagian pengguna merasa kurang memiliki kendali terhadap jenis jawaban yang dipilih model.

Terlepas dari kritik tersebut, GPT-5 menandai transisi penting menuju era kecerdasan buatan yang lebih bersifat agen. Model ini tidak sekadar menjawab pertanyaan, tetapi juga mampu menyesuaikan kedalaman analisis, menjaga konsistensi etika, dan berperan sebagai mitra dalam alur kerja manusia. Bila GPT-4o dapat dipandang sebagai pintu masuk menuju multimodalitas yang lebih alami, maka GPT-5 memperluas cakrawala dengan penalaran yang lebih matang sekaligus penerapan praktis yang lebih luas.

Evolusi dan Perbandingan Fitur GPT-3.5, GPT-4, dan GPT-4o (Omni)

Fitur Utama	GPT-3.5 (2022)	GPT-4 (2023)	GPT-4o (2024)
Tipe Model	Monomodal (teks saja)	Monomodal (teks saja), tetapi lebih kuat dalam konteks panjang	Multimodal (teks, suara, gambar – input & output real-time)
Kemampuan Bahasa	Baik untuk teks ringan dan tugas sehari-hari	Sangat baik, dengan pemahaman konteks dalam, mampu menyusun argumen kompleks	Sangat ekspresif, mendekati komunikasi alami manusia
Penalaran & Logika	Terbatas pada tugas sederhana	Tingkat tinggi, unggul dalam benchmark akademik (MMLU, bar exam, dsb.)	Setara GPT-4, dengan respons lebih cepat dan reaktif dalam konteks visual dan suara
Kecepatan Respons	Cepat	Lebih lambat karena kompleksitas tinggi	Sangat cepat, hampir real-time
Kemampuan Visual	Tidak tersedia	Terbatas pada versi khusus (GPT-4-vision via API tertentu)	Mampu membaca gambar, grafik, tulisan tangan, ekspresi wajah
Kemampuan Suara	Tidak tersedia	Tidak tersedia	Input dan output suara dengan ekspresi, nada, dan intonasi
Pemrograman & Kode	Dasar (bisa menulis skrip umum)	Sangat kuat (bisa menyusun kode kompleks dengan penjelasan)	Sangat baik, sama seperti GPT-4
Efisiensi Komputasi	Ringan, cocok untuk aplikasi skala kecil	Berat, memerlukan resource besar	Efisien, lebih ringan daripada GPT-4, cocok untuk perangkat edge/mobile

Kasus Penggunaan Ideal	Asisten penulisan, chatbot ringan, personal productivity	Analisis data, penulisan profesional, asisten akademik	Customer service interaktif, asisten AI multimodal, pembelajaran berbasis suara & visual
Tantangan/ Keterbatasan	Rentan error pada prompt kompleks, pemahaman konteks terbatas	Lambat pada perangkat rendah resource, biaya tinggi	Etika privasi multimodal, kebutuhan pengawasan data visual & suara lebih ketat

Secara garis besar ChatGPT adalah representasi konkret dari kemajuan teknologi kecerdasan buatan yang semakin menyatu dalam kehidupan sehari-hari. Dengan mengadopsi model Generative Pre-trained Transformer, ChatGPT mampu memahami, mengolah, dan menghasilkan teks secara kontekstual menyerupai cara manusia berpikir dan berkomunikasi. Teknologi ini tidak hanya berhenti pada pengolahan bahasa, tetapi juga meluas ke multimodalitas melalui GPT-4o, memungkinkan interaksi manusia-mesin menjadi lebih natural dan dinamis.

Penggunaan ChatGPT sangat beragam dan terus berkembang seiring peningkatan kemampuannya. Pada level personal, pengguna memanfaatkan ChatGPT untuk menulis artikel, menyusun skrip, menerjemahkan dokumen, atau sekadar berdiskusi untuk mendapatkan ide kreatif. Di sektor profesional dan industri, ChatGPT digunakan sebagai asisten virtual, penyusun laporan otomatis, alat bantu edukasi, hingga menjadi basis layanan pelanggan (customer service) berbasis chatbot. Bahkan di bidang akademik, model ini dimanfaatkan untuk membantu analisis literatur, ringkasan jurnal, dan simulasi diskusi. Kelebihan utama ChatGPT terletak pada kemampuannya dalam memahami konteks secara mendalam, fleksibilitas dalam berbagai topik, dan kecepatan respons yang tinggi. Khususnya GPT-4 dan GPT-4o menunjukkan performa luar biasa dalam pemrosesan bahasa alami (NLP), bahkan mampu menjawab pertanyaan berbasis logika, menyusun argumen, dan merespons pertanyaan teknis secara presisi. GPT-4o juga membuka era baru dalam AI dengan memperkenalkan interaksi multimodal secara simultan, menjadikannya lebih human-like dan adaptif terhadap berbagai situasi percakapan. Namun, perlu diketahui juga mengenai kekurangan ChatGPT yang tetap perlu dicermati. Model ini masih dapat menghasilkan informasi yang tidak akurat atau bersifat "halusinasi", terutama jika diberikan prompt yang ambigu atau jika datanya berada di luar cakupan pelatihan. Selain itu, kendala etis dan bias algoritma juga menjadi tantangan serius, mengingat model dilatih dari data internet yang tidak selalu netral. Aspek

privasi, keamanan data, serta potensi penyalahgunaan AI juga menjadi perhatian penting di tengah adopsi yang kian meluas. ChatGPT sebagai salah satu pencapaian paling revolusioner dalam bidang teknologi informasi. Kecerdasannya yang berkembang pesat membuka banyak peluang, tidak hanya untuk meningkatkan efisiensi kerja, tetapi juga dalam mendemokratisasi akses terhadap pengetahuan. Meski begitu, pengguna tetap perlu bersikap kritis dan bijak, menyadari bahwa AI bukanlah pengganti manusia, melainkan alat bantu yang perlu diarahkan dengan tepat. Dengan pendekatan yang bertanggung jawab, ChatGPT dan model-model sejenisnya dapat menjadi mitra teknologi yang membawa manfaat luas bagi masyarakat.

2.2 Gemini

Gemini merupakan LLM besutan Google DeepMind yang menggantikan Bard sebagai platform AI utama Google. Gemini didesain dengan kemampuan multimodal, memungkinkan pemrosesan teks, gambar, audio, hingga kode program dalam satu kerangka kerja. Salah satu kekuatan utama Gemini adalah integrasinya yang mendalam dengan ekosistem Google seperti Gmail, Docs, Drive, dan YouTube. Gemini diluncurkan dalam berbagai versi seperti Gemini 1.5 Pro yang mampu memproses konteks panjang hingga 1 juta token (Google DeepMind, 2024). Integrasi ini memungkinkan pengguna mengakses informasi personal secara kontekstual dengan tetap menjaga privasi melalui sistem perizinan granular.

Gemini unggul dalam summarization, reasoning numerik, dan analisis dokumen besar. Kehadiran Gemini menandai langkah strategis Google dalam menghadirkan kecerdasan buatan yang tidak hanya kuat secara teknis, tetapi juga terintegrasi secara alami dalam aktivitas digital sehari-hari. Berbeda dengan pendekatan LLM lainnya yang lebih bersifat generik, Gemini dirancang untuk beroperasi langsung di dalam layanan Google yang sudah digunakan oleh miliaran pengguna di seluruh dunia. Melalui integrasi dengan Gmail, Google Docs, Sheets, hingga YouTube, Gemini dapat membantu pengguna menyusun email, merangkum dokumen, mencari file, menyatakan isi video, hingga menulis kode secara kontekstual, tanpa perlu keluar dari ekosistem Google.

Salah satu terobosan besar dari Gemini, khususnya versi Gemini 1.5 Pro, adalah kemampuan memproses konteks sangat panjang hingga 1 juta token, yang menjadikannya sangat andal dalam memahami dokumen kompleks seperti laporan keuangan, makalah penelitian, atau transkrip panjang. Kemampuan ini juga memungkinkannya melakukan pelacakan konteks lintas dokumen dan file, sehingga cocok digunakan dalam kegiatan bisnis, akademik, hingga manajemen

proyek skala besar. Fitur ini memberi nilai lebih dibanding banyak model LLM lainnya yang masih terbatas dalam kapasitas konteks input. Dalam hal kemampuan multimodal, Gemini tidak hanya memahami teks dan kode, tetapi juga dapat memproses gambar dan audio, memungkinkan penggunaan yang lebih luas seperti menganalisis grafik, mendeteksi pola visual, atau menyarikan wawancara audio. Fitur ini sangat berguna untuk content creator, peneliti, jurnalis, dan pengguna profesional lainnya yang sering bekerja dengan data lintas format.

Kelebihan Gemini juga terletak pada pengelolaan privasi dan keamanan data, yang didukung oleh sistem perizinan granular milik Google. Pengguna memiliki kontrol atas data mana yang dapat diakses AI dan dalam konteks apa, sehingga tetap menjaga kepercayaan pengguna dalam lingkungan kerja berbasis cloud. Namun, seperti teknologi LLM lainnya, tantangan utama Gemini terletak pada kebutuhan resource komputasi tinggi dan kebutuhan koneksi internet yang stabil untuk menjalankan fitur-fitur canggihnya secara optimal. Gemini sebagai jawaban Google terhadap dominasi OpenAI dan Microsoft dalam dunia AI generatif, dengan keunggulan kompetitif yang terletak pada sinergi mendalam dengan layanan yang sudah akrab bagi pengguna. Potensinya besar untuk menjadi AI asisten personal dan profesional yang benar-benar terintegrasi, bukan hanya sebagai alat bantu percakapan, tetapi sebagai co-pilot digital yang memahami konteks pekerjaan secara menyeluruh.

2.3 Claude

Claude adalah model LLM dari Anthropic, perusahaan yang didirikan oleh mantan peneliti OpenAI. Keunikan Claude terletak pada pendekatan Constitutional AI, yaitu metode pelatihan yang membatasi dan membimbing respons model agar tetap dalam kerangka nilai dan etika tertentu. Claude 2 dan Claude 3 memiliki kemampuan untuk memproses dokumen sangat panjang, hingga 200.000 token, menjadikannya unggul untuk aplikasi legal, edukasi, atau korporasi yang membutuhkan analisis teks masif (Anthropic, 2024). Selain itu, respon Claude dikenal lebih halus, tidak konfrontatif, dan mempertimbangkan keamanan informasi. Pendekatan Constitutional AI yang dikembangkan oleh Anthropic pada Claude menjadi salah satu inovasi penting dalam dunia model bahasa besar (LLM).

Berbeda dengan metode pelatihan berbasis umpan balik manusia (reinforcement learning from human feedback atau RLHF) yang lazim digunakan oleh OpenAI dan lainnya, Constitutional AI bekerja dengan memberikan model seperangkat prinsip atau “konstitusi” sebagai acuan moral dan etika.

Prinsip-prinsip ini digunakan untuk membimbing perilaku model secara konsisten tanpa mengandalkan pengawasan manusia secara eksplisit dalam setiap respons. Hasilnya, Claude tidak hanya tampil sebagai AI yang cerdas, tetapi juga lebih terkontrol, empatik, dan minim bias, dengan kemampuan menjawab pertanyaan secara sopan, bertanggung jawab, dan menghindari topik berbahaya atau manipulatif. Dalam versi lanjutannya seperti Claude 2 dan Claude 3, model ini menunjukkan peningkatan luar biasa dalam kemampuan memahami konteks panjang, dengan dukungan hingga 200.000 token input—salah satu yang terbesar di antara model AI komersial saat ini. Kemampuan ini menjadikan Claude sangat ideal untuk membaca dan menganalisis dokumen hukum, laporan korporasi, riset akademik, dan data tekstual kompleks lainnya secara menyeluruh tanpa kehilangan konteks awal. Claude juga unggul dalam mencari informasi dari berbagai bagian dokumen panjang dan menyusunnya dalam bentuk yang ringkas, jelas, dan logis. Kelebihan Claude lainnya adalah gaya komunikasi yang lebih tenang dan tidak konfrontatif. Model ini dirancang untuk memberikan respons yang mengutamakan keamanan dan kenyamanan pengguna, termasuk dalam diskusi tentang isu sensitif. Bahkan dalam situasi yang memancing emosi atau argumen keras, Claude cenderung menjawab dengan nada netral dan penuh pertimbangan. Hal ini sangat penting untuk digunakan dalam layanan publik, pendidikan, dan asisten profesional, di mana nada komunikasi dan sensitivitas sangat menentukan pengalaman pengguna.

Namun, meskipun memiliki banyak keunggulan, Claude juga memiliki keterbatasan. Dalam beberapa kasus, model ini bisa terlalu berhati-hati atau menghindari menjawab secara langsung, terutama pada pertanyaan yang menyangkut opini, isu politis, atau konflik etis. Pendekatan yang terlalu protektif ini bisa menjadi hambatan dalam penggunaan tertentu yang membutuhkan eksplorasi lebih luas atau kreativitas yang tinggi. Selain itu, meskipun telah memiliki kemampuan multimodal, integrasi Claude dengan ekosistem teknologi lain masih belum komprehensif pesaing seperti Gemini (Google) atau GPT-4o (OpenAI).

2.4 Model Alternatif dan Open-Source

Selain model komersial, banyak model LLM yang dikembangkan secara open-source atau oleh perusahaan alternatif. Model-model ini sering kali digunakan untuk eksperimen, pengembangan lokal, atau integrasi sistem yang membutuhkan fleksibilitas tinggi. Kemunculan berbagai model alternatif dan open-source dalam ranah LLM (Large Language Models) membuka era baru dalam demokratisasi kecerdasan buatan.

Mistral

Mistral AI, startup asal Prancis, meluncurkan model LLM ringan dan efisien seperti Mistral 7B dan Mixtral. Model ini sepenuhnya open-source dan banyak digunakan dalam aplikasi edge AI dan deployment lokal (Mistral AI, 2023). Mistral telah menarik perhatian karena keberhasilannya mengembangkan model ringan dan efisien seperti Mistral 7B dan Mixtral yang berbasis Mixture of Experts (MoE). Model ini mampu memberikan performa tinggi pada perangkat dengan resource terbatas, menjadikannya sangat cocok untuk aplikasi edge AI, seperti chatbot offline, IoT, atau robotika. Karena bersifat sepenuhnya open-source dan bebas lisensi terbatas, Mistral menjadi pilihan utama bagi pengembang yang menginginkan kendali penuh atas model dan data mereka. Selain itu, komunitas yang aktif di sekitar Mistral memungkinkan iterasi dan peningkatan model secara berkelanjutan.

LLaMA (Meta AI)

Meta mengembangkan LLaMA (Large Language Model Meta AI) sebagai model riset open-source. Versi terbarunya, LLaMA 3, dirancang dengan arsitektur open-weight yang memungkinkan para peneliti dan developer untuk melakukan fine-tuning dan analisis (Meta AI, 2024). LLaMA (Large Language Model Meta AI) dikembangkan oleh Meta sebagai proyek riset terbuka yang sangat berdampak. Mulai dari LLaMA 1 hingga versi terbaru LLaMA 3, model ini hadir dengan pendekatan open-weights, memungkinkan siapa saja melakukan fine-tuning, benchmarking, dan pengujian langsung. LLaMA telah menjadi fondasi dari berbagai proyek LLM independen seperti Alpaca, Vicuna, dan OpenChat, menjadikannya pusat gravitasi dalam ekosistem LLM open-source. Dengan kapabilitas yang mendekati model-model tertutup, LLaMA mendorong pertumbuhan pesat dalam pengembangan AI lokal dan penelitian akademik.

Command R+ (Cohere)

Command R+ adalah LLM dari Cohere yang dikembangkan dengan pendekatan retrieval-augmented generation (RAG), yaitu menggabungkan kemampuan model dengan pencarian sumber eksternal secara real-time. Ini menjadikannya ideal untuk menjawab pertanyaan berbasis data terkini (Cohere, 2024). Model Command R+ dari Cohere mengusung pendekatan yang berbeda melalui integrasi retrieval-augmented generation (RAG), di mana model tidak

hanya mengandalkan memori internal, tetapi juga dapat mengakses dan menggabungkan informasi dari sumber eksternal secara real-time. Hal ini membuat Command R+ sangat efektif dalam menjawab pertanyaan berbasis data terkini, seperti berita harian, statistik waktu nyata, atau sumber dokumentasi teknis yang sering berubah. Kemampuan ini menjadi keunggulan tersendiri dalam aplikasi enterprise yang membutuhkan factual consistency dan keakuratan informasi yang tinggi.

Grok (xAI)

Grok adalah model dari xAI, perusahaan yang didirikan oleh Elon Musk. Model ini terintegrasi dengan platform X (sebelumnya Twitter), dan difokuskan pada pemrosesan data sosial, tren daring, serta pemantauan isu-isu terkini secara cepat (xAI, 2024). Grok dari xAI, startup milik Elon Musk, tampil sebagai model yang sangat kontekstual terhadap dunia sosial dan informasi daring. Dengan integrasi penuh ke dalam platform X (dulu Twitter), Grok dirancang untuk memahami percakapan daring, tren media sosial, dan dinamika opini publik secara cepat. Fokus ini menjadikan Grok relevan untuk aplikasi pemantauan isu, analisis sentimen, serta interaksi real-time dengan basis pengguna besar dalam ekosistem X. Meskipun masih berada dalam tahap pengembangan dan eksklusif pada platform tertentu, Grok menampilkan arah baru pengembangan AI yang terhubung erat dengan real-time data stream.

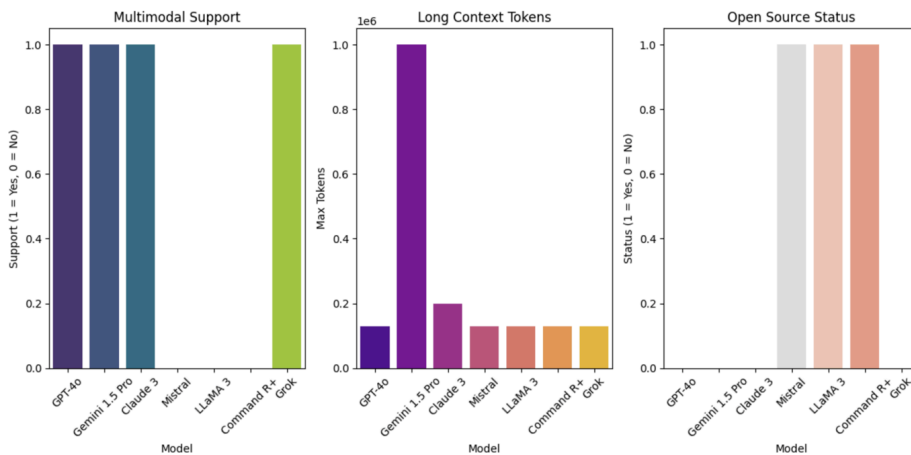
Secara keseluruhan, model-model alternatif ini memainkan peran penting dalam membentuk ekosistem AI yang lebih terbuka, terdesentralisasi, dan dapat disesuaikan, terutama bagi pengguna yang menginginkan kontrol lebih besar atas model dan data. Meskipun secara kapabilitas mereka masih bersaing ketat dengan model-model tertutup kelas atas, keunggulan mereka terletak pada keterbukaan, fleksibilitas, dan adaptabilitas untuk berbagai konteks pengguna. Dari sudut pandang pengembangan jangka panjang, model open-source seperti Mistral dan LLaMA memiliki potensi besar untuk mendorong inovasi lokal dan memperluas inklusivitas teknologi AI secara global.

2.5 Perbandingan Karakteristik dan Kecocokan Penggunaan

Tabel berikut menyajikan perbandingan berbagai model LLM berdasarkan karakteristik utama, keunggulan, dan konteks penggunaannya:

Model	Keunggulan Utama	Cocok untuk	Multimodal	Konteks Panjang	Open Source
GPT-4o	Respons cepat, reasoning kuat, input suara/gambar	Profesional, kreator konten	✓	Sedang (~128k token)	✗
Gemini 1.5 Pro	Terintegrasi Google Workspace, konteks sangat panjang	Pengguna Google, produktivitas	✓	✓ (1 juta token)	✗
Claude 3	Konstitusional, aman, stabil dalam konteks besar	Legal, edukasi, etika AI	✓	✓ (~200k token)	✗
Mistral	Ringan, efisien, open source	Deployment lokal, edge AI	✗	Sedang	✓
LLaMA 3	Riset, fleksibel, banyak dukungan komunitas	Akademisi, peneliti AI	✗	Bervariasi	✓
Command R+	Penguatan melalui pencarian eksternal	Pencarian jawaban berbasis data	✗	Sedang	✓
Grok	Aware terhadap tren sosial, data media X	Sosial media, analisis isu	✓	Sedang	✗

Dari tabel di atas, kita dapat melihat bahwa pemilihan model LLM sangat bergantung pada konteks kebutuhan, infrastruktur teknologi yang dimiliki, serta tingkat fleksibilitas yang diinginkan. Setiap model memiliki karakteristik khas yang membuatnya unggul dalam skenario tertentu, namun belum tentu cocok digunakan secara universal. Dapat diketahui bahwa tidak ada satu model yang unggul di semua aspek. Keputusan pemilihan model harus mempertimbangkan tujuan penggunaan, batasan infrastruktur, kebutuhan privasi, serta pendekatan integrasi. Dalam dunia AI yang semakin modular dan terfragmentasi, pemahaman akan karakteristik tiap model seperti ini menjadi sangat penting agar teknologi yang digunakan benar-benar optimal dan sesuai konteks.



Grafik diatas menyajikan perbandingan fitur utama dari beberapa model Large Language Models (LLM) berdasarkan tiga aspek: dukungan terhadap input multimodal, kapasitas konteks panjang, dan status open source. Pada grafik pertama, fitur dukungan multimodal dianalisis melalui indikator biner (1 = mendukung, 0 = tidak mendukung). Model seperti GPT-4o, Gemini 1.5 Pro, Claude 3, dan Grok menunjukkan kemampuan untuk memproses input multimodal, seperti teks, gambar, atau suara, yang menjadikannya unggul dalam aplikasi lintas media. Sebaliknya, model Mistral, LLaMA 3, dan Command R+ tidak mendukung multimodal, yang membatasi penggunaannya pada data berbasis teks saja. Hal ini menunjukkan bahwa dukungan multimodal lebih umum ditemukan pada model proprietary yang dirancang untuk kebutuhan komersial.

Grafik kedua membahas kapasitas konteks panjang, yang diukur berdasarkan jumlah token maksimal yang dapat diproses dalam satu konteks. Model Gemini 1.5 Pro menunjukkan kapasitas tertinggi dengan kemampuan memproses hingga 1 juta token, menjadikannya sangat cocok untuk aplikasi yang memerlukan analisis dokumen besar, seperti penelitian ilmiah atau dokumen hukum. Claude 3 memiliki kapasitas yang signifikan, yaitu 200.000 token, sementara model lainnya, termasuk GPT-4o, Mistral, LLaMA 3, Command R+, dan Grok, memiliki kapasitas yang lebih kecil, yaitu sekitar 128.000 token. Kapasitas konteks panjang merupakan parameter teknis yang sangat penting dalam menentukan kemampuan model untuk memahami dan memproses informasi dalam dokumen yang kompleks atau panjang.

Grafik ketiga membahas status open source, yang juga diukur dengan indikator biner (1 = open source, 0 = proprietary). Model Mistral, LLaMA 3, dan Command R+ adalah model open source, yang memberikan akses yang lebih luas kepada pengguna untuk mempelajari, memodifikasi, dan mengintegrasikan model ke dalam aplikasi mereka. Sebaliknya, GPT-4o, Gemini 1.5 Pro, dan Claude 3 adalah model proprietary, yang berarti mereka dimiliki dan dikelola oleh perusahaan komersial.

Command R+ bersifat open source, memberikan keuntungan dalam hal transparansi, kolaborasi komunitas, dan fleksibilitas untuk penelitian serta pengembangan. Sebaliknya, model GPT-4o, Gemini 1.5 Pro, Claude 3, dan Grok bersifat proprietary, yang biasanya ditujukan untuk aplikasi komersial dengan akses terbatas. Status open source menjadi penting dalam konteks akademik dan penelitian, karena memungkinkan pengguna untuk memodifikasi model sesuai kebutuhan spesifik mereka. Secara keseluruhan, grafik ini memberikan gambaran komprehensif tentang fitur-fitur utama dari berbagai LLM. Model seperti Gemini 1.5 Pro unggul dalam kapasitas konteks panjang, sementara model seperti Mistral dan LLaMA 3 menawarkan fleksibilitas melalui status open source. Perbandingan ini dapat digunakan untuk menentukan pilihan model berdasarkan kebutuhan spesifik, seperti dukungan multimodal, analisis dokumen panjang, atau pengembangan berbasis komunitas. Analisis ini juga menunjukkan bagaimana fitur teknis dan status kepemilikan model dapat memengaruhi penerapan teknologi dalam berbagai domain, baik komersial maupun akademik.

Dapat diketahui bahwa tidak ada satu model yang unggul di semua aspek. Keputusan pemilihan model harus mempertimbangkan tujuan penggunaan, batasan infrastruktur, kebutuhan privasi, serta pendekatan integrasi. Dalam dunia AI yang semakin modular dan terfragmentasi, pemahaman akan karakteristik tiap model seperti ini menjadi sangat penting agar teknologi yang digunakan benar-benar optimal dan sesuai konteks.

BAB III

DASAR-DASAR PROMPT ENGINEERING

3.1 Pengantar Prompt Engineering

Prompt engineering adalah keterampilan baru yang lahir seiring berkembangnya kecerdasan buatan generatif, khususnya Large Language Models (LLM). Istilah ini merujuk pada proses sistematis dalam merancang, mengatur, dan mengoptimalkan instruksi (prompt) agar model dapat menghasilkan keluaran yang lebih efektif, relevan, dan sesuai dengan kebutuhan pengguna. Prompt berfungsi sebagai jembatan komunikasi antara manusia dan mesin, menentukan bagaimana maksud manusia diterjemahkan menjadi teks masukan, serta bagaimana model merespons dengan informasi yang bernilai. Dengan demikian, kualitas keluaran yang diberikan LLM sangat dipengaruhi oleh kualitas prompt yang diajukan.

Karakteristik LLM menjadi alasan mengapa prompt engineering menjadi sangat penting. LLM tidak “berpikir” sebagaimana manusia, melainkan bekerja berdasarkan analisis pola bahasa, distribusi probabilitas kata, serta asosiasi statistik yang terbentuk dari data pelatihan dalam skala sangat besar. Model ini tidak memiliki pemahaman kognitif maupun kesadaran, sehingga keluaran yang dihasilkan sepenuhnya ditentukan oleh cara instruksi disusun. Sebuah instruksi yang ambigu, terlalu umum, atau tidak terstruktur sering kali menghasilkan jawaban yang dangkal atau tidak relevan. Sebaliknya, instruksi yang jelas, detail, dan kontekstual dapat memunculkan respons yang lebih kaya, spesifik, serta bermanfaat.

Kejelasan instruksi merupakan prinsip utama dalam prompt engineering. Dalam praktiknya, prompt tidak sekadar berbentuk pertanyaan sederhana, melainkan dapat dikembangkan menjadi perintah yang menyertakan konteks, gaya bahasa, batasan, hingga format keluaran yang diinginkan. Strategi penyusunan instruksi ini menjadikan prompt engineering bukan sekadar keterampilan teknis, melainkan juga keterampilan strategis yang memengaruhi efektivitas interaksi dengan model AI.

Dalam konteks praktis, prompt engineering memiliki cakupan yang luas. Di bidang bisnis, prompt yang tepat dapat digunakan untuk menyusun laporan, menganalisis data pasar, atau menghasilkan rekomendasi strategis. Dalam pendidikan dan penelitian, prompt dapat mengarahkan model untuk menyajikan materi pembelajaran yang adaptif, menyusun rangkuman literatur, atau membantu analisis data penelitian. Di ranah kreatif, prompt yang dirancang dengan baik mampu mendorong munculnya ide cerita, naskah, desain, hingga

karya seni digital. Bahkan dalam pengembangan perangkat lunak, prompt dapat diarahkan untuk membantu menulis kode, mendeteksi kesalahan, atau menghasilkan dokumentasi teknis yang rapi. Dengan demikian, penguasaan prompt engineering memberikan nilai tambah lintas disiplin.

Untuk memahami pengaruh penyusunan prompt, dapat diperhatikan perbandingan berikut. Jika seseorang memberikan prompt yang sangat umum seperti: “Jelaskan AI.”, model akan menghasilkan penjelasan singkat dan cenderung generik mengenai kecerdasan buatan. Namun, jika prompt disusun lebih spesifik, misalnya: “Jelaskan konsep Artificial Intelligence untuk pemula, sertakan tiga contoh penerapannya di bidang kesehatan, dan gunakan bahasa yang sederhana.”, maka hasil keluaran akan lebih terarah. Model tidak hanya mendefinisikan AI, tetapi juga menyesuaikan bahasa agar mudah dipahami, memberikan tiga contoh yang relevan di bidang kesehatan, serta menyajikan jawaban sesuai format yang diminta.

Perbandingan sederhana ini menunjukkan bahwa prompt engineering merupakan keterampilan yang berorientasi pada strategi komunikasi. Bukan hanya tentang “apa” yang ditanyakan, tetapi juga tentang “bagaimana” instruksi itu dirumuskan. Dengan penguasaan dasar-dasar prompt engineering, pengguna dapat memaksimalkan potensi LLM untuk mendukung produktivitas, kreativitas, maupun pengambilan keputusan di berbagai bidang.

3.2 Konsep Dasar Prompt Engineering

Untuk memahami prompt engineering secara menyeluruh, terlebih dahulu perlu dipahami apa itu prompt, elemen-elemen yang membentuknya, serta bagaimana variasi atau kategorinya digunakan dalam praktik. Konsep dasar ini menjadi fondasi penting sebelum beralih pada strategi lanjutan yang lebih kompleks.

Definisi Prompt

Prompt dapat didefinisikan sebagai instruksi, pertanyaan, atau arahan yang diberikan kepada model AI untuk menghasilkan keluaran tertentu, baik berupa teks, ringkasan, analisis, kode program, maupun rekomendasi. Dengan kata lain, prompt adalah bentuk komunikasi eksplisit yang menjembatani maksud pengguna dengan mekanisme prediktif model. Prompt bisa sesederhana sebuah pertanyaan seperti “Apa itu machine learning?”, bisa pula berbentuk perintah seperti “Tuliskan ringkasan artikel ini dalam 100 kata,” atau bahkan berupa instruksi kreatif seperti “Buat puisi bertema cinta dan teknologi dengan nuansa futuristik.”

Elemen Utama dalam Prompt

Agar lebih efektif, sebuah prompt idealnya memuat empat elemen kunci. Pertama adalah konteks, yakni informasi tambahan yang memberikan latar belakang sehingga model memahami situasi dengan lebih tepat. Misalnya, instruksi “Sebagai seorang dosen AI, jelaskan cara kerja algoritma Decision Tree” akan lebih relevan dibanding sekadar “Jelaskan Decision Tree.” Kedua adalah instruksi, yaitu tugas utama yang harus dilakukan model. Instruksi ini dapat berupa permintaan penjelasan, analisis, perbandingan, atau penyusunan laporan. Ketiga adalah detail, berupa spesifikasi tambahan untuk memperjelas hasil yang diharapkan, misalnya “gunakan bahasa formal” atau “sertakan diagram sederhana.” Terakhir adalah format output, yakni bentuk keluaran yang diinginkan, apakah berupa tabel, poin-poin, paragraf naratif, atau bahkan kode program. Dengan kombinasi keempat elemen ini, prompt dapat diarahkan untuk menghasilkan jawaban yang lebih terstruktur, konsisten, dan sesuai tujuan pengguna.

Kategori Prompt

Dalam praktiknya, prompt dapat diklasifikasikan ke dalam beberapa kategori. Prompt langsung (direct prompt) adalah instruksi singkat tanpa detail tambahan, cocok untuk tugas sederhana. Misalnya, “Definisikan Internet of Things.” Prompt kontekstual (contextual prompt) menambahkan informasi latar belakang agar hasil lebih relevan. Contohnya, “Sebagai ahli teknologi, jelaskan konsep Internet of Things untuk mahasiswa tingkat akhir.” Selanjutnya, prompt instruktif (instructional prompt) berisi arahan yang lebih terperinci, sering kali dalam bentuk langkah demi langkah. Misalnya, “Jelaskan lima manfaat Internet of Things, kemudian berikan contoh implementasinya di bidang pertanian.” Variasi kategori ini menunjukkan bahwa semakin kaya konteks dan detail sebuah prompt, semakin besar kemungkinan model menghasilkan keluaran yang akurat dan bermanfaat.

Prinsip Menulis Prompt yang Efektif

Selain mengetahui elemen dan kategori, penting bagi pengguna untuk memahami prinsip-prinsip menulis prompt yang efektif. Prinsip pertama adalah spesifik dan jelas. Instruksi yang kabur cenderung menghasilkan jawaban generik. Misalnya, perintah “Buat laporan” terlalu umum, sementara instruksi “Buat laporan keuangan sederhana dalam format tabel Excel untuk periode Januari–Maret 2025” jauh lebih terarah. Prinsip kedua adalah menggunakan

konteks, karena tanpa informasi pendukung, model akan menjawab secara umum sesuai pola data latihannya. Prinsip ketiga adalah menentukan format output secara eksplisit, sebab format yang tidak ditentukan bisa membuat model memilih bentuk jawaban yang tidak sesuai kebutuhan. Terakhir, penggunaan bahasa yang terstruktur, ringkas, dan konsisten juga sangat penting agar instruksi mudah ditangkap oleh model tanpa menimbulkan ambiguitas.

Dengan memahami definisi, elemen, kategori, dan prinsip dasar ini, pengguna dapat mengembangkan prompt yang lebih efektif. Fondasi inilah yang akan menentukan sejauh mana interaksi dengan LLM dapat menghasilkan jawaban yang berkualitas, relevan, dan sesuai dengan tujuan yang diinginkan. Bab-bab selanjutnya akan membahas teknik lanjutan yang memperluas cakupan prompt engineering, tetapi penguasaan konsep dasar ini tetap menjadi syarat utama untuk memaksimalkan potensi model kecerdasan buatan generatif.

3.3 Kesalahan Umum dalam Prompt Engineering

Meskipun konsep dasar prompt engineering relatif mudah dipahami, dalam praktiknya banyak pengguna yang masih melakukan kesalahan mendasar ketika menyusun instruksi untuk model. Kesalahan-kesalahan ini sering kali menyebabkan keluaran yang dihasilkan kurang relevan, tidak terstruktur, atau bahkan sama sekali tidak sesuai dengan tujuan awal. Dengan memahami kesalahan umum tersebut, pengguna dapat menghindarinya sekaligus meningkatkan efektivitas interaksi dengan model.

Salah satu kesalahan paling sering terjadi adalah menulis prompt yang terlalu umum atau ambigu. Instruksi seperti “Jelaskan teknologi” atau “Buat laporan” terlalu luas cakupannya, sehingga model hanya akan memberikan jawaban generik. Model tidak mampu menebak maksud spesifik pengguna tanpa arahan yang jelas. Sebaliknya, instruksi yang lebih detail, misalnya “Buat laporan keuangan sederhana dalam format tabel untuk periode Januari–Maret 2025,” akan menghasilkan keluaran yang lebih relevan dan bermanfaat.

Kesalahan berikutnya adalah tidak menentukan format output dengan jelas. Model generatif pada dasarnya fleksibel dalam menyusun jawaban, sehingga tanpa arahan eksplisit, format keluaran bisa sangat bervariasi. Jika pengguna menginginkan daftar, tabel, atau paragraf singkat, maka format tersebut perlu disebutkan secara langsung dalam prompt. Kegagalan menyebutkan format sering kali membuat hasil keluaran tidak sesuai harapan dan memerlukan penyuntingan tambahan.

Kesalahan ketiga adalah mengabaikan konteks. LLM bekerja berdasarkan asosiasi pola bahasa dari data pelatihannya. Tanpa konteks yang cukup, jawaban

model akan bersifat umum atau dangkal. Misalnya, prompt “Jelaskan Internet of Things” mungkin hanya menghasilkan definisi singkat. Tetapi dengan menambahkan konteks, seperti “Sebagai konsultan bisnis digital, jelaskan manfaat Internet of Things untuk usaha kecil menengah,” keluaran akan jauh lebih relevan dan terarah.

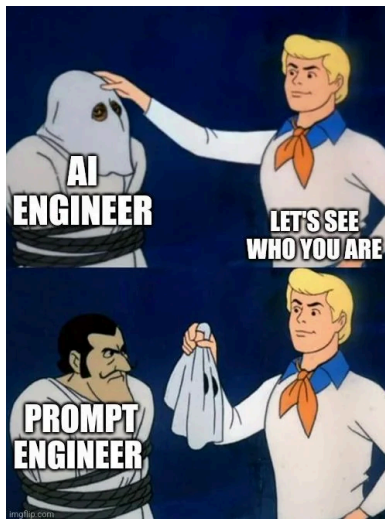
Kesalahan lain yang tidak kalah penting adalah berharap hasil sempurna tanpa iterasi. Banyak pengguna mengira bahwa prompt pertama yang mereka tulis akan langsung menghasilkan jawaban ideal. Padahal, prompt engineering pada dasarnya merupakan proses eksploratif dan iteratif. Dibutuhkan percobaan berulang, penyempurnaan instruksi, serta pengaturan ulang detail untuk mendapatkan hasil yang optimal. Dengan melakukan iterasi, pengguna tidak hanya memperbaiki hasil, tetapi juga belajar memahami cara model merespons berbagai bentuk instruksi.

Terakhir, kesalahan yang cukup sering dilakukan adalah menyusun prompt dengan kalimat terlalu panjang dan bertele-tele. Instruksi yang rumit dan penuh anak kalimat justru dapat membingungkan model, sehingga interpretasi menjadi keliru. Model lebih mudah dipandu dengan instruksi yang ringkas, terstruktur, dan menggunakan kalimat yang jelas. Oleh karena itu, penggunaan kalimat yang padat informasi tetapi tidak berlebihan menjadi kunci agar model memahami maksud pengguna secara akurat.

BAB IV

TEKNIK PROMPTING TINGKAT LANJUT

Teknik prompting tingkat lanjut dirancang untuk mengoptimalkan kemampuan Large Language Models (LLM) dalam menghasilkan jawaban yang lebih akurat, terstruktur, dan relevan. Tidak seperti teknik dasar (zero-shot, one-shot, few-shot), metode ini berfokus pada strategi yang memberi instruksi tambahan, membentuk pola penalaran, atau membangun iterasi agar hasil semakin sesuai kebutuhan.



Prompting Tingkat Lanjut Mampu Memberikan Hasil yang Lebih Baik

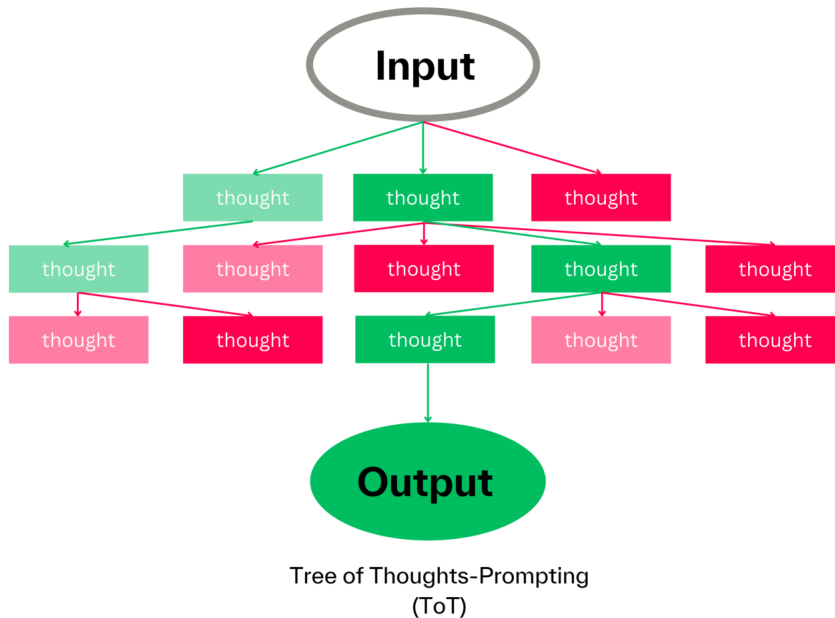
4.1 Chain-of-Thought (CoT) Prompting

Chain-of-Thought (CoT) Prompting merupakan teknik yang memungkinkan model bahasa besar (LLM) menuliskan proses berpikirnya terlebih dahulu sebelum sampai pada jawaban akhir. Alih-alih langsung memberi hasil, model diarahkan untuk menyusun langkah-langkah penalaran yang logis. Dengan cara ini, pembaca tidak hanya menerima jawaban mentah, melainkan juga bisa melihat bagaimana jawaban tersebut dibangun. Transparansi ini penting, terutama ketika model digunakan untuk persoalan yang menuntut ketelitian.

Teknik ini sangat bermanfaat untuk permasalahan yang membutuhkan penalaran bertahap, seperti:

- Soal matematika atau aritmetika yang memerlukan hitungan langkah demi langkah.

- Analisis logika formal dan pengambilan keputusan.
- Penalaran hukum berbasis norma dan fakta kasus.
- Analisis data yang menuntut interpretasi bertahap.



Alur Kerja Chain of Thought

Dalam analisis hukum, CoT membantu menelusuri dasar norma yang digunakan hingga kesimpulan dibuat. Dalam konteks data, model dapat menguraikan tahapan analisis sebelum akhirnya menampilkan hasil interpretasi. Dengan kata lain, CoT membuat proses berpikir model lebih mudah diikuti dan dievaluasi.

Salah satu cara sederhana untuk memicu CoT adalah dengan menambahkan instruksi seperti: “Jelaskan langkah-langkah berpikir Anda sebelum memberikan jawaban akhir.” atau dalam bahasa Inggris: “Let’s think step by step before giving the final answer.” Tambahan singkat ini sering kali cukup untuk mengubah cara model merespons, dari jawaban langsung menjadi penalaran yang runtut.

Sebagai contoh, bayangkan soal matematika sederhana: “Sebuah toko menjual 15 kotak pensil. Setiap kotak berisi 12 pensil. Berapa jumlah seluruh pensil?” Jika menggunakan CoT, model akan menjawab: pertama menjelaskan bahwa satu kotak berisi 12 pensil, lalu menghitung jumlah total dengan

mengalikan 15×12 , dan akhirnya menyimpulkan bahwa total pensil adalah 180 buah. Dengan cara ini, pengguna bisa memahami proses hitungannya, bukan hanya menerima angka final.

Dalam konteks hukum, CoT juga relevan. Misalnya, ketika diminta menganalisis Pasal 1320 KUH Perdata mengenai syarat sah perjanjian. Model dapat memulai dengan menuliskan empat syarat sah kontrak, lalu menghubungkannya dengan kasus yang diberikan. Jika ternyata syarat “kesepakatan bebas” tidak terpenuhi, maka model akan menjelaskan bahwa kontrak tersebut tidak sah, sebelum akhirnya menyampaikan kesimpulan hukum.

Secara garis besar, alur kerja CoT dapat digambarkan sebagai berikut: pengguna memberikan prompt yang meminta penjelasan langkah demi langkah, model kemudian menyusun proses penalaran, menguraikan logikanya secara bertahap, lalu menutup dengan jawaban akhir.

Namun, seperti teknik lainnya, CoT juga memiliki keterbatasan. Jawaban yang dihasilkan bisa menjadi terlalu panjang, bahkan kadang bertele-tele. Ada pula risiko overthinking, di mana model menambahkan penjelasan yang tidak relevan. Selain itu, karena memerlukan lebih banyak teks, teknik ini juga mengonsumsi token lebih banyak sehingga berimplikasi pada biaya penggunaan. Dan yang tak kalah penting, jika sejak awal model membuat kesalahan dalam penalaran, maka kesalahan tersebut bisa berlanjut hingga kesimpulan akhir.

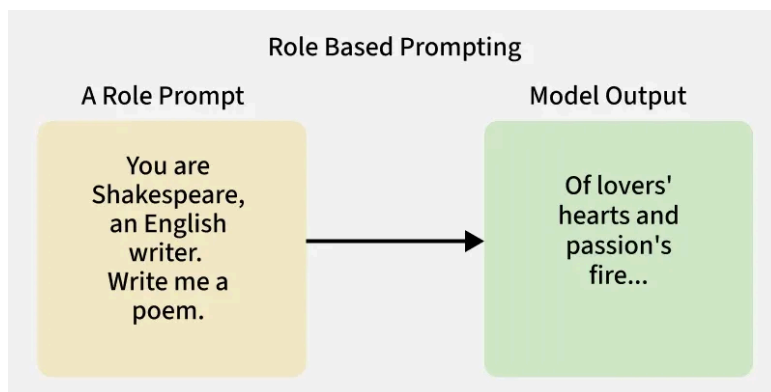
Walau demikian, CoT tetap menjadi salah satu teknik paling populer dan efektif dalam prompt engineering. Ia membantu menjembatani kebutuhan manusia untuk melihat logika di balik jawaban AI, dan membuka jalan menuju interaksi yang lebih transparan serta dapat dipertanggungjawabkan.

4.2 Role Prompting (“Act as...”)

Salah satu teknik prompting yang paling populer dan mudah dipahami adalah Role Prompting. Intinya, pengguna meminta model untuk berperan sebagai tokoh, profesi, atau sistem tertentu. Dengan cara ini, model diarahkan agar menjawab sesuai gaya, sudut pandang, dan pengetahuan yang biasanya dimiliki oleh peran tersebut.

Sebagai contoh, instruksi sederhana seperti “Act as a lawyer and draft a simple contract for a house rental agreement” akan membuat model menjawab dengan bahasa formal hukum. Jika instruksinya diganti menjadi “Act as a teacher and explain the theory of relativity with a simple example”, maka gaya jawabannya akan berubah: lebih sederhana, komunikatif, dan disesuaikan untuk audiens pelajar.

Kekuatan utama teknik ini terletak pada fleksibilitasnya. Model dapat diminta untuk berperan sebagai pengacara, dokter, dosen, penulis kreatif, jurnalis, bahkan sistem komputer tertentu. Dengan demikian, pengguna bisa menyesuaikan gaya bahasa, tingkat kedalaman penjelasan, atau sudut pandang jawaban dengan lebih presisi. Hal ini membuat Role Prompting sangat cocok untuk simulasi percakapan, penyusunan naskah kreatif, ataupun latihan profesional.



Ilustrasi Cara Kerja Role Prompting

Dalam praktiknya, format umum prompt biasanya berupa kalimat sederhana seperti “Act as a [role] and explain ...” atau dalam bahasa Indonesia, “Bertindaklah sebagai [peran] dan jelaskan ...”. Tambahan detail tentang audiens atau gaya komunikasi dapat memperkuat hasil. Misalnya: “Bertindaklah sebagai dosen hukum tata negara, jelaskan judicial review dalam bahasa yang mudah dipahami mahasiswa tingkat awal.”

Meski efektif, teknik ini juga memiliki keterbatasan. Jawaban yang dihasilkan tidak bisa langsung dianggap benar secara profesional, terutama dalam bidang yang menuntut akurasi tinggi seperti hukum, medis, atau keuangan. Model dapat menghasilkan teks yang terdengar meyakinkan, tetapi tetap memerlukan verifikasi oleh pakar manusia. Dengan kata lain, Role Prompting sangat berguna untuk pembelajaran, simulasi, dan eksplorasi, tetapi tidak boleh dijadikan pengganti mutlak keahlian profesional.

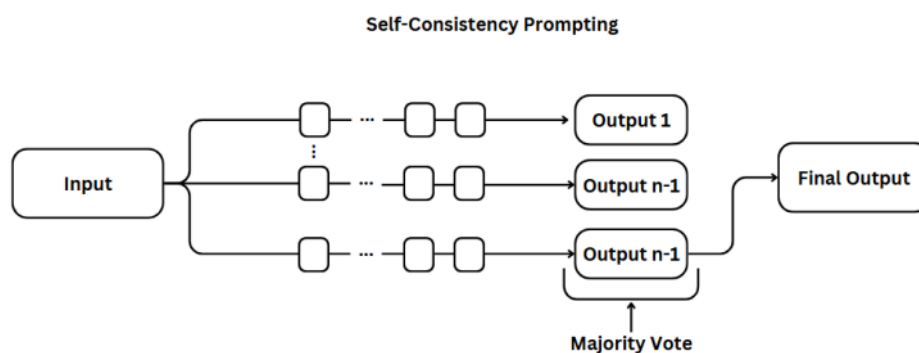
Secara sederhana, alurnya dapat digambarkan sebagai berikut: pengguna memberikan perintah “berperan sebagai ...”, model menyesuaikan gaya dan isi jawabannya dengan peran tersebut, lalu menghasilkan respons yang konsisten dengan karakter yang diminta. Teknik ini memberi kesan seolah-olah pengguna sedang berbicara langsung dengan sosok tertentu, meskipun pada kenyataannya yang menjawab tetaplah sebuah sistem kecerdasan buatan.

4.3 Self-asking dan Self-consistency

Selain Chain-of-Thought dan Role Prompting, terdapat teknik lain yang berfokus pada peningkatan kualitas penalaran model, yaitu Self-asking dan Self-consistency. Keduanya sama-sama berupaya membuat model “berpikir lebih dalam” sebelum memberikan jawaban akhir, tetapi dengan pendekatan yang berbeda.

Self-asking adalah teknik di mana model diarahkan untuk terlebih dahulu mengajukan pertanyaan-pertanyaan tambahan kepada dirinya sendiri sebelum mencoba menjawab. Dengan begitu, model tidak langsung melompat pada kesimpulan, melainkan menelusuri masalah dengan memecahnya menjadi bagian-bagian kecil. Misalnya, ketika diberi pertanyaan kompleks tentang sejarah, model dapat diarahkan untuk bertanya dulu: “Periode waktu mana yang relevan?” atau “Siapa tokoh utama dalam peristiwa ini?” Baru setelah itu model merangkai jawaban. Cara ini membuat hasil lebih runtut dan biasanya lebih akurat.

Sementara itu, Self-consistency berfokus pada pencarian jawaban yang paling stabil di antara beberapa kemungkinan. Model diminta menghasilkan beberapa jalur penalaran sekaligus, lalu memilih kesimpulan yang paling konsisten atau paling sering muncul. Teknik ini mirip dengan metode pemungutan suara dalam logika: semakin sering sebuah jawaban muncul dari jalur penalaran yang berbeda, semakin besar kemungkinan jawabannya benar.



Alur Self-consistency Prompting (ubiai)

Kedua teknik ini sangat berguna dalam konteks soal yang menuntut ketelitian tinggi, seperti matematika, logika formal, sains, maupun analisis hukum. Misalnya, dalam menyelesaikan soal hitungan, model dengan self-asking akan lebih hati-hati karena menanyakan langkah-langkah kecil sebelum

menghitung. Dalam analisis hukum, self-consistency membantu menemukan interpretasi norma yang paling masuk akal dari beberapa alternatif.

Tentu saja, teknik ini juga memiliki keterbatasan. Proses self-asking dan self-consistency biasanya memakan lebih banyak token, sehingga biayanya lebih tinggi dan waktu respons lebih lama. Selain itu, meskipun lebih hati-hati, model tetap bisa salah jika data latihannya tidak cukup kuat pada domain tertentu.

Jika digambarkan secara sederhana, alurnya adalah sebagai berikut:

- Self-asking: pertanyaan → model membuat pertanyaan tambahan → model menjawab berdasarkan pertanyaan tambahan.
- Self-consistency: pertanyaan → model membuat beberapa jalur penalaran → model memilih jawaban yang paling konsisten.

Dengan teknik ini, kualitas jawaban cenderung meningkat, karena model diarahkan untuk tidak hanya menebak sekali, melainkan menimbang ulang berbagai kemungkinan sebelum memberi kesimpulan akhir.

4.4 ReAct Prompting (Reasoning + Action)

Seiring berkembangnya kecerdasan buatan, muncul kebutuhan agar model tidak hanya mampu “berpikir”, tetapi juga dapat “bertindak” berdasarkan pemikirannya. Inilah yang melahirkan teknik ReAct Prompting, singkatan dari Reasoning and Action. Teknik ini menggabungkan dua kemampuan: penalaran langkah demi langkah seperti Chain-of-Thought dan tindakan konkret seperti melakukan pencarian data, memanggil API, atau mengeksekusi instruksi tertentu.

Dengan ReAct, model tidak berhenti pada analisis, tetapi juga diarahkan untuk mengambil langkah lanjut sesuai kebutuhan. Misalnya, jika ditanya tentang kurs mata uang terbaru, model tidak hanya menalar pentingnya data terkini, tetapi juga melakukan “aksi” berupa memanggil mesin pencari atau API keuangan, kemudian menyajikan jawaban berbasis data aktual.

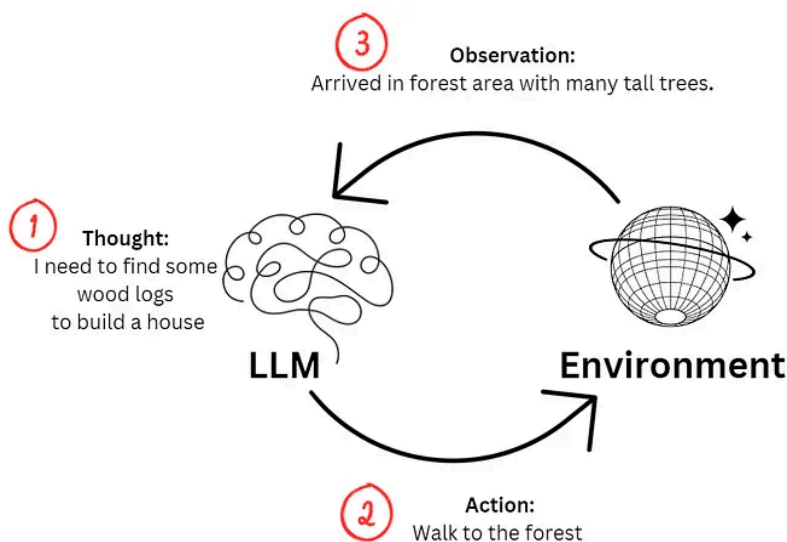
Contoh sederhana: ketika diberi prompt “Anda adalah asisten riset hukum. Analisis pasal terkait kebebasan pers dalam UUD 1945, lalu temukan minimal dua referensi akademik yang relevan dari database hukum.” Model akan melakukan dua hal: pertama, menganalisis norma kebebasan pers secara konseptual (reasoning), lalu kedua, mencari referensi akademik (action) untuk melengkapi analisisnya. Hasil akhirnya lebih kaya, karena tidak hanya berupa argumen, tetapi juga didukung bukti dari sumber eksternal.

Kekuatan utama teknik ini adalah fleksibilitasnya. ReAct cocok diterapkan pada sistem yang terintegrasi dengan berbagai alat tambahan, seperti mesin

pencarian, basis data, kalkulator, atau bahkan sistem manajemen perusahaan. Dengan memadukan penalaran dan aksi, AI berfungsi lebih mirip seperti “agen cerdas” yang dapat berpikir sekaligus bertindak sesuai kebutuhan konteks.

Namun, ada pula keterbatasan yang perlu diperhatikan. Integrasi aksi memerlukan sistem tambahan di luar model itu sendiri, sehingga tidak semua platform mendukung ReAct secara langsung. Selain itu, jika aksi yang diambil tidak tepat (misalnya pencarian yang salah arah), maka hasil akhirnya juga bisa menyesatkan meskipun penalarannya logis.

Jika dirangkum dalam bentuk alur, ReAct Prompting dapat digambarkan sebagai berikut:



Alur ReAct Prompting

Dalam kerangka ReAct Prompting, model diarahkan untuk tidak hanya menjawab secara langsung, melainkan menempuh tahapan berpikir dan bertindak. Setiap tahap memberi struktur yang jelas, sehingga proses yang ditempuh model lebih transparan dan hasil akhirnya lebih dapat dipertanggungjawabkan. Berikut adalah tahapan dalam ReAct Prompting.

Thought (Penalaran / Langkah Berpikir)

Tahap pertama adalah proses berpikir internal. Model menganalisis instruksi, memecah masalah, dan merencanakan cara penyelesaiannya. Bagian ini biasanya berupa penjelasan mengapa suatu langkah dibutuhkan atau informasi apa yang perlu dicari.

Action (Tindakan / Eksekusi)

Setelah menyusun rencana, model melakukan tindakan konkret. Tindakan bisa berupa pencarian informasi di internet, pemanggilan API, menjalankan kalkulasi, atau mengakses basis data. Pada tahap ini, model mencoba melaksanakan apa yang direncanakan dalam Thought.

Observation (Pengamatan / Hasil Aksi)

Hasil dari tindakan tadi kemudian diamati. Misalnya, jika model melakukan pencarian, maka data yang ditemukan akan diproses di tahap ini. Observation berfungsi sebagai bahan mentah yang akan dipadukan kembali dengan penalaran untuk memastikan arah jawaban tetap benar.

Final Answer (Jawaban Akhir)

Tahap terakhir adalah penyusunan kesimpulan. Model menggabungkan penalaran (Thought) dengan data dari hasil pengamatan (Observation) untuk menghasilkan jawaban akhir. Dengan demikian, keluaran tidak hanya berupa dugaan logis, tetapi juga didukung oleh data nyata atau aksi yang telah dilakukan.

Contoh sederhana:

Prompt: “Berapa hasil penjumlahan $25 + 47$? Tunjukkan langkah ReAct.”

Thought: Untuk menjawab, saya perlu melakukan perhitungan dasar.

Action: Hitung $25 + 47$.

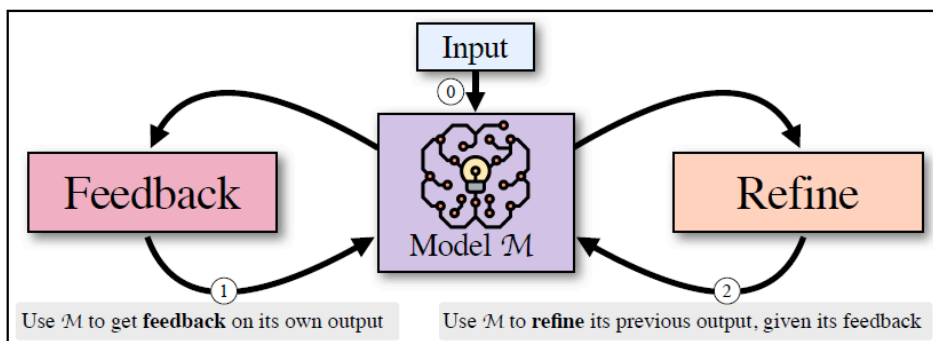
Observation: Hasil perhitungan adalah 72.

Final Answer: Jawaban akhirnya adalah 72.

4.5 Iterative Prompting & Refinement

Dalam praktik sehari-hari, jarang sekali sebuah prompt langsung menghasilkan jawaban yang sempurna pada percobaan pertama. Karena itu, muncul teknik yang disebut Iterative Prompting & Refinement. Intinya adalah menyusun prompt secara bertahap dengan melakukan perbaikan berulang sampai hasil yang diperoleh benar-benar sesuai kebutuhan.

Ada dua bentuk iterasi yang umum dikenal. Pertama, iterasi internal, yaitu ketika pengguna sendiri memodifikasi prompt berdasarkan output yang kurang tepat. Proses ini mirip dengan berdialog: model memberikan jawaban awal, lalu pengguna memperbaiki atau memperjelas instruksi agar hasil berikutnya lebih baik. Kedua, iterasi eksternal, yakni pendekatan otomatis di mana sistem secara mandiri memperbaiki prompt berdasarkan evaluasi terhadap output. Pendekatan ini dikenal juga dengan sebutan auto-prompt optimization, yang biasanya digunakan dalam aplikasi tingkat lanjut atau riset.



Cara Kerja Iterative Prompting & Refinement

Contoh penerapan teknik ini dapat ditemukan dalam berbagai konteks. Dalam penulisan artikel, misalnya, pengguna mungkin memulai dengan prompt yang sederhana: “Tuliskan artikel tentang energi terbarukan.” Hasil awal bisa jadi terlalu umum. Maka, prompt diperbaiki: “Tuliskan artikel sepanjang 800 kata tentang energi terbarukan di Indonesia, dengan fokus pada potensi tenaga surya.” Hasil berikutnya akan lebih spesifik, lebih fokus, dan sesuai kebutuhan.

Contoh lain ada pada analisis data. Prompt pertama mungkin hanya meminta ringkasan dataset. Setelah melihat hasilnya, pengguna dapat memperbaiki prompt dengan menambahkan instruksi baru: “Selain ringkasan, buat juga visualisasi grafik garis untuk tren lima tahun terakhir.” Dengan iterasi ini, output menjadi lebih lengkap dan informatif.

Keunggulan utama dari Iterative Prompting & Refinement adalah fleksibilitasnya. Teknik ini memungkinkan pengguna untuk terus mempersempit atau memperluas fokus jawaban sampai benar-benar sesuai dengan kebutuhan spesifik. Namun, di sisi lain, iterasi yang terlalu sering dan tidak sistematis bisa

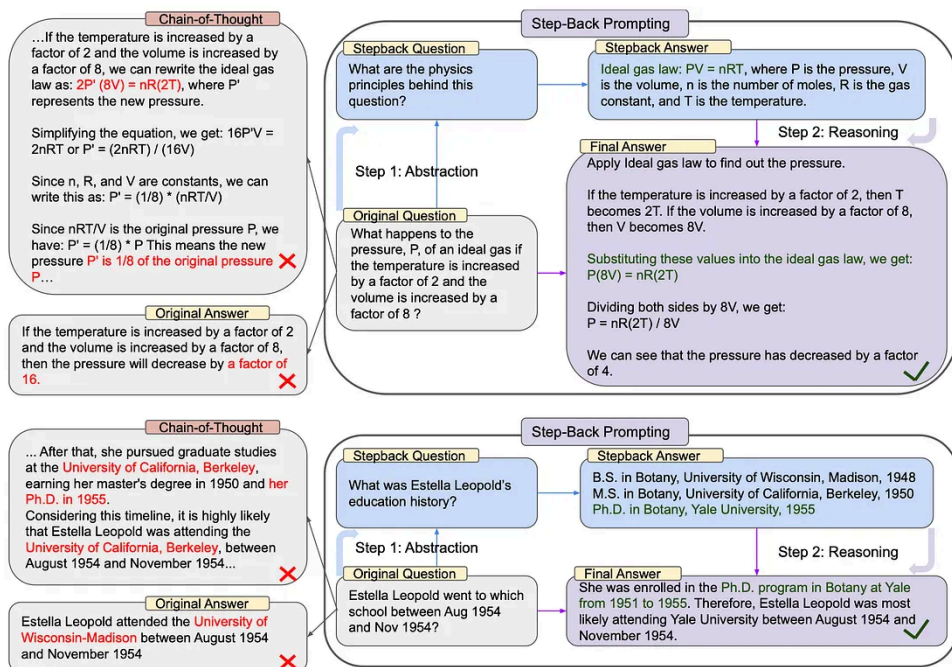
memakan waktu lebih lama, bahkan menimbulkan kebingungan jika arah perbaikan tidak jelas.

Alur kerja teknik ini dapat diringkas sebagai berikut: pengguna menyusun prompt awal → model memberikan output → pengguna atau sistem mengevaluasi hasil → prompt diperbaiki sesuai evaluasi → model menghasilkan output baru → proses diulang sampai hasil dirasa optimal.

Dengan teknik ini, interaksi dengan LLM menjadi proses kolaboratif yang bersifat dinamis, di mana manusia dan mesin saling menyesuaikan agar tercapai keluaran yang paling relevan dan bermanfaat.

4.6 Step-back Prompting

Salah satu pendekatan yang mulai banyak digunakan dalam prompt engineering adalah Step-back Prompting. Teknik ini mendorong model untuk tidak langsung melompat pada jawaban, melainkan “mundur selangkah” terlebih dahulu dengan cara menguraikan prinsip umum, asumsi dasar, atau kerangka penyelesaian sebelum memberikan hasil akhir. Dengan begitu, model diajak untuk berpikir lebih abstrak dan terstruktur, bukan sekadar bereaksi cepat terhadap pertanyaan yang diajukan.



Alur Cara Kerja Step-back Prompting (Google DeepMind)

Pendekatan ini sangat bermanfaat ketika berhadapan dengan masalah yang kompleks atau ambigu. Dengan melakukan “step-back”, model lebih dulu merumuskan aturan main: apa prinsip yang relevan, faktor apa yang harus diperhatikan, serta bagaimana rencana penyelesaiannya. Baru setelah itu, ia menyajikan jawaban akhir yang lebih matang. Dibandingkan dengan Chain-of-Thought yang langsung memecah persoalan ke langkah-langkah detail, Step-back menekankan penetapan kerangka besar sebelum detail dijabarkan.

Sebagai contoh, bayangkan sebuah soal cerita matematika sederhana tentang jumlah pensil dalam beberapa kotak. Jika hanya menggunakan CoT, model akan langsung menghitung 15 dikalikan 12 dan menyimpulkan hasilnya. Namun dengan Step-back, model akan lebih dulu menyebutkan prinsip yang digunakan (perkalian aritmetika), variabel yang terlibat (jumlah kotak dan isi per kotak), serta asumsi yang diambil (tidak ada faktor lain seperti pajak atau diskon). Setelah itu, barulah perhitungan dilakukan dan jawaban akhir diberikan.

Dalam konteks hukum, pendekatan ini juga sangat membantu. Misalnya ketika diminta menilai sah atau tidaknya suatu kontrak, model dengan Step-back akan memulai dengan menguraikan kerangka normatif terlebih dahulu, seperti syarat sah perjanjian menurut Pasal 1320 KUH Perdata. Lalu, ia mengidentifikasi fakta yang relevan dalam kasus yang diberikan, menyebutkan asumsi yang mendasari analisis, dan akhirnya memberikan kesimpulan hukum.

Alur kerja Step-back Prompting dapat digambarkan sebagai berikut: pengguna memberikan instruksi → model melakukan abstraksi masalah dengan menyebut prinsip dan asumsi → model menyusun rencana penyelesaian → model menyajikan jawaban akhir. Proses ini membuat hasil yang diperoleh lebih mudah ditelusuri, sekaligus mengurangi risiko bias atau kesalahan karena terburu-buru.

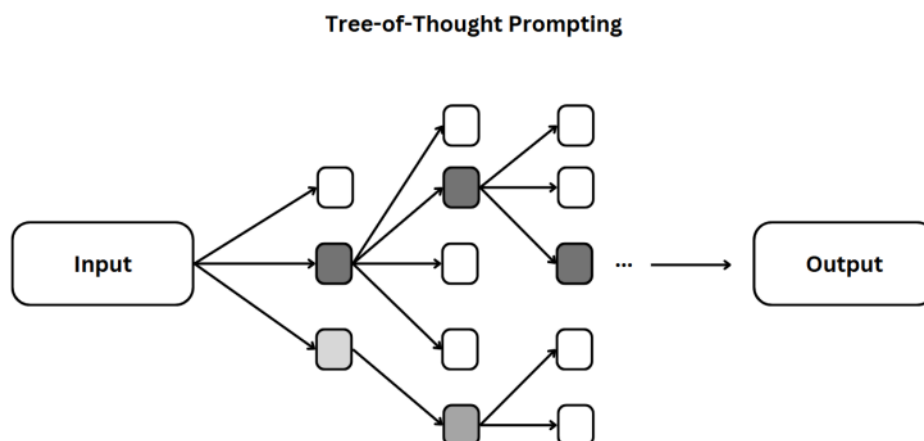
Meski demikian, teknik ini juga memiliki kelemahan. Terkadang respons yang dihasilkan menjadi terlalu umum atau bertele-tele, apalagi jika instruksi tidak menetapkan batasan panjang jawaban. Ada juga kemungkinan bahwa model memberikan kerangka abstrak yang sebenarnya tidak relevan. Oleh karena itu, dalam praktiknya, Step-back Prompting sering digabungkan dengan Chain-of-Thought. Dengan kombinasi ini, model terlebih dahulu menetapkan bingkai berpikir secara abstrak, lalu melanjutkan dengan langkah detail yang lebih konkret.

4.7 Tree-of-Thought (ToT) Prompting

Jika Chain-of-Thought menuntun model untuk menalar secara linear, maka Tree-of-Thought (ToT) mendorongnya untuk mengeksplorasi berbagai jalur penalaran seperti cabang sebuah pohon. Alih-alih hanya mengikuti satu alur berpikir dari awal hingga akhir, model diajak untuk menimbang beberapa kemungkinan solusi, menguji kelebihan dan kelemahan masing-masing, lalu memilih jalur yang paling masuk akal.

Pendekatan ini berangkat dari kenyataan bahwa banyak masalah tidak memiliki satu jawaban tunggal. Misalnya, dalam penyusunan strategi bisnis, ada beberapa opsi yang sama-sama masuk akal: menekan biaya produksi, meningkatkan kualitas produk, atau memperluas jaringan distribusi. Dengan ToT, model dapat menguraikan ketiga jalur tersebut, lalu membandingkannya sebelum merekomendasikan yang paling sesuai.

Cara kerja Tree-of-Thought biasanya dimulai dari identifikasi masalah utama, kemudian bercabang ke beberapa hipotesis atau alternatif solusi. Dari masing-masing cabang, model menelusuri implikasi atau konsekuensinya. Setelah semua opsi dieksplorasi, model menilai konsistensi dan relevansi setiap jalur, sebelum akhirnya memilih kesimpulan akhir.



Alur Kerja Tree-of-Thought (ToT) Prompting (ubiai)

Sebagai contoh, bayangkan pertanyaan: “Apa solusi terbaik untuk mengurangi kemacetan lalu lintas di kota besar?” Dengan pendekatan linear, model mungkin langsung memilih satu opsi, misalnya peningkatan transportasi umum. Namun dengan ToT, model akan mengeksplorasi beberapa cabang:

membangun infrastruktur jalan baru, meningkatkan layanan transportasi publik, menerapkan kebijakan ganjil-genap, atau mendorong telecommuting. Dari tiap cabang, model akan menilai dampaknya, biaya, dan keberlanjutan. Akhirnya, ia dapat menyimpulkan bahwa kombinasi antara transportasi umum yang lebih baik dan kebijakan pembatasan kendaraan adalah solusi paling realistis.

Alur kerja ToT dapat disederhanakan menjadi: identifikasi masalah → generasi beberapa jalur penalaran → eksplorasi konsekuensi tiap jalur → evaluasi konsistensi → pemilihan jalur terbaik.

Keunggulan utama teknik ini adalah kemampuannya menghasilkan jawaban yang lebih kreatif, beragam, dan mendekati cara berpikir manusia ketika menghadapi masalah terbuka (open-ended problem). Namun, ada pula risiko yang perlu diperhatikan. Karena menjelajahi lebih banyak jalur, ToT sering kali membutuhkan lebih banyak token dan waktu pemrosesan. Selain itu, jika cabang yang dieksplorasi terlalu banyak, model bisa terjebak dalam eksplorasi yang berlarut-larut tanpa menyajikan kesimpulan yang ringkas.

Dengan demikian, ToT paling tepat digunakan pada masalah yang memang memiliki banyak alternatif solusi, seperti perencanaan strategi, brainstorming ide kreatif, atau analisis kebijakan publik. Pada kasus yang lebih sederhana, pendekatan linear seperti Chain-of-Thought mungkin sudah cukup.

4.8 Automatic Prompt Engineering (APE)

Selama ini, perbaikan prompt biasanya dilakukan secara manual oleh pengguna: mencoba instruksi pertama, melihat hasilnya, lalu memperbaiki kata-kata atau menambahkan detail hingga keluaran semakin sesuai. Namun, dalam skala besar, cara ini kurang efisien. Di sinilah muncul konsep Automatic Prompt Engineering (APE), yaitu pendekatan yang membuat model atau sistem secara otomatis menghasilkan, mengevaluasi, dan menyempurnakan prompt tanpa banyak intervensi manusia.

Secara sederhana, APE bekerja seperti asisten yang bukan hanya menjawab pertanyaan, tetapi juga mencari cara terbaik untuk merumuskan pertanyaan itu sendiri. Sistem dapat menghasilkan beberapa variasi prompt, menguji hasil masing-masing, lalu memilih versi yang paling efektif. Pendekatan ini sangat membantu dalam situasi yang membutuhkan kualitas konsisten, misalnya dalam pembuatan soal ujian, sistem tanya jawab otomatis, atau pipeline analisis data skala besar.

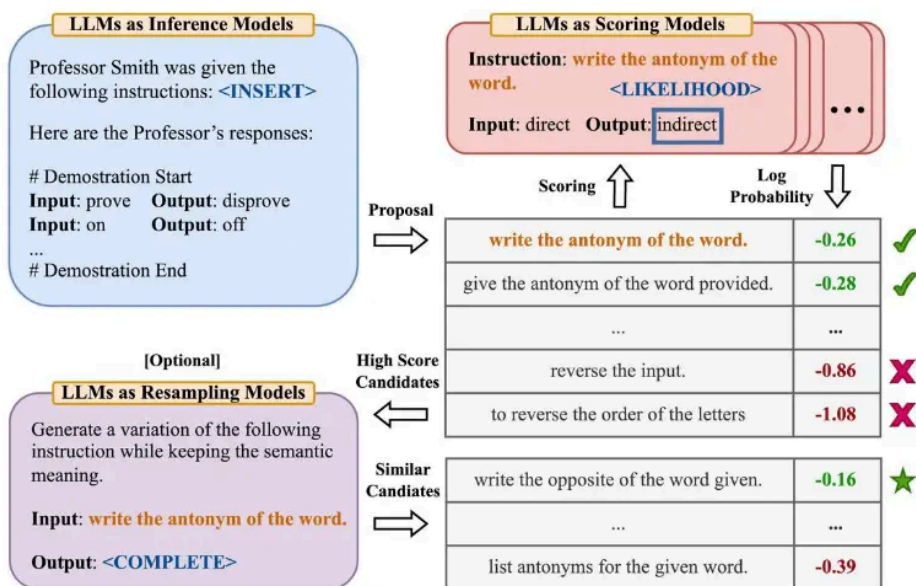
Sebagai contoh, bayangkan sebuah perusahaan ingin membuat chatbot layanan pelanggan yang mampu menjawab pertanyaan konsumen dengan nada sopan dan ramah. Daripada menyusun prompt manual satu per satu, sistem APE

dapat menghasilkan banyak versi instruksi seperti “Jawablah dengan sopan,” “Gunakan nada ramah dan profesional,” atau “Berikan solusi dengan bahasa yang mudah dipahami.” Setelah diuji, sistem akan memilih variasi yang menghasilkan tanggapan paling sesuai dengan standar perusahaan.

Alur kerja APE dapat diringkas sebagai berikut: pertama, sistem menghasilkan kumpulan kandidat prompt; kedua, setiap prompt diuji pada sejumlah contoh input; ketiga, output dari setiap kandidat dievaluasi (baik secara otomatis maupun manual); terakhir, prompt yang paling konsisten dipilih atau digabungkan untuk digunakan lebih lanjut.

Keunggulan utama teknik ini adalah skalabilitas. Dengan APE, penyusunan prompt tidak lagi terbatas pada eksperimen individu, melainkan dapat diotomatisasi sehingga lebih cepat dan konsisten. Namun, ada juga keterbatasan: proses ini bisa memerlukan sumber daya komputasi yang besar, terutama jika jumlah variasi prompt yang diuji sangat banyak. Selain itu, efektivitas APE sangat bergantung pada kriteria evaluasi yang digunakan—jika metriknya kurang tepat, hasil yang dipilih pun mungkin tidak benar-benar optimal.

Dalam praktiknya, APE sering digunakan pada aplikasi penelitian atau sistem produksi berskala besar, di mana kualitas output harus stabil dan dapat direplikasi. Untuk penggunaan sehari-hari, pengguna individu biasanya masih lebih mengandalkan iterasi manual yang sederhana.

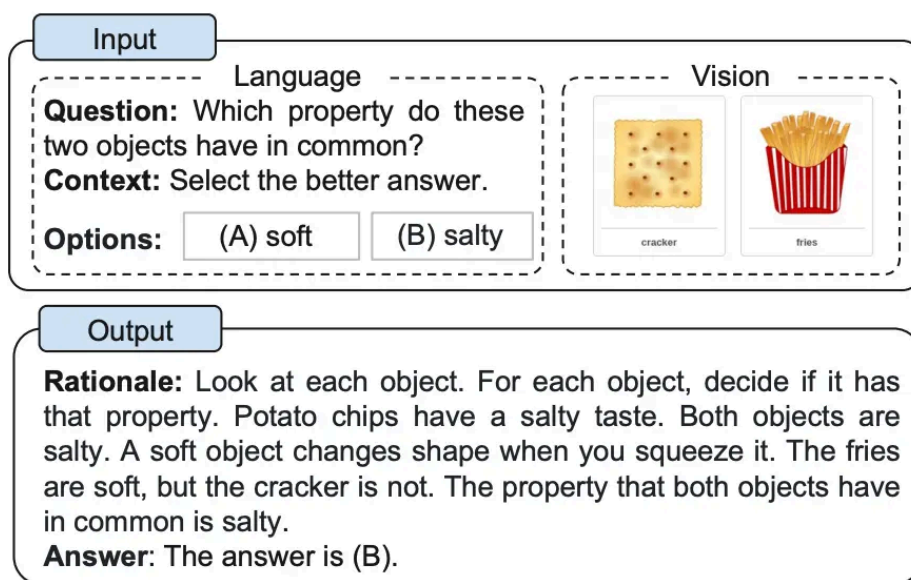


Alur Kerja Automatic Prompt Engineering (Zhou et al.)

4.9 Multimodal Prompting

Model bahasa generatif pada awalnya hanya berfokus pada teks. Namun, perkembangan terbaru menghadirkan kemampuan multimodal, yaitu model yang dapat menerima dan memproses lebih dari satu jenis input—misalnya teks, gambar, audio, atau video—dalam satu interaksi. Teknik multimodal prompting berarti menyusun instruksi yang menggabungkan beragam jenis masukan ini agar model dapat memberikan keluaran yang lebih kaya dan relevan.

Perbedaan utama dari prompt multimodal adalah bahwa instruksi tidak hanya berbentuk kalimat, melainkan bisa dilengkapi dengan konteks visual atau suara. Sebagai contoh, pengguna dapat mengunggah sebuah gambar produk dan memberi instruksi, “Deskripsikan produk ini dan buatkan teks iklan singkat untuk Instagram.” Model tidak hanya mengandalkan teks, tetapi juga membaca elemen visual dari gambar untuk memberikan hasil yang sesuai.



Contoh Multimodal Prompting (Zhang et al.)

Kemampuan multimodal membuka banyak kemungkinan. Dalam pendidikan, misalnya, seorang guru bisa mengunggah diagram fisika dan meminta penjelasan konsep di baliknya. Dalam medis, dokter dapat memberi instruksi berbasis gambar hasil pemindaian dengan tambahan pertanyaan diagnostik. Di dunia kreatif, seniman bisa memasukkan potongan musik atau

cuplikan video, lalu meminta AI mengusulkan pengembangan ide dari bahan tersebut.

Alur kerjanya dapat digambarkan sebagai berikut: pengguna menyusun prompt multimodal (teks + input lain), model memproses berbagai modalitas secara bersamaan, kemudian menggabungkan interpretasi tersebut menjadi jawaban akhir. Proses ini membuat keluaran lebih kontekstual karena mempertimbangkan lebih dari satu sumber informasi.

Kekuatan utama multimodal prompting adalah keluwesan: model dapat memahami instruksi dengan konteks visual, suara, maupun teks sekaligus. Namun, teknik ini juga memiliki keterbatasan. Tidak semua model mendukung input multimodal, dan dalam praktiknya kualitas keluaran sangat bergantung pada kecanggihan arsitektur model. Selain itu, penggunaan data visual atau audio sering kali menimbulkan tantangan privasi, terutama jika informasi yang diproses bersifat sensitif.

Dengan semua peluang dan tantangan tersebut, multimodal prompting kini dianggap sebagai salah satu arah masa depan interaksi manusia dengan AI. Di sinilah batas antara teks, gambar, dan suara mulai kabur, sehingga komunikasi dengan mesin semakin mendekati bentuk komunikasi manusia yang sebenarnya.

4.10 Prompt Chaining (Multi-step Prompting)

Tidak semua persoalan dapat diselesaikan dengan satu instruksi. Banyak kasus yang justru membutuhkan serangkaian prompt yang saling terhubung. Inilah yang disebut dengan Prompt Chaining atau Multi-step Prompting. Teknik ini memecah masalah besar menjadi beberapa langkah kecil, lalu setiap langkah dikerjakan oleh model secara berurutan hingga menghasilkan jawaban akhir yang lebih terarah dan kaya konteks.

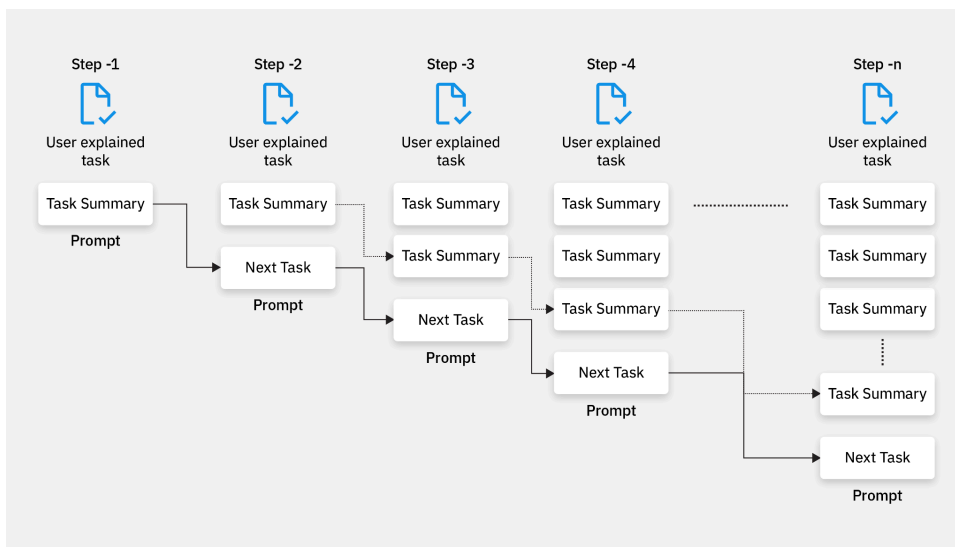
Prinsip dasarnya sederhana: alih-alih meminta model melakukan semuanya sekaligus, pengguna menyusun rantai instruksi. Output dari langkah pertama menjadi input bagi langkah berikutnya, dan seterusnya, sampai terbentuk sebuah alur penyelesaian yang utuh. Cara ini mirip dengan proses kerja manusia yang biasanya membagi tugas kompleks menjadi tahapan-tahapan lebih kecil agar mudah dikelola.

Sebagai contoh, bayangkan ingin membuat laporan bisnis lengkap. Dengan prompt chaining, langkah awal bisa berupa: “Ringkas laporan tahunan perusahaan X dalam lima poin utama.” Setelah itu, prompt kedua memanfaatkan ringkasan tersebut untuk meminta: “Analisis peluang dan risiko dari lima poin ini.” Lalu, prompt berikutnya bisa diarahkan: “Susun rekomendasi strategi

berdasarkan analisis risiko dan peluang tadi.” Hasil akhir pun jauh lebih fokus, karena dibangun dari proses yang sistematis.

Contoh lain adalah dalam konteks pendidikan. Prompt pertama dapat dipakai untuk mengekstrak konsep inti dari sebuah teks akademik. Prompt kedua lalu menyusun soal latihan berbasis konsep itu, dan prompt ketiga memberi kunci jawaban serta penjelasan singkat. Dengan begitu, lahirlah sebuah modul pembelajaran sederhana hanya dari rangkaian instruksi yang disusun secara berantai.

Kelebihan utama teknik ini terletak pada kendali yang lebih besar atas proses berpikir model. Setiap langkah dapat dievaluasi sebelum dilanjutkan, sehingga kesalahan bisa dikoreksi lebih dini. Selain itu, hasil akhirnya biasanya lebih rapi dan sesuai dengan tujuan. Namun, kelemahannya adalah waktu dan biaya pemrosesan yang lebih besar, karena melibatkan beberapa interaksi. Jika tidak dirancang secara efisien, rantai prompt bisa menjadi terlalu panjang dan justru membingungkan.



Langkah Prompt Chaining (IBM)

Dengan prompt chaining, model tidak lagi hanya menjawab instruksi tunggal, tetapi benar-benar diajak untuk mengikuti alur kerja yang bertahap, mendekati cara manusia menyelesaikan persoalan kompleks. Teknik ini sangat bermanfaat dalam aplikasi yang memerlukan konsistensi proses, seperti penyusunan dokumen hukum, perencanaan proyek, atau pembuatan modul pembelajaran.

4.11 Contrastive Prompting

Contrastive Prompting merupakan salah satu teknik prompt engineering yang berfokus pada pemberian instruksi dengan menampilkan contoh keluaran yang diinginkan (positive example) dan keluaran yang tidak diinginkan (negative example). Melalui pendekatan ini, model diarahkan tidak hanya untuk menghasilkan jawaban sesuai standar tertentu, tetapi juga untuk menghindari pola keluaran yang dianggap tidak tepat. Dengan demikian, teknik ini berperan sebagai mekanisme pengendalian kualitas hasil generatif melalui pembatasan eksplisit.

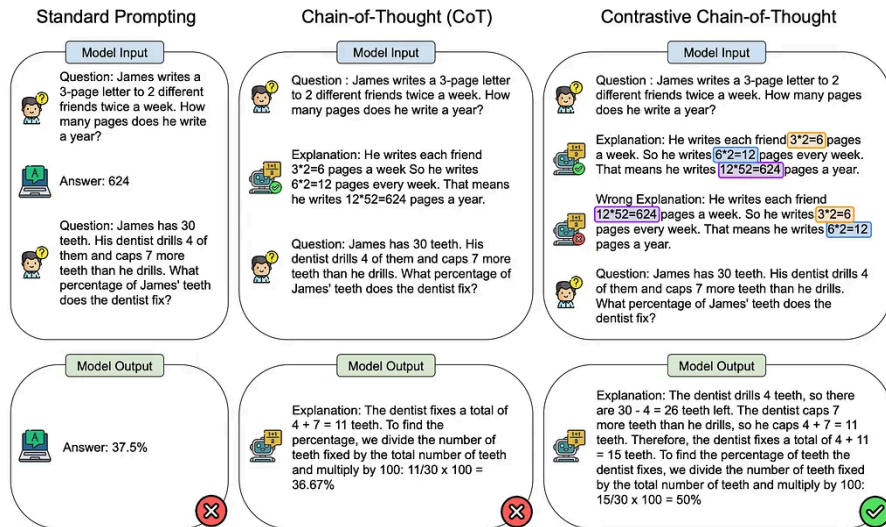
Secara konseptual, Contrastive Prompting dapat dipahami sebagai bentuk pembelajaran berbasis perbandingan (contrastive learning) yang diterapkan pada interaksi manusia dengan model bahasa. Pemberian contoh positif berfungsi sebagai rujukan normatif, sedangkan contoh negatif menjadi indikator kesalahan atau deviasi yang harus dihindari. Kombinasi keduanya menciptakan batasan semantik dan stilistika yang lebih jelas bagi model.

Sebagai ilustrasi, instruksi “Tuliskan iklan singkat untuk produk kopi” dapat diperluas dengan elemen kontras: “Contoh keluaran yang diinginkan: ‘Nikmati kopi pagi penuh energi!’ Contoh keluaran yang tidak diinginkan: ‘Kalau tidak minum kopi ini, harimu pasti gagal.’” Dengan kerangka ini, model diarahkan untuk meniru gaya persuasif positif dan sekaligus menghindari bentuk komunikasi yang manipulatif atau bernuansa negatif.

Penerapan Contrastive Prompting sangat relevan dalam berbagai konteks. Dalam penulisan akademik, misalnya, contoh kalimat yang terlalu informal dapat ditandai sebagai keluaran negatif, sementara gaya bahasa formal ditetapkan sebagai keluaran positif. Demikian pula dalam praktik hukum, perbedaan antara analisis normatif yang berbasis peraturan dan opini subjektif dapat ditetapkan secara eksplisit, sehingga model lebih konsisten menjaga objektivitasnya.

Dari sisi keunggulan, teknik ini memungkinkan kontrol yang lebih presisi terhadap gaya, isi, dan kualitas keluaran. Namun, terdapat keterbatasan tertentu. Jika contoh positif dan negatif tidak representatif atau tidak dirumuskan secara jelas, model berpotensi salah menafsirkan maksud instruksi. Selain itu, penggunaan banyak contoh kontras akan meningkatkan konsumsi token, sehingga berdampak pada efisiensi pemrosesan.

Secara keseluruhan, Contrastive Prompting dapat dipandang sebagai strategi yang memperkaya instruksi dengan dimensi normatif: bukan hanya menegaskan “apa yang harus dilakukan,” melainkan juga “apa yang harus dihindari.” Hal ini menjadikannya salah satu teknik yang efektif dalam memastikan konsistensi etis, stilistika, maupun substansi pada keluaran model bahasa.



Perbedaan Prompt Standar, CoT, dan Contrastive (Yew)

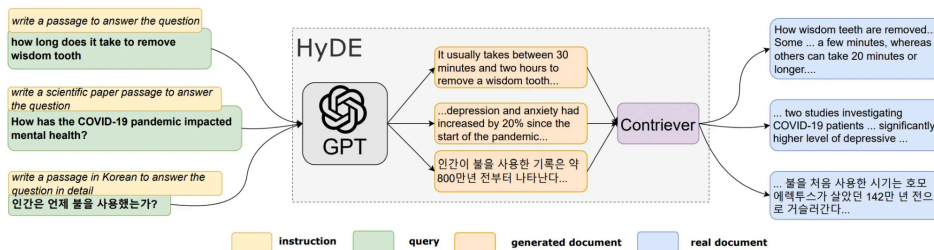
4.12 Hypothetical Document Embeddings (HyDE)

Hypothetical Document Embeddings (HyDE) merupakan teknik prompt engineering yang diperkenalkan dalam konteks retrieval-augmented generation (RAG). Prinsip dasar HyDE adalah menghasilkan sebuah dokumen hipotetis terlebih dahulu—berdasarkan pertanyaan atau instruksi pengguna—yang kemudian digunakan sebagai representasi (embedding) untuk melakukan pencarian informasi dalam basis data atau korpus eksternal.

Dengan pendekatan ini, model tidak langsung melakukan pencarian menggunakan query asli dari pengguna. Sebaliknya, ia terlebih dahulu “membayangkan” jawaban dalam bentuk dokumen singkat yang relevan. Dokumen hipotetis tersebut lalu diubah menjadi vektor embedding dan dibandingkan dengan vektor dokumen yang tersimpan dalam basis data. Strategi ini terbukti meningkatkan akurasi pencarian karena representasi hipotetis biasanya lebih kaya secara semantik dibanding query mentah yang mungkin terlalu singkat atau ambigu.

Sebagai contoh, jika pengguna mengajukan pertanyaan: “Apa dampak perubahan iklim terhadap ketahanan pangan di Asia Tenggara?”, alih-alih langsung menggunakan pertanyaan ini sebagai query, model akan terlebih dahulu menghasilkan sebuah ringkasan hipotetis, misalnya: “Dokumen ini membahas bagaimana perubahan iklim meningkatkan risiko gagal panen padi di Asia

Tenggara akibat banjir, kekeringan, dan pola cuaca ekstrem, serta dampaknya terhadap stabilitas harga pangan.” Ringkasan hipotetis ini kemudian dijadikan representasi embedding untuk mencari dokumen akademik yang paling mirip, sehingga hasil pencarian lebih relevan.



Ilustrasi Model HyDE (Luyu Gao)

Keunggulan utama HyDE terletak pada kemampuannya memperkaya query dengan konteks semantik yang lebih luas. Hal ini sangat berguna dalam domain akademik, hukum, medis, dan riset, di mana pertanyaan pengguna sering kali terlalu umum. Dengan HyDE, pencarian menjadi lebih tepat sasaran karena dokumen hipotetis berperan sebagai “jembatan” antara query mentah dan dokumen nyata.

Meski demikian, terdapat keterbatasan yang perlu dicatat. Dokumen hipotetis yang dihasilkan model tidak selalu akurat secara faktual, karena pada dasarnya ia merupakan “prediksi” atau konstruksi awal. Oleh karena itu, HyDE tidak boleh dianggap sebagai jawaban final, melainkan sebagai strategi untuk meningkatkan kualitas retrieval. Evaluasi manual atau validasi dengan sumber tepercaya tetap diperlukan untuk menjamin akurasi hasil.

Secara metodologis, alur HyDE dapat diringkas sebagai berikut:

- Pengguna mengajukan pertanyaan atau instruksi.
- Model menghasilkan dokumen hipotetis yang relevan dengan pertanyaan.
- Dokumen hipotetis tersebut diubah menjadi embedding.
- Embedding dibandingkan dengan basis data untuk menemukan dokumen serupa.
- Dokumen yang paling relevan diambil, kemudian digunakan untuk menyusun jawaban final.

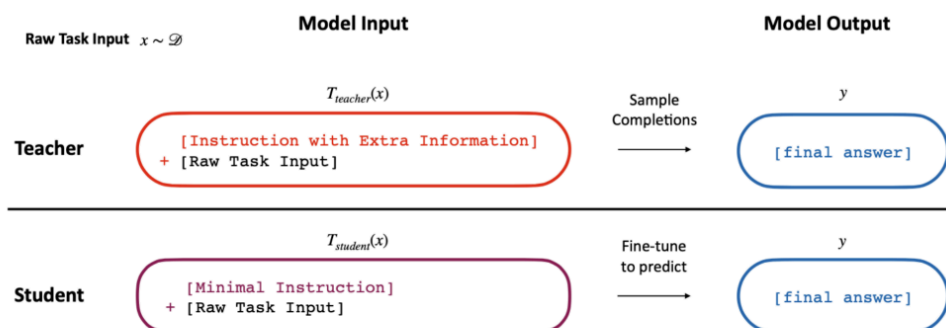
Dengan demikian, HyDE dapat dipandang sebagai teknik yang memperluas cakupan kemampuan retrieval-augmented generation dengan cara memperkaya

query awal, sehingga menghasilkan jawaban yang lebih relevan, terarah, dan bernuansa kontekstual.

4.13 Context Distillation Prompting

Context Distillation Prompting merupakan teknik prompt engineering yang berfokus pada penyaringan (distillation) informasi penting dari teks atau dokumen yang panjang sebelum digunakan untuk menjawab pertanyaan inti. Alih-alih memberikan keseluruhan dokumen kepada model, pengguna menginstruksikan model untuk terlebih dahulu mengekstrak poin-poin kunci, ringkasan, atau representasi singkat. Hasil penyaringan inilah yang kemudian dijadikan dasar untuk menghasilkan jawaban akhir.

Pendekatan ini berangkat dari keterbatasan model bahasa terhadap kapasitas jendela konteks (context window) serta kecenderungan model untuk kehilangan fokus ketika dipaksa membaca teks yang terlalu panjang. Dengan melakukan distilasi konteks, beban model dalam memproses informasi berkurang, sekaligus meningkatkan ketepatan dan relevansi jawaban.



Cara Kerja Context Distillation Prompting (Andrew H Lee)

Sebagai contoh, ketika diberikan sebuah laporan penelitian setebal 50 halaman, pengguna dapat menyusun prompt seperti: “Ringkas dokumen berikut menjadi poin-poin utama yang berkaitan dengan dampak sosial media terhadap kesehatan mental remaja. Setelah itu, gunakan ringkasan tersebut untuk menjawab pertanyaan: bagaimana media sosial memengaruhi kualitas tidur?” Melalui langkah ini, model lebih terarah karena hanya bekerja pada informasi yang telah dipadatkan.

Keunggulan utama context distillation prompting adalah efisiensi. Dengan konteks yang telah disaring, risiko terjadinya hallucination atau jawaban yang

menyimpang dapat dikurangi. Teknik ini juga meningkatkan transparansi, karena proses jawaban didahului dengan ringkasan yang dapat ditinjau kembali oleh pengguna.

Namun demikian, terdapat keterbatasan. Distilasi yang terlalu agresif dapat menghilangkan detail penting, sehingga jawaban akhir kehilangan kedalaman analisis. Selain itu, kualitas distilasi sangat bergantung pada instruksi yang diberikan: jika ringkasan tidak diarahkan dengan tepat, model mungkin tetap mempertahankan bagian yang tidak relevan atau justru mengabaikan informasi kunci.

Secara metodologis, alur context distillation prompting dapat digambarkan sebagai berikut:

- Input awal: dokumen panjang atau kumpulan teks.
- Distilasi konteks: model diminta menghasilkan ringkasan, poin utama, atau representasi singkat.
- Penggunaan hasil distilasi: ringkasan dipakai sebagai input tambahan untuk menjawab pertanyaan inti.
- Jawaban akhir: model menghasilkan respons yang lebih fokus dan relevan berdasarkan konteks terdistilasi.

Dalam praktik akademik maupun profesional, teknik ini sangat bermanfaat untuk analisis literatur, penyusunan ringkasan hukum, pengolahan data riset, maupun penyaringan laporan bisnis. Dengan demikian, context distillation prompting dapat dipandang sebagai strategi penting untuk menjaga keseimbangan antara keluasan informasi dan ketepatan jawaban dalam interaksi dengan LLM.

4.14 Persona/Character Prompting

Persona atau Character Prompting merupakan pengembangan dari role prompting yang lebih mendalam, di mana model tidak hanya diarahkan untuk berperan dalam suatu profesi atau fungsi, melainkan juga diminta untuk meniru karakteristik kepribadian, gaya komunikasi, dan sudut pandang tertentu secara konsisten. Dengan demikian, keluaran model tidak hanya menyesuaikan konten, tetapi juga menampilkan nuansa yang menyerupai persona yang ditetapkan.

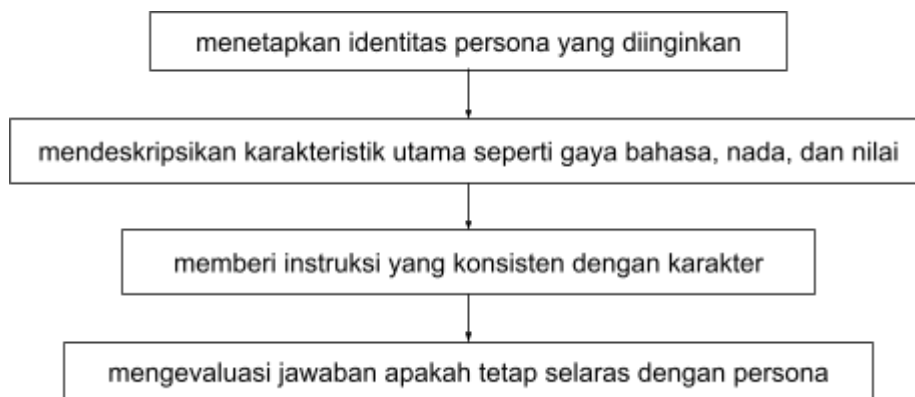
Secara konseptual, teknik ini memungkinkan pengguna membangun interaksi yang lebih imersif dengan LLM. Instruksi dapat mencakup dimensi persona seperti latar belakang, nilai, preferensi, atau bahkan gaya bahasa khas. Sebagai contoh, perintah “Bertindaklah sebagai filsuf Yunani kuno yang gemar menggunakan analogi alam dalam menjelaskan konsep etika” akan menghasilkan

respons yang berbeda dibanding instruksi “Bertindaklah sebagai jurnalis teknologi dengan gaya ringkas dan analitis.”

Manfaat utama persona prompting adalah konsistensi gaya. Dalam konteks pendidikan, misalnya, seorang guru dapat meminta model untuk berperan sebagai tutor sabar dengan gaya komunikasi sederhana. Dalam dunia bisnis, perusahaan dapat membentuk persona asisten virtual yang selalu ramah, sopan, dan berorientasi pada solusi. Sementara itu, dalam bidang kreatif, persona dapat berupa tokoh fiksi atau gaya literer tertentu yang memperkaya variasi narasi.

Meski memiliki keunggulan, teknik ini juga menghadapi keterbatasan. Pertama, semakin kompleks persona yang ditetapkan, semakin besar kemungkinan model gagal mempertahankan konsistensi, terutama dalam percakapan panjang. Kedua, penggunaan persona tertentu (misalnya meniru gaya tokoh nyata) dapat menimbulkan implikasi etika dan hukum, khususnya terkait hak cipta, privasi, atau potensi penyalahgunaan identitas. Oleh karena itu, pengguna perlu berhati-hati dalam merancang persona dan memastikan kesesuaiannya dengan prinsip etis.

Alur penerapan persona/character prompting umumnya mencakup empat langkah: (1) menetapkan identitas persona yang diinginkan, (2) mendeskripsikan karakteristik utama seperti gaya bahasa, nada, dan nilai, (3) memberi instruksi yang konsisten dengan karakter tersebut, dan (4) mengevaluasi jawaban apakah tetap selaras dengan persona.



Alur Persona/Character Prompting

Dengan demikian, persona/character prompting dapat dipandang sebagai strategi untuk memperkaya interaksi dengan LLM, sekaligus memberikan fleksibilitas dalam menciptakan gaya komunikasi yang lebih sesuai dengan kebutuhan pengguna, baik dalam pendidikan, bisnis, penelitian, maupun hiburan.

4.15 Multi-Agent Prompting

Multi-Agent Prompting merupakan teknik lanjutan dalam prompt engineering yang melibatkan lebih dari satu agen kecerdasan buatan (AI agent) untuk berinteraksi, berdiskusi, atau berkolaborasi dalam menyelesaikan suatu tugas. Alih-alih mengandalkan satu model yang bekerja sendiri, pendekatan ini meniru dinamika kerja tim, di mana setiap agen memiliki peran, keahlian, atau sudut pandang yang berbeda.

Secara konseptual, teknik ini dapat dipahami sebagai upaya memperkaya proses penalaran dengan menggabungkan “suara” dari berbagai agen. Satu agen dapat berperan sebagai pengusul solusi, agen lain sebagai pengkritik atau peninjau, sementara agen ketiga berfungsi sebagai penengah yang menyintesis pandangan. Dengan cara ini, model tidak hanya menghasilkan jawaban tunggal, tetapi juga menampilkan dialektika yang lebih dalam sebelum menyajikan kesimpulan akhir.

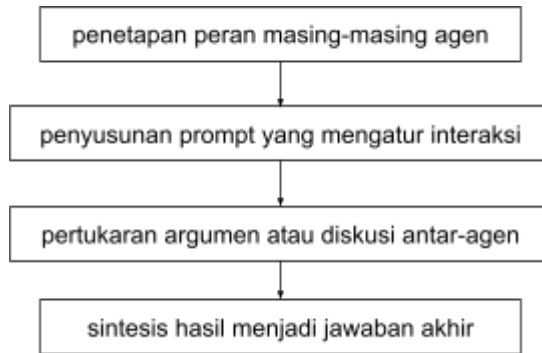
Contoh penerapan dapat dilihat pada konteks analisis kebijakan publik. Agen pertama dapat berperan sebagai ekonom yang mengusulkan kebijakan fiskal, agen kedua sebagai sosiolog yang mengkritisi dampak sosial, dan agen ketiga sebagai pembuat keputusan yang menimbang kedua argumen untuk menghasilkan rekomendasi yang seimbang. Dalam bidang hukum, multi-agent prompting dapat digunakan untuk menyimulasikan perdebatan antara jaksa dan pembela, dengan seorang “hakim” AI yang menyimpulkan hasil diskusi.

Keunggulan utama teknik ini adalah kemampuannya menghasilkan jawaban yang lebih kaya perspektif. Dengan adanya interaksi antar-agen, potensi bias satu arah dapat diminimalkan, dan pengguna memperoleh pandangan yang lebih beragam. Selain itu, multi-agent prompting juga dapat memperkuat kreativitas, misalnya dalam perancangan ide bisnis atau karya kreatif.

Namun, keterbatasannya juga signifikan. Diskusi antar-agen dapat menghasilkan keluaran yang panjang dan kompleks, sehingga memerlukan evaluasi tambahan untuk menyaring hasil yang relevan. Selain itu, karena tetap bergantung pada satu atau beberapa model dasar, risiko hallucination tidak sepenuhnya terhindarkan. Dari sisi teknis, penerapan multi-agent prompting sering membutuhkan integrasi sistem tambahan untuk mengatur alur komunikasi antar-agen.

Secara metodologis, alur umum teknik ini dapat diringkas menjadi: (1) penetapan peran masing-masing agen, (2) penyusunan prompt yang mengatur

interaksi, (3) pertukaran argumen atau diskusi antar-agen, dan (4) sintesis hasil menjadi jawaban akhir.



Alur Multi-Agent Prompting

Dengan demikian, multi-agent prompting dapat dipandang sebagai refleksi dari kolaborasi manusia dalam pengambilan keputusan: bukan sekadar satu suara, tetapi hasil dialog yang memperhitungkan berbagai sudut pandang. Potensi teknik ini semakin relevan seiring arah perkembangan AI menuju agentic AI, yakni sistem yang tidak hanya merespons, tetapi juga berperilaku layaknya entitas otonom yang mampu berkoordinasi.

BAB V

PROMPT UNTUK TEKS DAN BAHASA

Bahasa adalah pintu utama yang membuka ruang interaksi antara manusia dan kecerdasan buatan. Dari menulis artikel populer, menyusun cerita pendek, hingga menganalisis opini publik, model bahasa besar dapat menjadi mitra yang membantu mempercepat proses berpikir dan memperkaya imajinasi. Dengan instruksi yang dirancang secara tepat, mesin mampu menghasilkan teks yang bervariasi: kadang ringkas dan faktual, kadang naratif dan puitis, bahkan teknis seperti kode program atau query basis data.

Kemampuan ini juga menjangkau dunia pendidikan, di mana AI dapat menyusun soal latihan, memberikan penjelasan interaktif, atau mensimulasikan percakapan belajar yang menyerupai tutor pribadi. Dalam penelitian dan bisnis, AI digunakan untuk menganalisis laporan panjang, menyaring data yang kompleks, dan bahkan menyarankan bentuk visualisasi yang paling tepat. Semua ini menunjukkan bahwa teks bukan hanya sekadar kata-kata, melainkan medium utama untuk memanfaatkan potensi kecerdasan buatan secara maksimal.

5.1 Prompt untuk Penulisan Kreatif

Model bahasa besar tidak hanya unggul dalam menjawab pertanyaan faktual, tetapi juga dapat berfungsi sebagai mitra kreatif. Dengan instruksi yang tepat, model mampu menghasilkan karya tulis yang imajinatif: mulai dari artikel populer dengan gaya komunikatif, cerita pendek yang memikat, hingga puisi dengan nuansa emosional tertentu. Kreativitas ini muncul karena model dilatih dengan beragam gaya bahasa, sehingga dapat meniru pola penulisan yang diinginkan pengguna.

Dalam praktiknya, penulisan kreatif dengan bantuan AI mencakup berbagai bentuk. Artikel populer ilmiah, misalnya, dapat ditulis dengan gaya ringan agar mudah dipahami khalayak umum. Cerita pendek dan skenario dapat dikembangkan dengan cepat untuk eksplorasi ide, sementara puisi dan naskah drama dapat dihasilkan untuk tujuan estetis atau hiburan. Fleksibilitas ini membuat model berperan bukan sebagai pengganti penulis, melainkan sebagai kolaborator yang membantu memperluas kemungkinan ekspresi.

Beberapa teknik prompting sangat relevan digunakan untuk konteks kreatif. Role prompting dapat mengarahkan model agar menulis seolah-olah seorang penulis tertentu, misalnya “bertindaklah sebagai penyair romantis abad ke-19.” Dengan few-shot prompting, pengguna dapat memberikan contoh gaya tulis agar

model meniru pola yang sama. Sementara itu, *contrastive prompting* memungkinkan pengguna menunjukkan perbedaan antara contoh yang diinginkan dan yang harus dihindari, sehingga hasil lebih sesuai dengan selera.

Sebagai ilustrasi, pengguna dapat meminta model menulis artikel populer tentang energi terbarukan dengan nada optimistis dan komunikatif. Di lain kesempatan, instruksi bisa diarahkan untuk menghasilkan puisi cinta modern dengan gaya minimalis. Dalam kedua kasus ini, yang membedakan hasil adalah kejelasan instruksi dan teknik yang digunakan.

Untuk mempermudah pemahaman, berikut disajikan contoh template prompt yang dapat digunakan dalam penulisan kreatif.

Contoh Prompt Kreatif

Jenis Karya	Contoh Prompt	Hasil yang Diharapkan
Caption Instagram	“Buat caption singkat untuk promosi minuman sehat, gunakan gaya ceria, sertakan emoji buah.”	Teks singkat penuh energi, mudah diingat, dan sesuai budaya media sosial.
Slogan iklan	“Tulis slogan untuk aplikasi belajar online yang memberi kesan inspiratif.”	Kalimat pendek yang menekankan motivasi, mudah dipakai sebagai tagline.
Copywriting produk	“Susun deskripsi produk kopi premium untuk katalog, dengan nada elegan dan meyakinkan.”	Paragraf singkat, elegan, menonjolkan kualitas rasa dan aroma.
Headline kampanye	“Buat headline untuk kampanye ramah lingkungan, dengan nada positif dan penuh semangat.”	Kalimat persuasif yang mampu menarik perhatian publik.

Dengan pendekatan seperti ini, penulisan kreatif tidak lagi sekadar hasil spontanitas manusia, melainkan menjadi proses kolaboratif antara ide penulis dan kemampuan model bahasa dalam meniru pola, gaya, dan imajinasi. Kolaborasi ini dapat mempercepat proses eksplorasi kreatif, sekaligus memberi ruang bagi manusia untuk tetap menjadi kurator ide, rasa, dan estetika.

5.2 Prompt untuk Analisis Teks

Selain menghasilkan teks baru, model bahasa besar juga berperan penting dalam memahami, menafsirkan, dan menyederhanakan teks yang sudah ada. Melalui rancangan instruksi yang tepat, model dapat digunakan sebagai analis linguistik yang mampu membaca opini, mengidentifikasi sentimen, menyaring informasi penting, hingga merangkum dokumen yang panjang. Dengan cara ini, LLM tidak hanya berfungsi sebagai generator bahasa, tetapi juga sebagai alat analisis yang memperkuat pemahaman manusia terhadap arus informasi.

Analisis teks memiliki beragam aplikasi. Dalam konteks opini publik, model dapat membantu memahami bagaimana masyarakat merespons isu tertentu, misalnya dengan menilai kecenderungan sentimen positif atau negatif dari unggahan media sosial. Dalam kajian bisnis, model dapat digunakan untuk menganalisis ulasan produk secara massal guna menemukan pola kepuasan maupun keluhan pelanggan. Di bidang akademik dan pemerintahan, LLM dapat dimanfaatkan untuk merangkum dokumen panjang, seperti laporan penelitian atau draf kebijakan, sehingga pembaca dapat memperoleh gambaran cepat tanpa harus membaca seluruh isi.

Beberapa teknik prompting mendukung keandalan analisis teks. Context distillation prompting memungkinkan model menyaring dokumen panjang menjadi inti-inti penting sebelum menjawab pertanyaan. Step-back prompting bermanfaat agar model tidak hanya langsung memberi klasifikasi, tetapi juga menjelaskan prinsip analisis yang mendasarinya. Sementara itu, chain-of-thought prompting mendorong model untuk menuliskan alasan langkah demi langkah mengapa suatu teks dianggap positif, negatif, atau netral.

Sebagai contoh, berita politik yang panjang dapat diringkas menjadi poin-poin utama dengan instruksi sederhana seperti: “Ringkas artikel berikut menjadi tiga poin yang menyoroti kebijakan utama dan dampaknya.” Untuk analisis sentimen, ulasan produk dapat diproses dengan prompt: “Kategorikan ulasan berikut sebagai positif, netral, atau negatif, dan jelaskan alasannya dalam dua kalimat.” Melalui cara ini, LLM berfungsi sebagai pembaca kritis yang mampu menghubungkan data tekstual dengan interpretasi makna.

Contoh Template Prompt untuk Analisis Teks

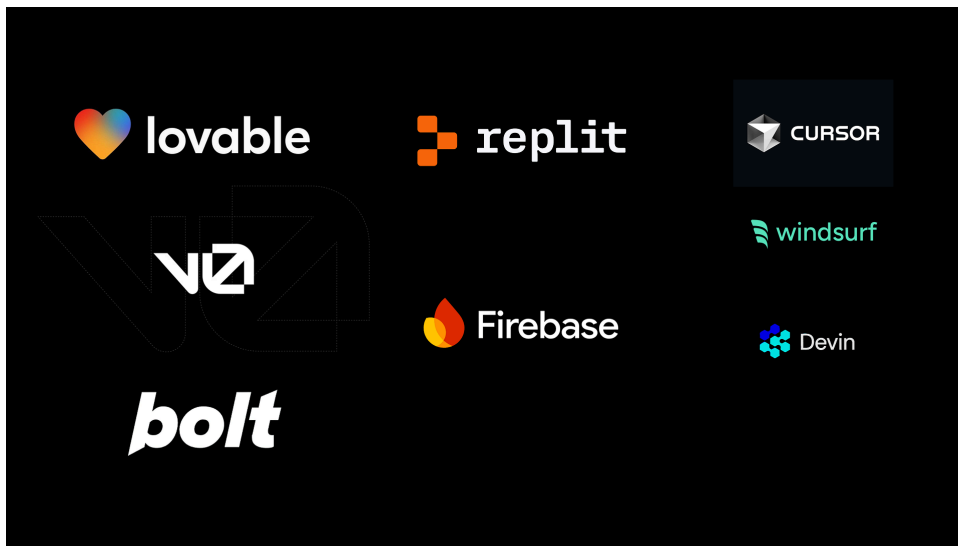
Jenis Analisis	Contoh Prompt	Keluaran yang Diharapkan
----------------	---------------	--------------------------

Ringkasan berita	“Ringkas artikel berikut menjadi tiga poin utama yang mudah dipahami orang awam.”	Ringkasan singkat berupa poin-poin inti berita tanpa detail berlebihan.
Analisis opini	“Apakah pernyataan berikut menunjukkan opini atau fakta? Jelaskan secara singkat.”	Identifikasi apakah teks berupa opini atau fakta, beserta alasan singkat.
Klasifikasi sentimen	“Kategorikan ulasan berikut sebagai positif, netral, atau negatif, lalu jelaskan alasannya.”	Label sentimen (positif/netral/negatif) dengan alasan yang logis.
Ekstraksi informasi	“Dari artikel berikut, sebutkan tiga tokoh penting dan perannya masing-masing.”	Daftar tokoh dengan peran sesuai konteks artikel.
Ringkasan dokumen panjang	“Buat ringkasan eksekutif sepanjang 200 kata dari dokumen berikut, fokus pada rekomendasi kebijakan.”	Ringkasan komprehensif namun padat, menyoroti inti kebijakan.

Dengan demikian, prompt engineering untuk analisis teks memungkinkan LLM berperan sebagai asisten analitik yang mampu menafsirkan makna, menilai emosi, dan menyusun ringkasan yang relevan. Teknik ini membuka peluang besar bagi dunia akademik, bisnis, maupun pemerintahan untuk mengolah informasi dalam skala yang semakin luas, dengan kecepatan dan efisiensi yang sulit dicapai oleh pembacaan manual semata.

5.3 Prompt untuk Coding

Model bahasa besar semakin efektif digunakan sebagai asisten pemrograman. Perannya meliputi menghasilkan kode dari spesifikasi singkat, menjelaskan potongan kode yang sulit, menyarankan perbaikan atau optimasi, hingga menyusun kueri basis data dan regular expression. Manfaat utama muncul ketika instruksi dirumuskan jelas—mencakup tujuan, batasan, lingkungan eksekusi, serta contoh input–output. Kualitas hasil akan meningkat ketika proses dilakukan secara iteratif: meminta draf awal, menjalankan uji sederhana, lalu mengarahkan model untuk memperbaiki bagian yang gagal.



Platform AI yang Dibuat Khusus untuk Coding

Jenis penggunaan umum mencakup penulisan fungsi-fungsi utilitas (Python/JavaScript), pembuatan komponen UI ringkas (JS/TS), penyusunan query agregasi dan window function (SQL), serta perancangan regex yang dapat diuji dengan contoh positif-negatif. Untuk menjaga reliabilitas, hasil AI sebaiknya diverifikasi melalui pengujian unit, linting, dan static analysis. Pada kasus lebih kompleks, pendekatan multi-agent prompting dapat dipakai: satu “agen” menghasilkan kode, agen lain bertindak sebagai reviewer yang memeriksa kompleksitas, keamanan, dan kinerja.

Template Prompt Coding dan Hasil yang Diharapkan

Tujuan	Contoh Prompt (ringkas & dapat langsung dipakai)	Hasil yang Diharapkan
Python – fungsi rerata aman	“Bertindaklah sebagai software engineer senior. Tulis fungsi Python <code>safe_mean(nums: List[float]) -> float</code>	None yang mengembalikan rerata atau None` jika input kosong/berisi non-numerik. Sertakan docstring dan contoh uji singkat (doctest). Kompleksitas O(n).”

Python – penjelasan kode	“Jelaskan baris demi baris kode Python berikut, fokus pada kompleksitas waktu dan potensi bug: <code>python ...</code> . Berikan saran perbaikan singkat.”	Uraian teknis yang jelas, analisis kompleksitas, dan rekomendasi refaktor.
JavaScript – utilitas format tanggal	“Tulis fungsi JS <code>formatDate(iso, locale)</code> yang mengubah ISO string ke format lokal dengan <code>Intl.DateTimeFormat</code> . Sertakan contoh pemakaian untuk <code>id-ID</code> dan <code>en-US</code> serta <i>fallback</i> jika input tidak valid.”	Fungsi JS portabel, contoh pemakaian, dan penanganan input tidak valid.
SQL – laporan penjualan bulanan	“Buat <i>query</i> SQL untuk menghitung total penjualan per produk per bulan (YYYY-MM), dari tabel <code>orders(order_date, product_id, qty, price)</code> . Tampilkan <code>month, product_id, revenue</code> , diurutkan <code>month ASC, revenue DESC</code> . Gunakan fungsi tanggal yang umum (PostgreSQL).”	<i>Query</i> dengan <code>DATE_TRUNC('month', order_date)</code> , agregasi <code>SUM(qty*price)</code> , <i>alias</i> kolom yang jelas, urutan sesuai permintaan.
SQL – deteksi top 5 produk	“Tulis <i>query</i> untuk mengambil 5 produk dengan <code>revenue</code> tertinggi selama 90 hari terakhir. Sertakan <i>CTE</i> untuk keterbacaan.”	<i>CTE</i> periode waktu, agregasi, <code>ORDER BY DESC, LIMIT 5</code> .
Regex – validasi email	“Rancang <i>regex</i> untuk validasi email umum (huruf, angka, titik, minus, underscore; subdomain opsional). Berikan 5 contoh yang cocok dan 5 contoh yang tidak, serta catatan keterbatasan.”	Pola <i>regex</i> yang wajar, daftar contoh <i>match/non-match</i> , dan catatan bahwa validasi penuh sebaiknya di sisi server.
Code review otomatis (multi-agen t)	“Agen A: tulis fungsi JS untuk <i>debounce</i> . Agen B: ulas keamanan dan kompatibilitas browser, sarankan perbaikan. Tampilkan versi final setelah revisi.”	Versi awal → ulasan terstruktur → revisi akhir dengan catatan desain.

Debugging iteratif	“Kode Python berikut gagal pada input <code>[]</code> dan <code>[\"3\", 4]</code> . Identifikasi penyebab, tulis <i>unit test</i> yang mereplikasi bug, perbaiki fungsi, dan tunjukkan hasil uji lulus.”	<i>Unit test</i> yang gagal → perbaiki kode → <i>test</i> lulus, disertai penjelasan.
---------------------------	---	---

Ketika menggunakan LLM untuk membantu pengembangan aplikasi, kualitas hasil sangat dipengaruhi oleh kejelasan instruksi. Prompt yang samar biasanya hanya menghasilkan potongan kode sederhana, sementara prompt yang lengkap dapat membantu menyusun struktur aplikasi yang lebih komprehensif. Berikut beberapa tips yang relevan:

1. Jelaskan kerangka aplikasi sejak awal
Sebutkan teknologi yang digunakan (misalnya Flask untuk Python, Express untuk Node.js, atau React untuk front-end). Instruksi yang jelas akan mengarahkan model untuk menghasilkan kode sesuai ekosistem yang dipilih.

Contoh:

“Buatkan skeleton aplikasi web dengan Flask yang memiliki endpoint `/hello` dan mengembalikan teks JSON `{message: 'Hello World'}`.”

2. Spesifikasikan fitur dan alur kerja
Alih-alih meminta “buatkan aplikasi e-commerce”, pecah menjadi fitur kecil: otentikasi pengguna, keranjang belanja, pembayaran. Prompt berantai (prompt chaining) membantu membangun modul demi modul.

Contoh:

“Tuliskan kode otentikasi sederhana untuk aplikasi e-commerce dengan Express.js, termasuk register, login, dan validasi password.”

3. Cantumkan skema data atau model yang jelas
Sertakan definisi tabel, tipe data, atau model ORM. Hal ini penting untuk aplikasi berbasis database agar query atau migrasi konsisten.

Contoh:

“Buatkan model Sequelize untuk tabel Users dengan kolom `id`, `username`, `email`, `password_hash`.”

4. Minta dokumentasi dan komentar dalam kode
Tambahkan instruksi agar model menghasilkan komentar di dalam kode atau dokumentasi singkat di luar kode. Hal ini memudahkan pemeliharaan.

Contoh:

“Tuliskan API dengan FastAPI untuk CRUD data Book. Sertakan komentar pada tiap fungsi dan docstring untuk model.”

5. Gunakan iterasi untuk debugging dan penyempurnaan
Setelah kode awal dihasilkan, uji secara manual, lalu berikan prompt lanjutan untuk memperbaiki kesalahan atau menambahkan fitur baru.

Contoh:

“Kode berikut gagal saat input kosong. Tolong debug, tambahkan validasi, dan sertakan unit test dengan pytest.”

6. Simulasikan kolaborasi dengan multi-agent prompting
Minta satu agen menulis kode, lalu agen lain bertindak sebagai reviewer untuk menilai keamanan, efisiensi, atau keterbacaan.

Contoh:

“Agen A: tuliskan endpoint untuk upload file dengan Django Rest Framework. Agen B: ulas potensi risiko keamanan, lalu revisi kode agar lebih aman.”

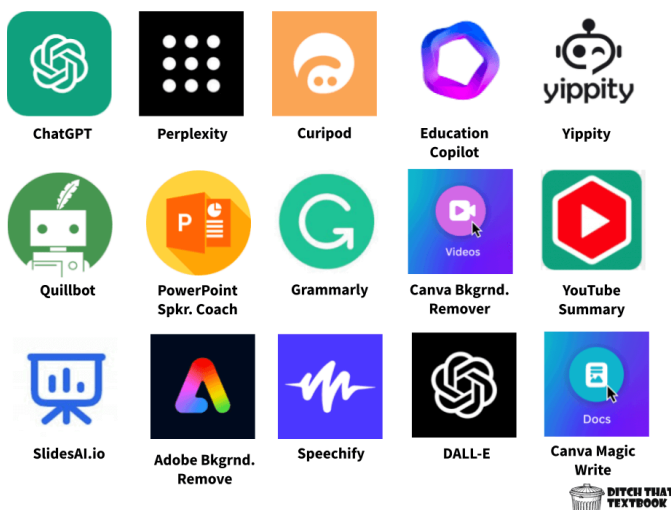
5.4 Prompt untuk Pendidikan dan Soal Interaktif

Model bahasa besar dapat dimanfaatkan sebagai asisten belajar yang adaptif. Dengan instruksi yang tepat, AI tidak hanya berfungsi sebagai penyaji materi, tetapi juga sebagai pengajar yang mampu membuat soal, memberikan penjelasan, bahkan mensimulasikan dialog interaktif. Pendekatan ini memungkinkan pembelajaran menjadi lebih personal, di mana siswa memperoleh materi sesuai kebutuhan, sementara guru atau dosen dapat terbantu dalam menyusun bahan ajar dan evaluasi.

Jenis penggunaan dalam pendidikan sangat beragam. AI dapat membuat soal pilihan ganda, esai, atau kuis singkat untuk latihan. Ia juga dapat menjelaskan konsep sulit dengan analogi sederhana agar mudah dipahami, misalnya menjelaskan hukum Newton dengan perumpamaan keseharian. Tidak hanya itu,

AI juga dapat menyusun slide presentation, draf makalah di Word, atau bahkan merancang struktur laporan sesuai format akademik tertentu. Dengan cara ini, teknologi LLM berfungsi sebagai mitra pedagogis sekaligus alat produktivitas.

15 AI tools to use in the classroom



Alat AI di Bidang Pendidikan (Flame)

Teknik prompting yang relevan antara lain persona prompting dengan meminta AI berperan sebagai guru yang sabar, tutor khusus bidang tertentu, atau dosen yang menyesuaikan penjelasan untuk tingkat pendidikan tertentu. Selain itu, prompt chaining dapat digunakan untuk membuat alur lengkap: AI menyusun soal, lalu memberikan kunci jawaban, dan akhirnya menambahkan penjelasan yang ringkas namun informatif.

Sebagai contoh, untuk mata pelajaran fisika SMA, model dapat diminta membuat soal tentang hukum gravitasi. Setelah menghasilkan soal, AI juga memberikan jawaban yang benar, diikuti penjelasan singkat tentang prinsip yang mendasari. Contoh lain adalah kuis hukum tata negara di tingkat universitas, di mana AI membuat pertanyaan pilihan ganda, menyediakan kunci jawaban, dan memberikan alasan normatif mengapa jawaban tersebut benar.

Contoh Template Prompt untuk Pendidikan dan Soal Interaktif

Jenis Tugas	Contoh Prompt	Soal/Instruksi	Jawaban	Feedback/Penjelasan
Soal pilihan ganda (Fisika SMA)	“Buat soal pilihan ganda tentang hukum gravitasi Newton untuk siswa SMA. Sertakan 4 opsi, kunci jawaban, dan penjelasan singkat.”	“Benda A massanya 2 kg, benda B massanya 4 kg. Jarak keduanya 3 m. Menurut hukum gravitasi Newton, bagaimana gaya tarik-menarik antara keduanya?”	Jawaban benar: C (nilai gaya)	Penjelasan: Rumus hukum gravitasi menunjukkan hubungan langsung dengan massa dan berbanding terbalik dengan kuadrat jarak.
Kuis hukum tata negara (Mahasiswa)	“Susun 2 soal pilihan ganda tentang judicial review di Indonesia. Sertakan jawaban benar dan penjelasan normatif.”	Soal 1: “Lembaga yang berwenang menguji undang-undang terhadap UUD 1945 adalah....”	Jawaban: A (Mahkamah Konstitusi)	Penjelasan: MK memiliki kewenangan berdasarkan Pasal 24C UUD 1945.
Soal esai	“Tulis satu soal esai tentang dampak globalisasi ekonomi, lalu berikan poin kunci jawaban.”	“Jelaskan dampak globalisasi ekonomi terhadap UMKM di Indonesia.”	Jawaban kunci: integrasi pasar global, persaingan meningkat, peluang ekspor, kebutuhan inovasi	Feedback: Mahasiswa diharapkan menjelaskan peluang dan tantangan UMKM dalam konteks global.

Penjelasan analogi	“Jelaskan konsep hukum permintaan dalam ekonomi dengan analogi sederhana untuk siswa SMA.”	Instruksi: “Hubungkan hukum permintaan dengan kehidupan sehari-hari.”	Jawaban: “Semakin mahal harga es krim, semakin sedikit orang yang membelinya.”	Feedback: Analogi ini membantu siswa memahami hubungan negatif antara harga dan jumlah permintaan.
Pembuatan presentasi	“Susun kerangka 5 slide presentasi PowerPoint tentang energi terbarukan, sertakan poin utama per slide.”	Slide 1: Definisi energi terbarukan. Slide 2: Jenis-jenis (matahari, angin, air). Slide 3: Manfaat. Slide 4: Tantangan. Slide 5: Kesimpulan.	Jawaban: Kerangka isi untuk setiap slide.	Feedback: Kerangka dapat langsung digunakan untuk membuat presentasi visual di PowerPoint atau Google Slides.

Selain berfungsi sebagai alat untuk membuat soal atau ringkasan materi, model bahasa besar juga dapat berperan sebagai mitra belajar interaktif maupun guru daring. Dalam perspektif pedagogis, LLM mampu menyesuaikan tingkat penjelasan dengan kebutuhan peserta didik, menjawab pertanyaan secara adaptif, serta menyediakan latihan kontekstual yang menyerupai interaksi langsung dengan pengajar manusia.

Sebagai “guru daring”, LLM dapat memberikan uraian teoretis, contoh soal, serta umpan balik secara real-time. Sementara itu, dalam posisi sebagai “teman belajar”, model dapat diajak berdiskusi dalam bentuk percakapan yang lebih santai, namun tetap informatif. Kedua peran ini melengkapi pendekatan pembelajaran formal dengan pengalaman belajar mandiri yang lebih personal.

Contoh penerapan yang populer adalah pembelajaran bahasa asing, seperti bahasa Inggris. Mahasiswa dapat berinteraksi dengan LLM dalam bentuk percakapan bilingual, meminta koreksi tata bahasa, hingga memperoleh penjelasan gramatikal yang biasanya membutuhkan waktu panjang dalam kelas tradisional. LLM bahkan dapat diminta untuk mensimulasikan percakapan sehari-hari, seperti di restoran, bandara, atau kelas akademik, lengkap dengan catatan kosakata dan tata bahasa.

5.5 Prompt untuk Analisis Data & Visualisasi

Model bahasa besar dapat digunakan bukan hanya untuk menghasilkan teks, tetapi juga untuk membantu analisis data dan pembuatan visualisasi. Melalui prompt yang tepat, AI mampu menafsirkan tabel, menyusun ringkasan dataset, memberi rekomendasi grafik, bahkan menuliskan instruksi kode untuk Python, SQL, atau tool visualisasi seperti Mermaid.js. Dengan demikian, LLM berfungsi sebagai asisten analitik yang menjembatani antara data mentah dan interpretasi yang dapat dipahami.

Konsep dasar dari pendekatan ini adalah memanfaatkan kemampuan model dalam memahami pola teks yang merepresentasikan data. Data dalam bentuk tabel, daftar, atau hasil survei dapat dimasukkan ke dalam prompt, kemudian AI diminta untuk mengidentifikasi tren, hubungan variabel, atau anomali. Dari sana, AI juga dapat menyarankan cara paling efektif untuk menyajikan informasi dalam bentuk grafik.

Jenis penggunaan meliputi ringkasan hasil survei, analisis tren penjualan bulanan, rekomendasi jenis grafik (misalnya batang, garis, lingkaran), hingga pembuatan kode program untuk memvisualisasikan data. LLM juga dapat diminta untuk menghasilkan flowchart atau diagram relasi menggunakan markup language seperti Mermaid.js, sehingga kolaborasi dengan tool eksternal menjadi lebih praktis.

Teknik prompting yang relevan antara lain chain-of-thought prompting untuk menguraikan proses analisis secara bertahap, iterative prompting untuk memperbaiki atau menyempurnakan visualisasi, serta multimodal prompting ketika data numerik dipadukan dengan elemen grafis.

Sebagai contoh, sebuah survei kepuasan pelanggan dapat dianalisis dengan prompt: “Ringkas hasil survei berikut dalam tiga poin utama, lalu sarankan grafik yang tepat untuk menyajikannya.” Untuk analisis penjualan, pengguna dapat meminta: “Tuliskan kode Python dengan matplotlib untuk membuat grafik garis penjualan bulanan dari data berikut.” Sementara itu, untuk diagram hubungan antar-proses, prompt dapat diarahkan ke format Mermaid.js: “Buatkan diagram alur proses rekrutmen karyawan dalam format Mermaid.js.”

Contoh Template Prompt untuk Analisis Data & Visualisasi

Input Data	Contoh Prompt	Rekomendasi Analisis/Visualisasi
------------	---------------	----------------------------------

Tabel hasil survei kepuasan pelanggan	“Ringkas hasil survei berikut menjadi tiga poin utama, lalu sarankan grafik yang tepat untuk presentasi.”	Ringkasan poin: mayoritas puas dengan layanan, keluhan terkait waktu tunggu, usulan peningkatan pada sistem online. Rekomendasi: grafik batang per kategori.
Data penjualan bulanan (produk A, Jan–Des)	“Tuliskan kode Python dengan matplotlib untuk membuat grafik garis penjualan bulanan dari data berikut.”	Kode Python dengan <code>plt.plot()</code> yang menampilkan tren penjualan per bulan.
Dataset penjualan per wilayah	“Analisis data penjualan per wilayah, lalu sarankan jenis grafik yang sesuai.”	Ringkasan: Wilayah X memiliki penjualan tertinggi. Rekomendasi: diagram batang horizontal untuk perbandingan antar wilayah.
Proses rekrutmen (tahapan: lamaran → seleksi → wawancara → penawaran)	“Buat diagram alur proses rekrutmen karyawan dalam format Mermaid.js.”	Output dalam sintaks Mermaid: <pre>graph TD; A[Lamaran]-->B[Seleksi]; B-->C[Wawancara]; C-->D[Penawaran].</pre>
Dataset nilai ujian mahasiswa	“Hitung rata-rata, nilai tertinggi, dan terendah, lalu sarankan visualisasi yang cocok.”	Analisis numerik dasar. Rekomendasi: histogram untuk distribusi nilai, atau boxplot untuk menggambarkan sebaran.

Dengan cara ini, LLM menjadi lebih dari sekadar alat penjawab teks. Ia berfungsi sebagai fasilitator analisis yang mampu menyederhanakan data mentah, memberi konteks interpretasi, serta mengarahkan pada bentuk visualisasi yang sesuai. Kolaborasi dengan tool lain seperti Mermaid.js atau Python memperluas cakupan penggunaannya, menjadikan AI sebagai mitra yang efektif dalam penelitian, bisnis, maupun pendidikan.

BAB VI

PROMPT MULTIMODAL

Kebutuhan kita dalam membuat suatu materi untuk presentasi, penulisan buku, pembelajaran dan pekerjaan lainnya akan terasa lebih hidup dan lebih menarik jika bisa menggunakan bukan hanya teks tetapi juga gambar, audio, suara, musik dan juga video. Disinilah pentingnya bagaimana kita bisa menggunakan prompt multimodal di aplikasi Artificial Intelligence (AI) yang biasa kita gunakan. Multimodal mengacu pada kemampuan untuk memproses, menafsirkan, dan menghasilkan konten dari beragam jenis media, seperti teks, audio, gambar, dan video. Dalam konteks AI, sistem multimodal dirancang untuk memahami dan mengintegrasikan informasi dari berbagai jenis format, sehingga menghasilkan keluaran yang lebih kontekstual. Kemampuan ini diperlukan untuk mengembangkan antarmuka AI yang berinteraksi dengan dunia dengan cara yang sangat mirip dengan kognisi manusia, yang secara intrinsik mengintegrasikan berbagai masukan sensorik. Misalnya, asisten virtual berbasis AI pada ponsel pintar dapat merespons perintah suara, mengenali objek dalam foto, atau memproses pesan teks. Dengan menggunakan asisten tersebut, seseorang dapat mengambil foto suatu produk, bertanya kepada asisten tersebut, lalu menerima respons lisan atau teks dengan informasi yang relevan, seperti deskripsi lengkap, harga, dan kemungkinan toko untuk membelinya.

Dalam rekayasa cepat, multimodal dapat meningkatkan kekayaan dan akurasi hasil yang dihasilkan AI karena model AI dapat menghasilkan respons yang relevan secara kontekstual. Misalnya, sebuah prompt mungkin menyertakan gambar dan meminta AI untuk menghasilkan deskripsi teks atau analisis kontennya. Kemampuan untuk mensintesis informasi di berbagai jenis media ini membuka kemungkinan baru untuk aplikasi di bidang-bidang seperti pembuatan konten dan analisis data. Rekayasa prompt multimodal dapat sangat berguna dalam bidang-bidang seperti deteksi berita palsu, dengan memungkinkan sistem AI untuk menganalisis, memverifikasi silang informasi, dan mengidentifikasi ketidakonsistenan antara konten yang disajikan dalam berbagai format, seperti mendeteksi gambar atau video yang dimanipulasi, yang menyertai teks yang menyesatkan. Selain itu, dalam layanan kesehatan, prompt multimodal dapat digunakan untuk menganalisis pemeriksaan gambar dan rekam medis pasien secara bersamaan, sehingga memberikan rekomendasi diagnosis yang lebih baik.

6.1. Prompt Gambar (Image Generation)

Prompt gambar AI adalah instruksi teks yang diberikan kepada model AI untuk menghasilkan gambar. Prompt ini bertindak sebagai panduan bagi AI untuk memahami jenis, gaya, dan elemen visual yang diinginkan dalam gambar yang dihasilkan. Semakin detail dan spesifik prompt yang diberikan, semakin akurat dan sesuai dengan harapan hasil gambar yang akan diperoleh.

Apa itu Prompt Gambar AI? Perintah gambar AI biasanya berupa teks yang menginstruksikan model AI generatif, seperti ChatGPT atau Gemini, untuk menghasilkan konten visual. Perintah ini digunakan untuk memberikan informasi agar algoritma AI dapat mereproduksi gambar yang diminta. Petunjuk gambar AI membantu model memahami karakteristik, objek, atau konsep yang diinginkan untuk ditampilkan dalam gambar yang dihasilkan.

Dengan memberikan perintah yang jelas dan spesifik kepada generator AI, pengguna dapat memengaruhi beberapa faktor terkait gambar, yang dapat membantu model menghasilkan hasil yang cukup akurat dan selaras dengan visi kreatif mereka. Seiring AI semakin cerdas dari waktu ke waktu, pengguna kini dapat membuat perintah gambar AI yang sangat detail yang menangkap setiap aspek dari apa yang ingin mereka capai. Ada dua jenis perintah gambar AI yang dapat digunakan untuk menghasilkan gambar:

Prompt Teks AI

Seperti yang telah disebutkan, perintah teks AI dapat berupa kalimat singkat hingga instruksi yang lebih komprehensif dan terperinci. Semakin detail yang kita berikan pada model generatif, semakin akurat gambar yang dihasilkan sesuai keinginan. Perhatikan tabel berikut sebagai contoh.

Aplikasi AI yang digunakan	ChatGPT
URL Aplikasi	https://chatgpt.com/
Prompt simple yang dimasukkan	buat gambar seorang anak kecil yang sedang bermain di taman

Hasil	
-------	---

Sekarang coba kita buat dengan perintah prompt yang lebih detail. Perhatikan kembali tabel berikut:

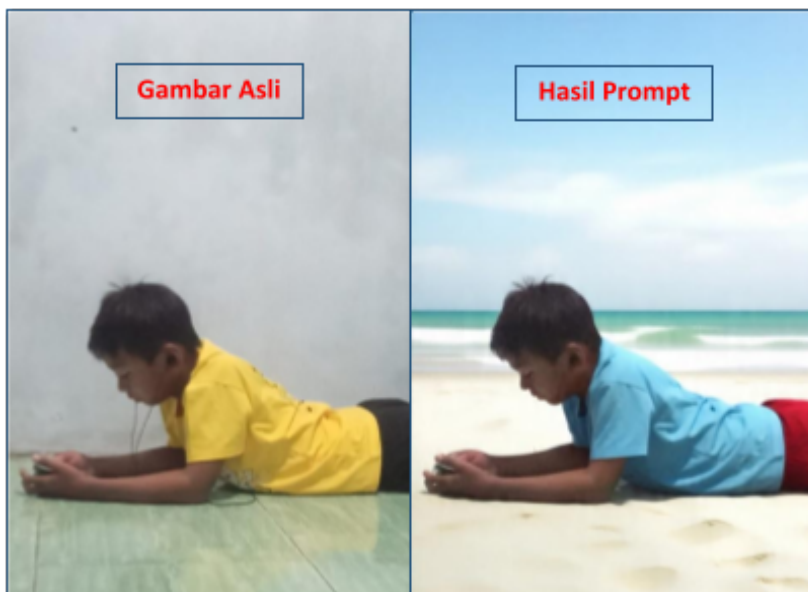
Contoh Prompt Detail

Aplikasi AI yang digunakan	ChatGPT
URL Aplikasi	https://chatgpt.com/
Prompt detail yang dimasukkan	buat gambar seorang anak kecil berambut keriting yang sedang bermain di taman, memakai kaos berwarna biru motif garis dan celana berwarna merah tua dan di tangannya memegang sebuah botol minuman
Hasil	

Seperti yang Anda lihat, ChatGPT mampu memproses kedua perintah tersebut. Dengan kita memasukkan perintah yang lebih detail, maka hasilnya pun akan dapat tergambar dengan baik sesuai yang kita inginkan.

Petunjuk Gambar AI

Bentuk lain dari pembuatan foto AI adalah perintah gambar, di mana kita mengunggah gambar agar model AI dapat bekerja, alih-alih hanya memberikan instruksi tekstual. Teknik ini dapat berguna untuk menyesuaikan foto yang sudah ada atau membuat foto baru berdasarkan gambar asli. Misalnya, kita dapat mengunggah potret seseorang dan menulis perintah: "Ganti warna baju foto ini menjadi biru, celananya warna merah dan backgroundnya sedang berjemur di pantai", seperti yang saya lakukan dengan foto anak saya dengan menggunakan Meta AI dari Handphone.



Contoh Prompt dengan Petunjuk Gambar

Kita juga dapat meminta model AI tertentu untuk memperluas latar belakang gambar yang ada, menambahkan elemen yang dihasilkan AI ke foto, menghasilkan ulang foto dalam definisi yang lebih tinggi, atau mengubah pencahayaan dan warna.

6.2 Prompt Video

Prompt yang dirancang dengan baik adalah kunci untuk menentukan konten video yang dihasilkan oleh model AI, baik membuat video AI dari teks maupun gambar. Dalam buku ini, kami menyajikan formula untuk menulis prompt pembuatan video AI yang dapat membantu anda mencapai hasil video yang optimal. Contoh-contoh prompt yang ada di buku ini, kami praktikkan dengan menggunakan Sora AI <https://sora.chatgpt.com/explore> yang merupakan openio dari ChatGPT. Struktur Prompt Pembuatan Video AI

Teks ke Video

Prompt = Subjek + Aksi + Adegan + (Bahasa Kamera + Pencahayaan + Gaya)

Subjek : Apa atau siapa yang menjadi fokus video? Bisa berupa orang, hewan, tumbuhan, benda, dll. Untuk memperjelas, subjek harus dijelaskan secara detail, mencakup elemen-elemen seperti:

- ❖ Penampilan (misalnya, performa atletik, gaya rambut, pakaian, aksesoris)
- ❖ Fitur wajah, ekspresi, dan emosi
- ❖ Postur tubuh...

Aksi : Apa yang sedang dilakukan subjek? Inilah inti dari prompt Anda, karena ini menggerakkan alur cerita video. Pastikan aksinya jelas dan ringkas, karena prompt ini dirancang untuk video berdurasi 5-10 detik.

Adegan : Di mana aksi berlangsung? Ini mencakup latar depan, latar belakang, dan elemen lain yang membentuk latar.

Bahasa Kamera : Mengacu pada jenis pengambilan gambar, sudut, dan gerakan kamera yang menambah daya tarik narasi dan visual. Gunakan teknik kamera seperti:

- ❖ Close-up, bidikan lebar
- ❖ Pengambilan gambar sudut rendah dan sudut tinggi
- ❖ Pemandangan udara, kedalaman bidang

Pencahayaan : Pencahayaan dalam video dapat memengaruhi suasana dan kedalaman video secara signifikan. Deskripsi pencahayaan harus meningkatkan suasana dan emosi video, seperti cahaya hangat, cahaya pagi, sorotan pada subjek, dan cahaya latar.

Gaya : Menentukan nada dan gaya video. Ini dapat mencakup gaya visual, nada emosional, dan suasana hati secara keseluruhan.

Gambar ke Video

Prompt = Subjek + Aksi + Latar Belakang + Gerakan Latar Belakang

Subjek : Sama seperti dalam Teks-ke-Video, subjek mewakili fokus utama, dan kemunculannya harus dijelaskan secara rinci.

Aksi : Menggambarkan gerakan subjek dalam adegan. Karena ini merupakan perintah gambar-ke-video, Anda dapat menggambarkan gerakan halus yang mengubah gambar diam menjadi video pendek yang dinamis.

Latar Belakang : Menggambarkan lingkungan sekitar, yang dapat membantu menciptakan suasana yang lebih mendalam.

Gerakan Latar Belakang : Mengacu pada elemen dinamis atau pergeseran halus dalam lingkungan yang membantu menghidupkan pemandangan.

Berikut adalah contoh-contoh pengguna prompt untuk video dengan menggunakan aplikasi sora yang bisa di akses di <https://sora.chatgpt.com>

Contoh Teks ke Video terlihat dalam tabel berikut.

No.	Prompt	Hasil
1	Sebuah mobil terbang futuristik terbang di atas cakrawala kota modern saat matahari terbenam. Mobil ini memiliki desain yang ramping dan aerodinamis dengan lampu depan yang menyala dan dikelilingi oleh gedung pencakar langit yang tinggi. Langit dipenuhi dengan cahaya keemasan yang hangat	

2	Bidikan close-up sedang, seekor anjing Pomeranian berbulu berdiri di bangku taman, mengenakan dasi kupu-kupu kotak-kotak dan kemeja kotak-kotak, mengamati sekeliling dengan rasa ingin tahu. Sinar matahari keemasan yang hangat menembus pepohonan, memancarkan cahaya lembut pada bulunya	
3	Tas kurir kulit cokelat yang stylish dengan detail penutup dan gesper, diletakkan di atas permukaan kaca. Tas ini berkilau di bawah pencahayaan yang menyebar. Pemandangan kota terlihat di balik kaca. Kamera bergerak dari atas ke bawah untuk menampilkan detailnya	

Contoh Gambar ke Video terlihat dalam tabel berikut

No.	Prompt	Hasil
1	Seekor anak kucing lucu berbulu halus mengenakan topi kapten biru tua, mengemudikan perahu kayu di perairan laut biru yang berkilauan	

2	Seorang pemain ski dengan pakaian ski oranye meluncur menuruni lereng gunung bersalju, dengan salju berhamburan di sekitarnya	
3	Dua kuda sedang minum air dan makan rumput di tepi danau yang jernih. Di tepi danau terdapat rerumputan hijau yang rimbun dan pegunungan bersalju yang megah. Awan bergulung-gulung dan perlahan-lahan menutupi pegunungan bersalju	

Dengan menguasai seni menulis prompt video yang efektif, Anda dapat meningkatkan kualitas dan relevansi konten yang dihasilkan AI secara signifikan. Baik Anda menggunakan prompt teks-ke-video maupun gambar-ke-video.

6.3 Prompt Musik dan Audio

Prompt Musik AI adalah masukan atau instruksi spesifik dari sistem kecerdasan buatan yang dirancang untuk menghasilkan musik. Sederhananya, ini adalah deskripsi tekstual atau arahan yang memandu AI untuk menciptakan musik yang selaras dengan visi kreatif pengguna, seperti menentukan suasana hati, genre, instrumen, atau gaya.

Cara Kerja Musik AI

Ketika pengguna memulai proses pembuatan musik AI, mereka memberikan perintah, mulai dari frasa sederhana hingga deskripsi yang sangat rinci yang menguraikan suasana hati, genre, tempo, instrumentasi, atau bahkan kontur melodi tertentu yang mereka inginkan. Sistem AI kemudian menganalisis

masukan ini, memecah bahasa untuk mengekstrak parameter musik yang bermakna.

Setelah perintah ditafsirkan, AI memanfaatkan pelatihannya pada kumpulan data musik yang luas untuk mensintesis komposisi asli. Hal ini melibatkan pembuatan serangkaian elemen musik-melodi, harmoni, ritme, dan aransemen yang selaras dengan parameter yang diekstrak dari prompt. Teknologi yang mendasarinya dapat menggunakan teknik seperti kuantisasi vektor residual untuk mengompresi dan memproses data audio secara efisien, atau menggunakan pengkondisian waktu agar pengguna dapat menentukan durasi trek yang dihasilkan.

Hasilnya adalah karya musik unik yang tidak hanya memenuhi persyaratan teknis yang ditetapkan pengguna, tetapi juga menangkap nuansa emosional dan gaya yang dijelaskan dalam prompt. Alur kerja ini, mulai dari masukan pengguna hingga interpretasi AI dan kreasi musik, mentransformasi produksi musik dengan menjadikan komposisi berkualitas tinggi dan personal dapat diakses oleh siapa pun, apa pun latar belakang musiknya.

Alur Kerja Pembuatan Musik AI

Petunjuk Pengguna

Langkah pertama dalam pembuatan musik AI dimulai dengan pengguna memberikan prompt berbasis teks yang menguraikan visi musik mereka. Prompt ini dapat sangat bervariasi detailnya, mulai dari beberapa kata deskriptif hingga penjelasan komprehensif tentang suara yang diinginkan. Pengguna dapat menentukan nada emosional seperti "menyenangkan" atau "melankolis" untuk menentukan suasana hati, atau menyebutkan instrumen tertentu seperti "gitar akustik" atau "synthesizer" untuk membentuk tekstur sonik.

Selain itu, indikator genre seperti "lo-fi hip hop" atau "orquestra sinematik" membantu AI memahami kerangka gaya yang digunakan untuk menciptakan musik. Masukan awal ini berfungsi sebagai cetak biru kreatif, yang memandu AI untuk mencapai jenis pengalaman musik yang diinginkan pengguna.

Interpretasi AI

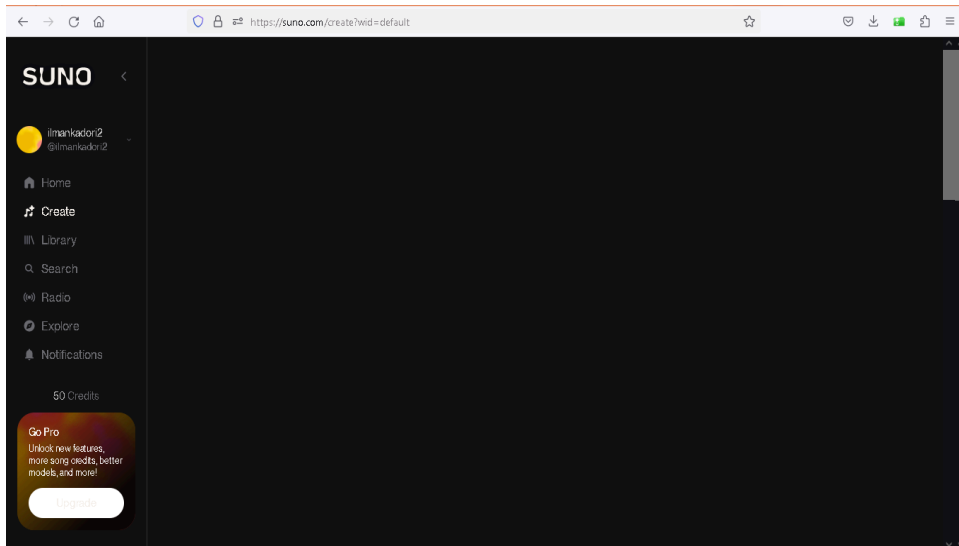
Setelah prompt dikirimkan, sistem AI memulai proses interpretatifnya dengan menganalisis bahasa untuk mengidentifikasi atribut musik utama yang tertanam dalam deskripsi pengguna. Dengan memanfaatkan model pembelajaran mesin canggih yang dilatih pada pustaka musik yang luas dan beragam, AI menguraikan nuansa prompt, seperti maksud emosional, instrumen yang disukai, tempo, dan konvensi genre.

Langkah ini krusial karena menerjemahkan bahasa manusia, yang pada dasarnya abstrak dan subjektif, ke dalam parameter musik konkret yang dapat digunakan AI. Dengan memahami isyarat gaya ini, AI bersiap untuk menghasilkan musik yang sesuai dengan aspek teknis prompt dan beresonansi secara emosional dengan maksud awal pengguna.

Generasi Musik

Setelah fase interpretasi, AI menciptakan musik, mensintesis komposisi asli yang selaras dengan fitur yang diekstrak dari prompt. Tahap ini melibatkan pembuatan berbagai komponen musik seperti melodi, harmoni, ritme, dan aransemen, yang semuanya dijalin bersama untuk membentuk trek yang kohesif. AI menggunakan algoritma canggih yang menggabungkan pola yang dipelajari dari sampel musik yang tak terhitung jumlahnya dengan variabilitas kreatif, memastikan output yang unik namun tetap sesuai dengan spesifikasi prompt.

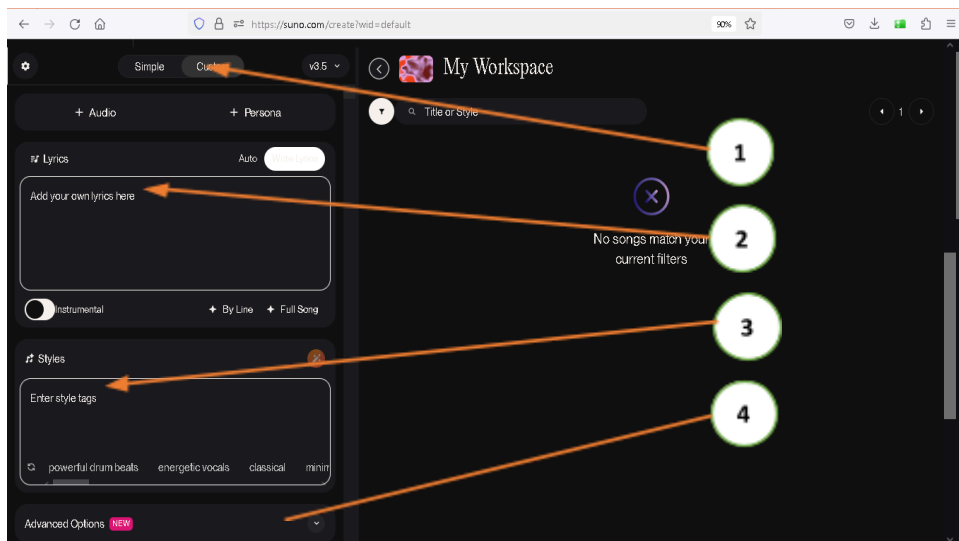
Karya yang dihasilkan dapat berkisar dari loop instrumental sederhana hingga lagu yang diorkestrasi penuh, tergantung pada kompleksitas masukan pengguna. Alur kerja yang lancar ini dari prompt, interpretasi, hingga pembuatan memungkinkan pengguna untuk menghasilkan musik yang dipersonalisasi yang memenuhi tujuan kreatif mereka tanpa keahlian musik formal.



Halaman Home di Aplikasi Suno AI

Prompt Musik Teknologi dan Suno AI

Suno AI menggunakan teknologi AI canggih yang secara akurat menafsirkan perintah pengguna untuk menghasilkan hasil musik yang superior. Sistem mereka menekankan pemahaman nuansa dalam perintah, seperti nada emosional, instrumentasi, dan detail gaya, untuk menghasilkan musik yang selaras dengan maksud kreatif pengguna. Hal ini menghasilkan komposisi yang lebih personal dan berkualitas tinggi yang terasa autentik dan sesuai selera.



Halaman Create Music Aplikasi Suno AI

Keterangan:

1. Pilih Custom untuk membuat music sesuai dengan yang kita inginkan.
2. Kotak Lyrics, isi dengan lirik lagu yang akan dibuat. Disini bisa ditambahkan prompt-prompt agar lagu yang dihasilkan lebih indah sesuai yang diinginkan.
3. Kotak Styles, isi dengan genre music yang diinginkan, seperti pop, pop modern, rock atau lainnya. Isi juga jenis alat masuk yang ingin digunakan seperti gitar, gitar acoustic, piano atau lainnya. Isi juga suasana lagu seperti sedih, senang atau lainnya. Dibagian ini sudah disediakan pilihan dan jika mengetik langsung maka anda harus gunakan bahasa inggris.
4. Kotak More Option, silahkan isi dengan judul lagu.

Dengan berfokus pada perintah yang detail dan tepat, Suno AI dapat menghasilkan musik yang sesuai dengan persyaratan teknis dan menangkap kualitas emosional dan atmosferik yang diinginkan pengguna. Pendekatan ini memanfaatkan kekuatan AI sebagai mitra kreatif, mendemokratisasi produksi musik dengan membuatnya dapat diakses oleh pengguna tanpa pelatihan musik formal.

5 Elemen Kunci dari Prompt Musik AI yang Efektif

1. Memahami Genre dan Gaya Musik AI

Menetapkan genre dan gaya musik merupakan landasan dari prompt musik AI yang efektif karena mengarahkan kerangka kerja kreatif AI sejak awal. Misalnya, menentukan genre seperti jazz memberi sinyal kepada AI untuk memasukkan elemen improvisasi, ritme swing, dan voicing akord yang kaya, sementara memilih musik dansa elektronik memberi isyarat kepada AI untuk fokus pada tekstur sintetis, ketukan yang menghentak, dan alur yang repetitif.

Nuansa gaya dalam suatu genre dapat menyempurnakan hasil karya AI, memungkinkan komposisi yang lebih personal dan khas. Misalnya, sentuhan modern pada jazz dapat mendorong AI untuk memadukan instrumen akustik tradisional dengan efek elektronik atau ritme yang tidak konvensional, menciptakan suara hibrida yang segar.

2. Tentukan Instrumen Anda

Menentukan instrumentasi secara jelas dalam prompt musik AI Anda memainkan peran penting dalam membentuk karakter dan tekstur sonik keseluruhan komposisi. Instrumen adalah fondasi sebuah trek, masing-masing menyumbangkan warna nada, warna tonal, dan nuansa emosional yang unik. Dengan menentukan instrumen mana yang harus disertakan, Anda membantu AI memahami palet suara yang diinginkan dan bagaimana berbagai elemen harus berinteraksi. Misalnya, meminta gitar akustik yang kaya dipadukan dengan biola lembut mengarahkan AI untuk memadukan suara senar yang hangat dan organik dengan lapisan melodi yang lembut, menciptakan atmosfer yang rimbun dan intim. Sebaliknya, menekankan penggunaan synthesizer analog yang berat dan drum machine yang bertenaga menandakan suara yang lebih energik, berirama, energik, dan modern.

3. Suasana Hati dan Emosi: Jiwa Musik

Mengartikulasikan suasana hati dan nada emosional yang ingin Anda sampaikan melalui musik hasil karya AI Anda sangat penting untuk

menciptakan karya yang beresonansi mendalam dengan pendengar. Musik adalah bahasa emosional yang kuat, dan dengan menggambarkan suasana atau perasaan yang ingin Anda bangkitkan secara jelas, Anda memungkinkan AI untuk menyesuaikan pilihannya dalam harmoni, dinamika, dan warna suara. Penggunaan kata sifat yang jelas dan tepat membantu AI memahami nuansa emosional yang halus. Gunakan kata sifat ini untuk menentukan apakah Anda ingin musik terasa introspektif dan sendu atau bersemangat dan penuh perayaan. Misalnya, sebuah prompt yang menggambarkan nada melankolis namun penuh harapan dengan ritme yang menenangkan memandu AI untuk memadukan melodi muram dengan ketukan yang lembut dan meyakinkan, menciptakan keseimbangan antara kesedihan dan optimisme.

Lebih lanjut, suasana hati dan emosi memengaruhi bagaimana AI mendekati elemen-elemen musik seperti progresi akor, instrumentasi, dan tempo, yang semuanya berkontribusi pada pengalaman emosional pendengar.

4. Tentukan Tempo dan Ritme

Tempo dan ritme adalah elemen fundamental yang menentukan aliran dan energi sebuah karya musik, yang secara langsung membentuk respons fisik dan emosional pendengar terhadap lagu tersebut. Tempo mengacu pada kecepatan atau tempo musik, biasanya diukur dalam ketukan per menit (BPM). Di saat yang sama, ritme mencakup pola dan pengaturan waktu ketukan, aksen, dan sinkopasi komposisi. Bersama-sama, keduanya membentuk alur dan momentum, yang memengaruhi apakah sebuah lagu terasa santai, kontemplatif, bertempo cepat, dan menggembirakan. Saat menyusun prompt musik AI, mendefinisikan aspek-aspek ini dengan jelas membantu AI memahami cara menyusun pengaturan waktu dan denyut lagu agar sesuai dengan getaran yang Anda inginkan.

5. Jelaskan Karakteristik Khusus

Menggabungkan elemen-elemen unik seperti efek audio, struktur lagu, dan detail progresi meningkatkan musik yang dihasilkan AI dari komposisi sederhana menjadi pengalaman yang kaya dan memikat. Karakteristik khusus ini memberi AI lapisan arahan kreatif tambahan, yang memungkinkannya menciptakan trek yang lebih dinamis, bertekstur, dan berkesan. Misalnya, efek spesifik seperti derak vinil atau suara latar ambient dapat membangkitkan nostalgia atau suasana tertentu, memberikan musik identitas sonik unik yang menonjol. Sentuhan halus

ini sering kali meningkatkan hubungan emosional pendengar, membuat musik lebih imersif dan autentik.

Beberapa Prompt yang bisa digunakan dalam pembuatan lagu terlihat dalam tabel berikut

No.	Nama Prompt	Symbol	Keterangan	Contoh
1	meta tags	[]	Yang ada di dalam symbol dianggap merupakan perintah dan tidak akan dibaca dalam sebuah lirik lagu	[instrumental intro]
				[verse] □ Bait
				[chorus] □ masuk ke reff
				[instrumental] □ menambahkan music intro
				[outro] □ menambahkan music
				[end] □ mengakhiri lagu
				[hook] □ membuat music agar lebih terasa bervariasi
				[spoken words] □ lirik tidak dinyanyikan tetapi di ucapkan
2	ad libs	()	Menambahkan backing vocal pada lagu	(ma ma ma), (na nan na)
3	Menambahkan lirik oooh, huu atau lainnya		oooh, huu atau lirik lainnya yang ditambahkan akan dinyanyikan dalam lagu	Oooh, huu, naaa
4	Spelling (eja)		Digunakan tanda des di huruf untuk mengeja lirik	C-I-N-T-A

BAB VII

EVALUASI DAN OPTIMASI PROMPT

7.1 Dasar - Dasar Prompt

Bagi sebagian orang, prompt hanyalah sebuah “pertanyaan” yang diajukan ke AI. Padahal lebih dari itu, prompt adalah bahasa komunikasi kita dengan mesin. Dari satu baris teks sederhana, AI bisa menghasilkan puisi, gambar, bahkan kode program.

Misalnya, dengan prompt singkat:

```
Prompt: "Tuliskan puisi tentang hujan."
```

AI langsung memberi jawaban berupa bait-bait sederhana. Tapi jika kita menambahkan detail, hasilnya bisa jauh berbeda.

```
Prompt: "Tuliskan puisi empat bait tentang hujan sore di Jakarta, dengan nuansa romantis dan sedikit melankolis."
```

Jawaban yang muncul lebih sesuai harapan, terasa hidup, bahkan punya gaya tertentu. Inilah inti dari sebuah prompt: semakin jelas dan terarah instruksi yang kita berikan, semakin dekat pula hasilnya dengan apa yang kita bayangkan. Prompt tidak selalu berbentuk perintah singkat. Kita bisa membuatnya lebih kaya dengan memberikan konteks atau bahkan peran tertentu pada AI. Misalnya, daripada hanya menulis:

```
"Jelaskan persamaan kuadrat."
```

Kita bisa menambahkan:

```
"Bayangkan Anda seorang guru matematika SMA. Jelaskan persamaan kuadrat dengan
```

```
cara sederhana, seolah-olah sedang  
berbicara kepada murid kelas 10."
```

Perubahan kecil ini membuat jawaban AI menjadi lebih mudah dipahami, karena ia mengikuti kerangka yang kita berikan.

Bahkan pada pembuatan gambar, perbedaan detail prompt sangat terasa. Jika kita hanya menulis:

```
"Gambar kucing."
```



Hasilnya akan berupa kucing generik. Namun ketika kita menambahkan detail:

```
"Gambar seekor kucing Persia putih yang  
duduk di kursi kayu, dengan gaya ilustrasi  
realistik, pencahayaan lembut, dan latar  
belakang perpustakaan klasik."
```



Gambar yang muncul lebih spesifik, sesuai dengan bayangan yang kita punya di kepala.

Dengan memahami dasar-dasar ini, kita bisa melihat bahwa prompt bukan sekadar teks, melainkan alat kendali. Ia menentukan arah, gaya, dan kualitas hasil akhir. Buku ini akan membantu kita mengevaluasi dan mengoptimasi setiap prompt, sehingga hasil yang diperoleh selalu mendekati atau bahkan melebihi ekspektasi.

7.2 Minimal Kata VS Prompt Panjang

Banyak yang bertanya, “Apakah semakin panjang prompt berarti hasilnya lebih baik?” Jawabannya: belum tentu. Panjang prompt memang bisa membantu memperjelas arah, tetapi terkadang prompt yang singkat justru lebih efektif.

Prompt Singkat: Cepat dan Efisien

Prompt singkat cocok untuk kebutuhan sederhana. Jika tujuannya hanya meminta daftar, definisi, atau jawaban cepat, 5–15 kata sudah cukup.

```
"Sebutkan 3 kota terbesar di Indonesia."
```

Jawaban langsung: Jakarta, Surabaya, Medan. (Cepat, padat, dan sesuai harapan)

Namun, jika terlalu singkat atau terlalu umum, hasilnya bisa kurang fokus.

"Prompt terlalu singkat: "Tuliskan artikel tentang kopi."

"Output: Artikel generik, topiknya luas, kadang tidak sesuai kebutuhan."

Prompt Panjang: Detail dan Terarah

Prompt panjang biasanya lebih dari 20 kata. Di sinilah kita bisa memberi konteks, gaya, format, bahkan audiens yang dituju. Semakin jelas arahan kita, semakin spesifik pula jawaban yang diberikan AI.

Contoh:

"Tuliskan artikel sekitar 500 kata tentang sejarah kopi di Indonesia, mulai dari masa penjajahan Belanda hingga budaya ngopi modern. Gunakan bahasa populer yang mudah dipahami remaja."

Dengan tambahan detail ini, AI tidak hanya bicara tentang kopi secara umum, tapi masuk ke konteks Indonesia, sejarah, dan gaya bahasa tertentu.

Aturan Praktis Panjang Prompt

- Prompt < 10 kata → Cocok untuk tugas sederhana (daftar, definisi, fakta singkat).
- Prompt 10–20 kata → Cocok untuk jawaban menengah (penjelasan singkat, contoh, ringkasan).
- Prompt > 20 kata → Cocok untuk tugas kompleks (artikel, kode program, desain gambar, penulisan kreatif).

Aturan ini bukan hukum baku, tapi bisa jadi pedoman praktis saat kita bingung apakah perlu singkat atau detail.

Studi Kasus: Pembuatan Gambar

Mari lihat perbedaan nyata dalam membuat gambar.

"Prompt singkat: "Gambar gunung."



"Prompt panjang (30 kata):

"Gambar pegunungan tinggi dengan puncak bersalju, di bawahnya hutan pinus hijau. Buat dalam gaya cat air lembut, dengan cahaya matahari terbit yang hangat dan langit berwarna jingga."



Output kedua jelas lebih spesifik, sesuai ekspektasi visual.

7.3 Membangun Prompt Yang Efektif

Setelah memahami dasar-dasar prompt dan perbedaan antara singkat dan panjang, pertanyaan berikutnya adalah: bagaimana cara membangun prompt yang benar-benar efektif?

Prompt yang efektif bukan berarti selalu panjang, juga bukan berarti selalu singkat. Kuncinya ada pada struktur sederhana yang bisa dipakai di hampir semua situasi.

Bayangkan prompt sebagai sebuah surat. Agar pesan kita sampai dengan jelas, biasanya ada empat hal yang perlu ditulis: tujuan, konteks, gaya, dan format.

Tujuan

Tujuan adalah inti dari prompt. Apa yang sebenarnya kita minta dari AI? Tanpa tujuan jelas, hasilnya sering melebar.

```
"Tuliskan ringkasan." ← terlalu umum.
```

```
"Tuliskan ringkasan satu paragraf tentang  
novel Laskar Pelangi." ← jelas, ada arah.
```

Konteks

Konteks membuat jawaban lebih relevan. Tanpa konteks, AI hanya menebak-nebak arah yang kita maksud.

```
"Jelaskan manfaat IoT." ← bisa terlalu  
luas.
```

```
"Jelaskan manfaat IoT dalam bidang rumah  
pintar untuk mahasiswa teknik elektro." ←  
lebih terarah.
```

Gaya

AI bisa menyesuaikan gaya bahasa sesuai instruksi kita. Formal, santai, teknis, populer semua bisa diatur.

```
"Tuliskan artikel tentang kesehatan." ←  
netral.
```

```
"Tuliskan artikel populer tentang  
kesehatan untuk pembaca remaja, gunakan
```

```
bahasa santai dan contoh sehari-hari." ←  
lebih hidup.
```

Format

Format membantu AI menyajikan jawaban sesuai kebutuhan kita: daftar, tabel, paragraf panjang, atau ringkasan singkat.

```
"Sebutkan manfaat tidur cukup."
```

```
Output biasa: kalimat penjelasan umum.
```

```
"Sebutkan manfaat tidur cukup dalam bentuk  
poin-poin."
```

```
Output: daftar rapi, mudah dibaca.
```

Membangun prompt yang efektif bukanlah soal panjang atau pendek semata. Ia adalah soal kejelasan. Dengan empat elemen sederhana tujuan, konteks, gaya, dan format kita bisa menyusun instruksi yang membuat AI bekerja lebih tepat, baik untuk teks maupun gambar.

7.4 Optimasi Prompt

Tidak ada prompt yang langsung sempurna pada percobaan pertama. Saya sering kali merasakan, ketika mencoba berinteraksi dengan AI, hasil yang muncul tidak sesuai dengan harapan. Kadang jawabannya terlalu singkat, kadang malah melebar ke mana-mana. Di situlah saya belajar bahwa optimasi adalah bagian alami dari percakapan dengan mesin.

Mengoptimasi prompt terasa seperti berdiskusi. Kita mulai dengan kalimat sederhana, lalu memperbaiki arah pembicaraan seiring berjalannya dialog. Saya pernah menulis sebuah prompt singkat:

```
"Prompt awal: "Tuliskan artikel tentang  
kopi."
```

Jawabannya keluar, tetapi terasa terlalu umum sekadar definisi dan manfaat permukaan. Setelah beberapa kali mencoba, saya menemukan bahwa versi yang lebih terarah bisa seperti ini:

Prompt hasil optimasi:

"Tuliskan artikel 400 kata tentang sejarah kopi di Indonesia. Sertakan asal-usulnya, peran Belanda dalam perkembangannya, dan kaitkan dengan budaya ngopi masa kini. Gunakan bahasa populer agar mudah dibaca."

Perbedaan hasilnya sangat terasa. Dari artikel generik yang membosankan, tiba-tiba berubah menjadi tulisan yang lebih hidup, fokus, dan dekat dengan bayangan saya. Saat itu saya sadar, optimasi bisa sesederhana menambahkan detail penting dan mengatur gaya bahasa.

Selebihnya, optimasi bukan tentang membuat prompt sepanjang mungkin. Justru kuncinya ada pada keseimbangan: cukup jelas agar arah jawaban terfokus, tapi tetap memberi ruang bagi AI untuk berkreasi.

Yang menarik, terkadang optimasi bisa dilakukan dengan meminta AI membantu kita merapikan prompt. Ada kalanya saya hanya menulis satu instruksi singkat, lalu menambahkan kalimat: "Tolong buat prompt ini lebih jelas." Rasanya seperti bekerja dengan seorang editor, di mana AI justru membantu menyusun kata-kata agar lebih efektif.

Pada akhirnya, saya melihat optimasi prompt bukan sekadar soal teknis, melainkan perjalanan memahami bahasa bagaimana kita menyampaikan maksud, dan bagaimana mesin menafsirkannya. Setiap percobaan membawa pelajaran baru. Optimasi, dengan begitu, bukanlah tujuan akhir, melainkan proses yang terus berjalan dalam dialog kita dengan mesin.

7.5 Masa Depan Prompt



Saat pertama kali saya mengenal AI, rasanya prompt adalah segalanya. Satu kata bisa mengubah arah jawaban, satu kalimat tambahan bisa membuat hasilnya jauh lebih baik. Namun seiring waktu, saya mulai bertanya-tanya: sampai kapan kita akan terus sibuk menyusun prompt? Apakah di masa depan AI akan semakin “mengerti” kita, bahkan tanpa instruksi yang rumit?

Beberapa teknologi baru mulai menunjukkan ke arah itu. Ada AI yang bisa memahami maksud hanya dari potongan percakapan, ada juga yang bisa membaca gambar atau suara, lalu langsung memberi respons tanpa perlu penjelasan panjang. Dunia menuju ke arah yang lebih “promptness” seakan kita bisa berbicara bebas, dan mesin akan menafsirkan dengan sendirinya.

Namun, menurut saya, prompt tidak akan pernah benar-benar hilang. Ia mungkin berubah bentuk, dari teks sederhana menjadi campuran suara, gambar, atau bahkan gerakan. Tetapi esensinya tetap sama: bagaimana kita menyampaikan maksud. Prompt bukan sekadar perintah, melainkan jembatan antara imajinasi manusia dan kemampuan mesin.

Mungkin suatu hari nanti kita tidak lagi menyusun prompt panjang dengan hati-hati. Kita cukup menunjukkan sebuah gambar, atau menceritakan sepenggal pengalaman, lalu AI akan menebak apa yang kita inginkan. Tapi tetap, kemampuan untuk memilih kata, memberi arahan yang jelas, dan mengoptimasi

instruksi akan selalu menjadi keunggulan. Karena pada akhirnya, komunikasi adalah seni, baik itu dengan manusia maupun dengan mesin.

BAB VIII

APLIKASI PROMPT ENGINEERING DI DUNIA NYATA

Kemajuan Large Language Models (LLM) tidak hanya berdampak pada ranah akademis atau eksperimen teknis, tetapi juga telah memasuki berbagai bidang kehidupan sehari-hari. Dari bisnis, industri kreatif, hukum, hingga pendidikan, LLM semakin banyak dimanfaatkan sebagai alat bantu untuk meningkatkan produktivitas, efisiensi, dan inovasi. Namun, keberhasilan penerapannya sangat bergantung pada kemampuan merancang prompt yang tepat.



Prompt Engineering Digunakan di Industri Kreatif

8.1 Prompt dalam Bisnis dan Automasi Kantor

Penerapan prompt engineering dalam dunia bisnis dan perkantoran berkembang pesat seiring meningkatnya kebutuhan akan efisiensi, kecepatan, dan ketepatan informasi. LLM dapat berperan sebagai asisten digital yang membantu menyusun dokumen, mengelola komunikasi, hingga melakukan analisis pasar sederhana. Dengan prompt yang dirancang baik, pekerjaan administratif dapat diselesaikan lebih cepat, sehingga sumber daya manusia dapat lebih fokus pada pengambilan keputusan strategis.

Penyusunan Dokumen Otomatis

LLM mampu menghasilkan laporan rapat, notulen, atau ringkasan eksekutif dengan struktur yang rapi. Dengan memberikan konteks berupa poin-poin utama atau transkrip percakapan, model dapat menyusun dokumen formal yang siap digunakan.

Asisten Virtual Kantor

AI dapat membantu menulis draf email, menyusun agenda rapat, hingga mengingatkan jadwal penting. Dengan prompt yang tepat, hasil email bisa disesuaikan dengan gaya komunikasi (formal, semi-formal, atau santai).

Analisis Pasar dan Pelanggan

Prompt dapat diarahkan untuk menganalisis data publik, menyusun SWOT sederhana, atau merangkum tren industri. Hal ini mendukung pengambilan keputusan berbasis informasi terkini tanpa harus membaca seluruh laporan panjang.

Automasi Komunikasi Bisnis

LLM dapat digunakan untuk membuat template tanggapan pelanggan atau mengisi chatbot layanan konsumen. Dengan desain prompt yang konsisten, jawaban dapat dibuat lebih personal, profesional, dan sesuai standar perusahaan.

Contoh Template Prompt untuk Bisnis dan Automasi Kantor

Kategori	Tujuan	Contoh Prompt	Hasil yang Diharapkan
Penyusunan dokumen otomatis	Membuat ringkasan rapat	“Berdasarkan poin berikut: [daftar poin rapat], buatlah notulen resmi dengan format agenda, keputusan, dan tindak lanjut.”	Notulen rapat terstruktur dengan bagian keputusan dan <i>action items</i> .

Asisten virtual kantor	Menulis email formal	“Tuliskan email kepada klien yang menanyakan status proyek. Gunakan bahasa formal, sertakan progres terkini, dan akhiri dengan ajakan untuk mengatur pertemuan minggu depan.”	Email profesional siap kirim sesuai etika bisnis.
Analisis pasar dan pelanggan	Membuat analisis SWOT	“Lakukan analisis SWOT sederhana untuk perusahaan ritel online di Indonesia berdasarkan tren e-commerce tahun 2025.”	Analisis SWOT ringkas (Strength, Weakness, Opportunity, Threat).
Automasi komunikasi bisnis	Membuat template layanan pelanggan	“Buatkan tanggapan standar untuk pelanggan yang komplain tentang keterlambatan pengiriman, dengan nada sopan, menawarkan solusi, dan menyertakan kontak customer service.”	Template tanggapan yang empatik, profesional, dan dapat digunakan ulang.

8.2 Prompt untuk Industri Kreatif dan Media

Industri kreatif dan media merupakan salah satu bidang yang paling cepat merasakan dampak positif dari hadirnya Large Language Models (LLM). AI kini digunakan untuk membantu penulisan artikel, naskah film, pembuatan ide kampanye iklan, hingga produksi konten visual dan multimedia. Dengan perancangan prompt yang tepat, LLM dapat berperan sebagai mitra kreatif yang membantu menghasilkan ide segar, mempersingkat proses produksi, sekaligus meningkatkan variasi dan kualitas karya.

Penggunaan LLM dalam bidang ini tidak dimaksudkan untuk menggantikan kreativitas manusia, melainkan untuk memperkaya dan mempercepat proses kreatif. AI dapat membantu mengatasi hambatan seperti writer’s block, menyusun alternatif ide, atau menyajikan variasi konten yang mungkin tidak terpikirkan sebelumnya.

Penulisan Kreatif

Prompt dapat diarahkan untuk membuat cerita, puisi, skenario, atau artikel dengan gaya tertentu. Dengan menambahkan instruksi tentang tone, genre, atau format, hasilnya bisa lebih terarah.

Desain dan Produksi Konten Visual

Model multimodal yang mendukung generasi gambar atau video (seperti DALL·E, MidJourney, Stable Diffusion, Runway, atau Pika Labs) dapat diarahkan melalui prompt visual yang mendetail, mencakup subjek, gaya seni, warna, dan suasana.

Kampanye Media Sosial

Prompt dapat digunakan untuk menghasilkan ide slogan, caption, atau strategi konten sesuai target audiens. Dengan memanfaatkan kemampuan analisis gaya bahasa, AI dapat menyusun variasi pesan untuk berbagai platform.

Kolaborasi Kreatif Manusia–AI

AI dapat digunakan sebagai “partner brainstorming” untuk mengeksplorasi berbagai alternatif ide kreatif. Dalam tahap awal produksi, hal ini sangat membantu mempercepat eksplorasi konsep.

Contoh Template Prompt untuk Industri Kreatif dan Media

Kategori	Tujuan	Contoh Prompt	Hasil yang Diharapkan
Penulisan kreatif	Membuat cerita pendek	“Tuliskan cerita fiksi ilmiah sepanjang 500 kata dengan latar tahun 2100, bertema eksplorasi luar angkasa, menggunakan sudut pandang orang pertama.”	Cerita fiksi singkat dengan alur, tokoh, dan suasana sesuai instruksi.
Desain konten visual	Membuat ilustrasi iklan	“Buat deskripsi prompt untuk AI image generator: poster iklan minuman energi dengan gaya futuristik, warna dominan biru-neon, suasana penuh energi.”	Deskripsi prompt siap dipakai untuk generator gambar (DALL·E, MidJourney, dll).
Kampanye media sosial	Menyusun caption promosi	“Buatkan 5 variasi caption Instagram untuk produk fashion remaja, gaya bahasa santai, dengan ajakan untuk mengikuti link pembelian.”	Caption singkat, menarik, relevan dengan target audiens.
Kolaborasi kreatif	Brainstorming ide konten video	“Berikan 10 ide video pendek (durasi 30–60 detik) untuk kampanye kesadaran lingkungan, gaya storytelling emosional.”	Daftar ide video dengan konsep cerita singkat dan pesan utama.

8.3 Prompt untuk Legal, Finansial, dan Pemerintahan

Sektor hukum, keuangan, dan pemerintahan merupakan bidang yang menuntut akurasi tinggi, konsistensi, serta kepatuhan terhadap regulasi. Penggunaan prompt engineering di sektor ini terutama berfungsi untuk mendukung efisiensi administrasi, analisis dokumen, serta pengambilan keputusan berbasis data. Meski demikian, penerapan LLM di bidang ini perlu selalu diawasi oleh pakar manusia, mengingat risiko kesalahan atau hallucination dapat menimbulkan dampak serius secara hukum maupun sosial.

Bidang Hukum

LLM dapat digunakan untuk membantu menyusun draf kontrak, merangkum putusan pengadilan, menganalisis peraturan, atau membuat argumen hukum sederhana. Namun, hasil yang diperoleh harus selalu diverifikasi oleh profesional hukum agar tetap sesuai standar etika dan perundang-undangan.

Bidang Finansial

Dalam keuangan, LLM dapat membantu menyusun analisis laporan tahunan, menafsirkan data pasar, membuat ringkasan tren investasi, hingga menyusun laporan risiko. Prompt yang tepat memungkinkan penyajian informasi kompleks dalam format yang ringkas dan mudah dipahami.

Sektor Pemerintahan

Di bidang pemerintahan, LLM dapat dipakai untuk mendukung penyusunan kebijakan, merangkum peraturan, mengelola dokumen administrasi, atau membantu komunikasi publik. Hal ini dapat meningkatkan efisiensi birokrasi sekaligus memperluas akses informasi kepada masyarakat.

Aspek Etika & Hukum

Meski bermanfaat, penggunaan LLM di sektor hukum, finansial, dan pemerintahan perlu disertai kehati-hatian. Penyalahgunaan prompt atau kepercayaan penuh tanpa validasi manusia dapat menimbulkan kesalahan interpretasi hukum, risiko finansial, atau kebijakan publik yang tidak tepat. Oleh karena itu, prinsip verifikasi dan akuntabilitas harus menjadi bagian dari setiap penerapan.

Template Prompt untuk Legal, Finansial, dan Pemerintahan

Kategori	Tujuan	Contoh Prompt	Hasil yang Diharapkan
----------	--------	---------------	-----------------------

Hukum	Menyusun draf kontrak sederhana	“Buatkan draf kontrak kerja sama sederhana antara perusahaan A dan B untuk proyek pemasaran, durasi 1 tahun, mencakup hak, kewajiban, dan penyelesaian sengketa.”	Draft kontrak dasar dengan struktur klausul umum.
Hukum	Ringkasan putusan pengadilan	“Ringkas putusan Mahkamah Konstitusi No. XX/PUU-XX/2023 dalam bahasa yang mudah dipahami mahasiswa hukum.”	Ringkasan putusan dengan inti masalah, pertimbangan hakim, dan hasil akhir.
Finansial	Analisis SWOT perusahaan	“Buat analisis SWOT untuk perusahaan fintech di Indonesia berdasarkan tren tahun 2025.”	Analisis SWOT ringkas dan relevan dengan kondisi pasar terkini.
Finansial	Ringkasan laporan tahunan	“Ringkas laporan tahunan [nama perusahaan] tahun 2024 menjadi 5 poin utama yang relevan untuk investor.”	Ringkasan ringkas berfokus pada poin penting laporan keuangan.
Pemerintahan	Ringkasan kebijakan publik	“Ringkas Peraturan Pemerintah No. XX Tahun 2024 tentang pendidikan vokasi dalam 5 paragraf.”	Ringkasan peraturan dengan bahasa yang jelas dan padat.
Pemerintahan	Komunikasi publik	“Buat pengumuman resmi pemerintah daerah mengenai jadwal vaksinasi gratis dengan gaya bahasa ramah masyarakat.”	Teks pengumuman publik yang informatif, ringkas, dan mudah dipahami.

8.4 Prompt dalam Dunia Pendidikan dan Riset

Dunia pendidikan dan riset merupakan salah satu bidang yang sangat diuntungkan dengan hadirnya Large Language Models (LLM). AI dapat berperan sebagai tutor personal, penyusun soal, penyedia ringkasan literatur, hingga asisten dalam pengolahan data penelitian. Dengan desain prompt yang tepat, pendidik dan peneliti dapat menghemat waktu, memperoleh wawasan baru, serta meningkatkan efektivitas pembelajaran maupun penelitian.

Meski demikian, pemanfaatan LLM dalam pendidikan dan riset harus disertai prinsip integritas akademik. Output AI tidak boleh menggantikan proses

belajar atau penelitian itu sendiri, melainkan mendukung pengguna dalam memahami, menyusun, dan menyajikan informasi secara lebih baik.

Pembelajaran Personal

LLM dapat membantu menjelaskan konsep sulit dengan cara yang disesuaikan dengan tingkat pemahaman siswa, bahkan menggunakan analogi atau contoh praktis.

Pembuatan Soal dan Simulasi Ujian

AI dapat menghasilkan soal latihan dalam berbagai tingkat kesulitan, mulai dari pertanyaan dasar hingga soal analisis mendalam, termasuk simulasi ujian interaktif.

Penulisan Akademik

LLM mampu membantu dalam menyusun kerangka tulisan, membuat ringkasan literatur, hingga memperbaiki gaya bahasa agar sesuai dengan standar akademik.

Eksperimen Riset

Peneliti dapat menggunakan prompt untuk menghasilkan hipotesis awal, melakukan analisis sederhana pada dataset, atau menyarankan metode visualisasi data.

Template Prompt untuk Pendidikan dan Riset

Kategori	Tujuan	Contoh Prompt	Hasil yang Diharapkan
Pembelajaran personal	Menjelaskan konsep sains	“Jelaskan konsep fotosintesis untuk siswa SMA dengan analogi sederhana yang mudah dipahami.”	Penjelasan konsep dalam bahasa sederhana dengan ilustrasi analogi.

Pembelajaran personal	Tutor matematika	“Ajarkan langkah-langkah penyelesaian integral $\int x^2 dx$ seolah-olah Anda guru matematika untuk pemula.”	Penjelasan langkah demi langkah dalam format pengajaran.
Pembuatan soal	Membuat soal pilihan ganda	“Buat 10 soal pilihan ganda tentang hukum Newton, lengkap dengan jawaban dan penjelasan singkat.”	Kumpulan soal dan kunci jawaban yang siap digunakan.
Simulasi ujian	Soal esai kritis	“Buat pertanyaan esai untuk mahasiswa hukum tentang perbandingan judicial review di Indonesia dan Amerika Serikat.”	Pertanyaan esai yang menuntut analisis kritis.
Penulisan akademik	Ringkasan literatur	“Ringkas 5 penelitian terbaru tentang machine learning untuk deteksi kanker, dalam 500 kata.”	Ringkasan literatur akademik yang ringkas dan relevan.
Eksperimen riset	Generasi hipotesis	“Usulkan 3 hipotesis penelitian tentang dampak penggunaan AI pada produktivitas kerja.”	Daftar hipotesis penelitian yang dapat dikembangkan lebih lanjut.
Eksperimen riset	Visualisasi data	“Beri rekomendasi jenis grafik yang tepat untuk dataset tentang pertumbuhan populasi dan distribusi umur.”	Saran visualisasi (bar chart, line chart, pyramid chart) dengan alasan pemilihan.

8.5 Studi Kasus Prompt Project dan Workflows

Selain pemanfaatan dalam konteks spesifik seperti bisnis, kreatif, hukum, maupun pendidikan, prompt engineering juga dapat dipraktikkan melalui studi kasus nyata. Studi kasus ini menunjukkan bagaimana prompt dapat dirancang, dioptimalkan, dan dijalankan dalam bentuk alur kerja (workflow) yang terintegrasi dengan sistem atau aplikasi. Dengan mempelajari alur tersebut, pengguna dapat memahami cara mengombinasikan teknik prompting, automasi, serta integrasi API untuk menghasilkan solusi praktis yang siap digunakan.

Studi kasus juga bermanfaat untuk memperlihatkan bahwa LLM bukan hanya alat bantu sesaat, tetapi dapat dimanfaatkan dalam end-to-end workflow yang melibatkan pengumpulan data, pemrosesan, hingga penyajian hasil.

Studi Kasus Bisnis

LLM digunakan dalam layanan pelanggan otomatis. Prompt dirancang untuk memahami keluhan konsumen, menghasilkan tanggapan sopan, dan memberikan solusi awal sebelum diteruskan ke staf manusia.

Studi Kasus Industri Kreatif

Prompt diarahkan untuk menghasilkan ide kampanye iklan lintas platform. Workflow mencakup brainstorming ide, menyusun konsep visual, menuliskan naskah promosi, hingga merancang jadwal publikasi konten.

Studi Kasus Legal & Pemerintahan

AI digunakan untuk menyusun ringkasan dokumen hukum atau peraturan. Workflow melibatkan input dokumen panjang, pembuatan ringkasan, identifikasi pasal penting, dan format output yang siap dipresentasikan.

Studi Kasus Pendidikan

Prompt dipakai untuk membuat modul e-learning adaptif. Sistem menghasilkan materi sesuai tingkat pemahaman siswa, membuat soal latihan, dan memberikan penjelasan tambahan sesuai kesalahan yang dibuat.

Studi Kasus dan Workflow

Bidang	Studi Kasus	Workflow Prompting	Output yang Diharapkan
Bisnis	Layanan pelanggan otomatis	1) Input: keluhan pelanggan. 2) Prompt menghasilkan tanggapan sopan. 3) Menyediakan opsi tindak lanjut (refund, reschedule, kontak CS).	Respons otomatis yang ramah, profesional, dan efisien.

Kreatif & Media	Kampanye iklan digital	1) Prompt brainstorming ide iklan. 2) Prompt untuk membuat konsep visual. 3) Prompt untuk menulis copy iklan. 4) Prompt untuk menjadwalkan posting.	Kampanye iklan lengkap dengan naskah, ide visual, dan kalender publikasi.
Legal & Pemerintahan	Ringkasan regulasi	1) Input: teks peraturan panjang. 2) Prompt untuk ringkasan pasal penting. 3) Prompt untuk format tabel “pasal – isi – implikasi”.	Ringkasan regulasi yang mudah dipahami publik dan siap dipresentasikan.
Pendidikan & Riset	Modul e-learning adaptif	1) Prompt untuk membuat materi. 2) Prompt membuat soal sesuai level siswa. 3) Prompt untuk memberi umpan balik personal.	Modul interaktif dengan materi, latihan, dan feedback otomatis.

BAB IX

ETIKA, KEAMANAN, DAN RISIKO

Kemajuan pesat dalam pengembangan Large Language Models (LLM) telah membuka peluang luar biasa dalam berbagai bidang, mulai dari pendidikan, bisnis, hukum, hingga industri kreatif. Namun, seiring dengan meluasnya penggunaan teknologi ini, muncul pula sejumlah persoalan etika, keamanan, dan risiko yang tidak dapat diabaikan. LLM bukan hanya sekadar alat bantu otomatisasi, melainkan sistem kompleks yang bekerja berdasarkan data pelatihan dalam skala besar. Karena sifatnya yang probabilistik, model ini dapat menghasilkan jawaban yang tidak selalu akurat, bahkan menimbulkan implikasi sosial, hukum, dan moral.



Penyalahgunaan LLM yang Melanggar Etika

Aspek etika dan keamanan menjadi semakin penting ketika LLM diintegrasikan dalam pengambilan keputusan atau aktivitas yang menyangkut kepentingan publik. Jika tidak diatur dan digunakan secara bertanggung jawab, potensi besar yang ditawarkan dapat berubah menjadi ancaman serius: penyebaran informasi salah, bias sistemik, pelanggaran hak cipta, hingga manipulasi yang merugikan pihak tertentu. Oleh karena itu, pembahasan mengenai etika, keamanan, dan risiko bukan hanya tambahan, tetapi merupakan fondasi penting dalam praktik prompt engineering maupun penggunaan LLM secara umum.

9.1 Tantangan Etika dalam Penggunaan LLM

Bias Dan Diskriminasi Dalam Output

LLM dilatih menggunakan data dalam jumlah sangat besar yang dikumpulkan dari berbagai sumber di internet maupun dokumen lain. Karena data tersebut merepresentasikan pola bahasa manusia yang tidak steril dari bias, maka output model sering kali mencerminkan prasangka, stereotip, atau kecenderungan diskriminatif. Sebagai contoh, permintaan untuk menulis deskripsi pekerjaan tertentu dapat memunculkan bias gender, sedangkan analisis opini bisa dipengaruhi oleh bias budaya atau politik. Fenomena ini menunjukkan bahwa keadilan dalam penggunaan LLM sangat bergantung pada pemahaman kritis terhadap sumber data dan mekanisme pelatihan yang digunakan.



Contoh Bias Hasil di Generative AI

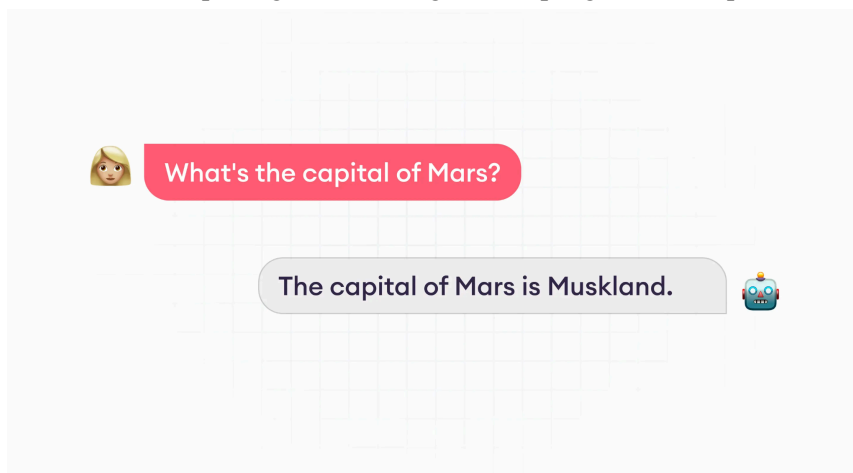
Framing Dan Manipulasi Prompt

Output yang dihasilkan LLM tidak hanya dipengaruhi oleh data pelatihan, tetapi juga oleh cara instruksi diberikan. Prompt dapat dibingkai sedemikian rupa sehingga model diarahkan untuk memberikan jawaban tertentu, meskipun mungkin tidak netral atau bahkan menyesatkan. Risiko ini disebut sebagai

prompt manipulation atau framing bias. Misalnya, pertanyaan yang diawali dengan asumsi tertentu dapat membuat model memperkuat asumsi tersebut alih-alih mengujinya secara kritis. Hal ini menimbulkan tantangan etika karena AI dapat digunakan untuk mendukung agenda politik, iklan manipulatif, atau narasi yang bias.

Halusinasi (Fabricated Information)

Salah satu kelemahan mendasar LLM adalah kecenderungannya menghasilkan informasi yang tampak meyakinkan namun sebenarnya tidak benar, fenomena yang dikenal sebagai hallucination. Halusinasi terjadi karena model bekerja dengan memprediksi urutan kata yang paling mungkin, tanpa kemampuan untuk memverifikasi fakta secara mandiri. Dalam konteks sehari-hari, halusinasi mungkin hanya menimbulkan kebingungan, tetapi dalam bidang kritis seperti hukum atau medis, konsekuensinya bisa berbahaya. Oleh karena itu, etika penggunaan LLM menuntut adanya verifikasi silang oleh manusia sebelum output digunakan sebagai dasar pengambilan keputusan.



Halusinasi LLM

Tanggung Jawab Pengguna Vs Pengembang

Pertanyaan etis besar lainnya adalah siapa yang bertanggung jawab ketika LLM menghasilkan jawaban salah, menyesatkan, atau bahkan berbahaya. Pengembang model biasanya menyatakan bahwa tanggung jawab akhir berada di tangan pengguna, sementara pengguna beranggapan bahwa pengembang memiliki kewajiban moral dan hukum untuk memastikan sistem aman digunakan. Perdebatan ini menunjukkan adanya area abu-abu dalam etika AI: di

satu sisi diperlukan akuntabilitas teknis dari pengembang, di sisi lain literasi digital dan kesadaran kritis dari pengguna tidak dapat dikesampingkan.



Kasus Penyalahgunaan Generative AI, Pengguna Jadi Tersangka

9.2 Keamanan Interaksi dengan Model AI

Isu keamanan menjadi salah satu perhatian utama dalam penggunaan LLM. Model ini berinteraksi langsung dengan pengguna melalui teks, gambar, atau instruksi multimodal lainnya. Karena interaksi bersifat terbuka, terdapat sejumlah risiko yang perlu diantisipasi, mulai dari kebocoran data pribadi hingga penyalahgunaan sistem untuk tujuan berbahaya.

Prompt Injection dan Prompt Leaks

Salah satu ancaman yang semakin dikenal adalah prompt injection. Serangan ini dilakukan dengan menyisipkan instruksi tersembunyi dalam sebuah prompt sehingga model diarahkan untuk melakukan hal yang tidak diinginkan, seperti mengabaikan perintah keamanan atau mengakses informasi yang seharusnya dilindungi. Serangan serupa juga dapat memicu prompt leaks, yaitu kebocoran instruksi internal model atau konfigurasi sistem yang seharusnya tidak diketahui pengguna. Risiko ini menunjukkan bahwa meskipun AI tampak "taat", instruksi berbahaya yang dirancang dengan cerdas bisa mengeksploitasi kerentanannya.

Data Sensitif dan Privasi

LLM sering digunakan untuk mengolah informasi yang bersifat pribadi atau rahasia. Jika pengguna memasukkan data sensitif seperti identitas pribadi,

informasi medis, atau dokumen hukum, ada potensi kebocoran jika data tersebut disimpan, digunakan ulang, atau tidak diproses sesuai standar privasi. Beberapa penyedia LLM sudah menetapkan kebijakan bahwa data percakapan tidak digunakan untuk pelatihan lebih lanjut tanpa izin, namun risiko tetap ada apabila pengguna tidak memahami cara kerja penyimpanan data. Oleh sebab itu, prinsip *privacy by design* harus diterapkan dalam setiap interaksi dengan LLM.

Risiko Penggunaan dalam Sektor Kritis

LLM semakin banyak dipakai dalam bidang-bidang strategis, seperti hukum, medis, keuangan, dan pemerintahan. Namun, tingkat keandalan model belum cukup untuk sepenuhnya menggantikan profesional manusia di sektor-sektor tersebut. Kesalahan kecil dalam interpretasi data hukum dapat menimbulkan konsekuensi serius di pengadilan, sementara diagnosis medis yang keliru dapat membahayakan nyawa pasien. Oleh karena itu, penerapan LLM di sektor kritis seharusnya selalu berada di bawah pengawasan ketat ahli manusia.

Kontrol terhadap Konten Berbahaya

LLM juga dapat dimanfaatkan untuk menghasilkan konten berbahaya, misalnya panduan membuat bahan peledak, menyebarkan ujaran kebencian, atau memproduksi disinformasi berskala besar. Untuk mengurangi risiko tersebut, penyedia layanan AI biasanya menerapkan *content moderation* dan filter otomatis. Namun, sistem penyaringan ini tidak selalu sempurna; masih ada celah yang dapat dieksploitasi. Oleh karena itu, keamanan interaksi dengan model tidak hanya bergantung pada algoritma penyaring, tetapi juga pada tanggung jawab pengguna dalam membatasi konten yang dihasilkan.

Dengan mempertimbangkan berbagai risiko ini, keamanan interaksi dengan model AI harus dipahami sebagai tanggung jawab bersama: pengembang menyediakan perlindungan teknis, sementara pengguna wajib menjaga etika dan kehati-hatian dalam memberi instruksi maupun menggunakan output.

9.3 Hak Cipta, Kepemilikan, dan Lisensi

Perkembangan LLM menimbulkan perdebatan baru mengenai hak cipta dan kepemilikan hasil karya yang dihasilkan oleh kecerdasan buatan. Jika sebuah puisi, kode program, atau gambar dihasilkan oleh model AI, pertanyaan yang muncul adalah: siapakah pemilik karya tersebut, dan apakah karya itu dilindungi hukum hak cipta?

Status hukum output AI

Secara umum, hak cipta tradisional mensyaratkan adanya pencipta manusia. Banyak yurisdiksi, termasuk Amerika Serikat, Inggris, dan Uni Eropa, belum mengakui karya yang sepenuhnya dibuat oleh AI tanpa campur tangan manusia sebagai karya yang dilindungi hak cipta. Namun, jika ada kontribusi signifikan dari manusia dalam proses penciptaan (misalnya penyusunan prompt yang kompleks, pengeditan hasil AI, atau kurasi akhir), maka karya tersebut dapat diakui memiliki perlindungan hukum. Dengan demikian, keterlibatan manusia menjadi kunci dalam klaim kepemilikan.

Panduan resmi penyedia LLM

Setiap penyedia model besar memiliki kebijakan berbeda terkait kepemilikan dan hak cipta output:

- OpenAI (ChatGPT): OpenAI memberikan hak kepada pengguna untuk menggunakan hasil yang dihasilkan, baik untuk tujuan pribadi maupun komersial. Namun, pengguna bertanggung jawab penuh atas konten yang dihasilkan, termasuk memastikan tidak ada pelanggaran hak cipta pihak ketiga. OpenAI juga menegaskan bahwa tidak ada jaminan bebas dari klaim hak cipta atas hasil yang dibuat dengan ChatGPT.
- Google (Gemini): Google mengizinkan penggunaan hasil output untuk keperluan pribadi maupun komersial, dengan catatan pengguna wajib memastikan penggunaan tersebut tidak melanggar hukum atau hak kekayaan intelektual pihak lain. Google juga menekankan bahwa hasil generatif dapat berbeda kualitasnya dan memerlukan verifikasi manusia.
- Anthropic (Claude): Anthropic memberikan lisensi penggunaan output kepada pengguna, tetapi secara tegas melarang pemanfaatan Claude untuk kegiatan yang melanggar hukum, melanggar hak kekayaan intelektual, atau merugikan pihak lain. Anthropic menekankan prinsip Constitutional AI yang menekankan etika dan kepatuhan hukum.
- Model Sumber Terbuka (LLaMA, Mistral, Falcon, dll.): Model open-source biasanya dirilis dengan lisensi perangkat lunak seperti MIT, Apache 2.0, atau lisensi komunitas khusus. Lisensi ini memungkinkan penggunaan luas, termasuk komersial, tetapi tetap mengharuskan pengguna mematuhi hukum hak cipta dan regulasi setempat. Karena sifatnya yang terbuka, tanggung jawab pengguna lebih besar untuk memastikan bahwa output yang dihasilkan tidak melanggar hak pihak lain.

Contoh Kasus Hukum

Beberapa kasus hukum mulai muncul terkait karya yang dihasilkan AI. Misalnya, sengketa mengenai apakah gambar yang sepenuhnya dibuat oleh AI dapat didaftarkan sebagai karya berhak cipta ditolak oleh pengadilan di Amerika Serikat. Di Eropa, diskusi serupa juga berlangsung mengenai status “penulis” dalam konteks AI. Hal ini menunjukkan bahwa kerangka hukum internasional masih berkembang, dan aturan yang berlaku dapat berbeda antarnegara.

Rekomendasi Penggunaan LLM

Untuk meminimalkan risiko pelanggaran hak cipta, pengguna LLM sebaiknya selalu:

- Melakukan verifikasi sumber informasi yang digunakan AI.
- Menggunakan AI sebagai alat bantu, bukan sebagai satu-satunya pencipta.
- Memberikan atribusi bila diperlukan, terutama ketika hasil generatif mengandung unsur yang menyerupai karya berhak cipta.
- Memahami syarat dan ketentuan masing-masing penyedia LLM sebelum menggunakan hasilnya untuk kepentingan publik atau komersial.

Dengan demikian, persoalan hak cipta dalam konteks LLM menuntut pendekatan hati-hati yang menggabungkan kepatuhan hukum, etika penggunaan, dan literasi digital yang memadai.

9.4 Risiko Sosial dan Dampak Luas

Selain persoalan etika, keamanan teknis, dan hak cipta, penggunaan Large Language Models (LLM) juga menimbulkan dampak sosial yang signifikan. Risiko ini muncul bukan hanya dari interaksi langsung dengan teknologi, melainkan juga dari implikasi jangka panjang terhadap masyarakat, budaya, dan lingkungan.

Ketergantungan Berlebihan pada AI

Kemudahan akses terhadap LLM dapat mendorong pengguna untuk mengandalkan AI dalam hampir semua aktivitas, mulai dari menulis laporan hingga membuat keputusan kompleks. Jika ketergantungan ini terjadi tanpa keseimbangan, keterampilan berpikir kritis, kreativitas, dan kemampuan analitis manusia dapat tergerus. Risiko tersebut menciptakan generasi pengguna yang

lebih pasif, mengandalkan jawaban instan tanpa kemampuan mengevaluasi atau mengkritisi informasi.

Penyebaran Disinformasi dan Deepfake

LLM memiliki kemampuan menghasilkan teks, gambar, hingga video yang sangat meyakinkan. Kemampuan ini bisa dimanfaatkan secara positif, tetapi juga berpotensi disalahgunakan untuk menyebarkan disinformasi atau deepfake. Dalam konteks politik, hal ini dapat mengancam stabilitas demokrasi dan menurunkan kepercayaan publik terhadap informasi yang beredar. Di ranah sosial, penyebaran konten palsu dapat merusak reputasi individu maupun lembaga.

Etika Distribusi dan Akses

Teknologi AI generatif belum dapat diakses secara merata. Hanya negara atau kelompok tertentu yang memiliki sumber daya finansial dan infrastruktur memadai untuk memanfaatkan LLM secara penuh. Kondisi ini memperlebar kesenjangan digital antara masyarakat yang memiliki akses dan yang tertinggal. Kesenjangan tersebut tidak hanya berdampak pada ekonomi, tetapi juga memperkuat ketidakadilan dalam akses informasi dan kesempatan pendidikan.

Dampak Lingkungan

Pelatihan dan penggunaan LLM dalam skala besar memerlukan energi yang sangat tinggi. Data center yang mengoperasikan model-model ini berkontribusi pada emisi karbon dan konsumsi listrik global. Jika tidak dikelola secara berkelanjutan, dampak lingkungan dari teknologi AI dapat menimbulkan masalah baru yang berlawanan dengan tujuan pembangunan hijau. Isu ini mendorong diskusi mengenai efisiensi energi dalam desain model dan penggunaan sumber energi terbarukan untuk mendukung infrastruktur AI.

Secara keseluruhan, risiko sosial dan dampak luas dari LLM menuntut perhatian tidak hanya dari pengembang dan regulator, tetapi juga dari masyarakat sebagai pengguna. Literasi digital, regulasi yang tepat, serta kesadaran akan dampak lingkungan menjadi kunci untuk memastikan bahwa teknologi ini memberikan manfaat tanpa mengorbankan nilai-nilai sosial dan keberlanjutan.

9.5 Prinsip-Prinsip Desain Prompt yang Etis dan Aman

Perancangan prompt tidak hanya bertujuan untuk menghasilkan keluaran yang akurat, tetapi juga harus mempertimbangkan aspek etika dan keamanan. Mengingat LLM mampu memengaruhi opini, perilaku, bahkan pengambilan keputusan, penyusunan prompt yang sembarangan dapat menimbulkan dampak negatif. Oleh karena itu, diperlukan seperangkat prinsip agar interaksi dengan model tetap bertanggung jawab, transparan, dan aman digunakan dalam berbagai konteks.

Transparansi

Prompt sebaiknya dirancang dengan cara yang jelas dan dapat dipahami maksudnya, baik oleh model maupun oleh pihak lain yang mungkin meninjau hasilnya. Selain itu, penting untuk menyatakan secara terbuka apabila sebuah karya dihasilkan dengan bantuan AI. Transparansi ini membantu menjaga kepercayaan publik dan mencegah kebingungan terkait asal-usul informasi.

Akuntabilitas

Pengguna yang menyusun prompt bertanggung jawab penuh atas hasil yang dihasilkan. Tanggung jawab ini mencakup verifikasi kebenaran informasi, kesesuaian dengan hukum yang berlaku, serta dampak sosial dari distribusi konten. Prinsip akuntabilitas menegaskan bahwa AI adalah alat bantu, bukan entitas yang dapat dimintai pertanggungjawaban hukum atau moral secara mandiri.

Keadilan

Prompt tidak boleh dirancang dengan tujuan yang bias, diskriminatif, atau merugikan kelompok tertentu. Misalnya, permintaan untuk menggambarkan profesi tertentu dengan stereotip gender atau etnis berisiko memperkuat prasangka sosial. Prinsip keadilan menuntut perancang prompt untuk mengantisipasi kemungkinan bias dan mengupayakan representasi yang inklusif.

Privasi

Aspek privasi harus dijaga dengan tidak menyertakan data pribadi, informasi rahasia, atau dokumen sensitif dalam prompt. Risiko kebocoran informasi semakin besar ketika data tersebut diproses oleh layanan berbasis cloud atau

model yang menyimpan riwayat interaksi. Oleh sebab itu, prinsip kehati-hatian wajib diterapkan dalam setiap penyusunan prompt.

Ketaatan Hukum

Setiap prompt yang dibuat harus mematuhi aturan hukum yang berlaku, termasuk regulasi hak cipta, privasi data, dan larangan konten berbahaya. Menggunakan prompt untuk membuat konten ilegal, seperti instruksi pembuatan senjata atau penyebaran ujaran kebencian, tidak hanya melanggar kebijakan penyedia LLM, tetapi juga berpotensi menimbulkan konsekuensi hukum.

Dengan menerapkan prinsip-prinsip tersebut, praktik prompt engineering dapat berkembang menjadi keterampilan yang bukan hanya teknis, tetapi juga bertanggung jawab secara sosial. Prinsip ini pada akhirnya membantu memastikan bahwa teknologi AI generatif benar-benar menjadi sarana yang bermanfaat, aman, dan selaras dengan nilai-nilai etika masyarakat.

9.6 Rekomendasi

Setelah memahami tantangan etika, risiko keamanan, hak cipta, serta dampak sosial dari penggunaan Large Language Models (LLM), diperlukan langkah-langkah praktis untuk memastikan pemanfaatan teknologi ini tetap aman, etis, dan bertanggung jawab. Rekomendasi berikut ditujukan bagi berbagai pihak yang terlibat dalam penggunaan maupun pengembangan LLM.

Bagi Pengguna Umum

Pengguna individu perlu membiasakan diri untuk bersikap kritis terhadap setiap keluaran AI. Informasi yang dihasilkan hendaknya tidak langsung diterima begitu saja, tetapi diverifikasi dengan sumber lain yang lebih dapat dipercaya. Selain itu, pengguna sebaiknya tidak memasukkan data pribadi atau rahasia ke dalam prompt, guna mencegah potensi kebocoran informasi. Pemahaman terhadap syarat penggunaan dari penyedia model juga menjadi langkah penting untuk melindungi diri dari risiko hukum maupun etika.

Bagi Peneliti dan Pendidik

Kalangan akademik memiliki peran strategis dalam meningkatkan literasi digital dan literasi AI di masyarakat. Peneliti diharapkan tidak hanya mengembangkan teknik baru, tetapi juga mengeksplorasi dampak sosial dan etika dari teknologi ini. Pendidik dapat mengintegrasikan pembahasan tentang etika AI

dalam kurikulum, sehingga generasi baru tidak hanya mahir menggunakan AI, tetapi juga memahami implikasi moral dan hukum dari penggunaannya.

Bagi Bisnis dan Industri

Perusahaan yang menggunakan LLM untuk kepentingan operasional atau komersial sebaiknya memiliki kebijakan internal yang jelas mengenai penggunaan AI. Hal ini meliputi prosedur verifikasi hasil, pedoman privasi, serta mekanisme audit untuk memastikan kepatuhan terhadap regulasi. Selain itu, pelatihan bagi karyawan tentang penggunaan prompt yang efektif dan aman dapat membantu memaksimalkan manfaat teknologi tanpa melanggar prinsip etika maupun hukum.

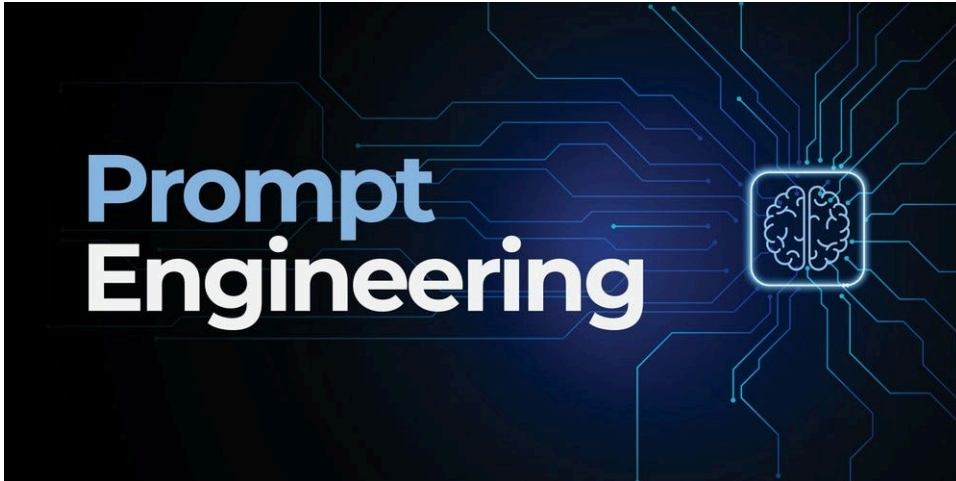
Bagi Pemerintah dan Regulator

Otoritas publik memiliki tanggung jawab untuk menyusun kerangka hukum dan regulasi yang mampu mengimbangi perkembangan teknologi AI. Regulasi tersebut perlu mencakup perlindungan privasi data, perlindungan hak cipta, pencegahan disinformasi, serta standar keamanan dalam penggunaan LLM di sektor-sektor vital. Selain itu, pemerintah dapat mendorong kolaborasi internasional untuk membangun norma global mengenai penggunaan AI yang aman dan etis.

Secara keseluruhan, keberhasilan dalam memanfaatkan LLM tidak hanya ditentukan oleh kemampuan teknis, tetapi juga oleh kesadaran etis dan kebijakan praktis yang mendukung. Dengan penerapan rekomendasi ini, potensi besar AI generatif dapat dimanfaatkan secara maksimal tanpa mengorbankan aspek keamanan, keadilan, dan tanggung jawab sosial.

BAB X

MASA DEPAN PROMPT ENGINEERING



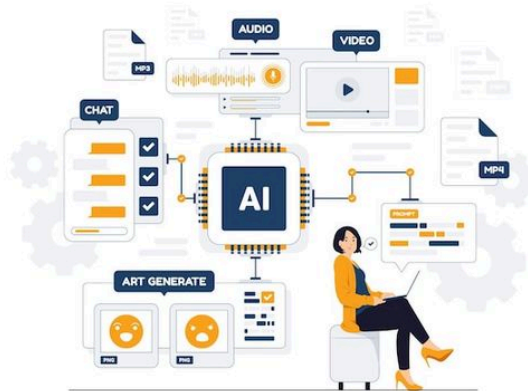
Perjalanan teknologi selalu ditentukan oleh cara manusia berkomunikasi dengan mesin. Dulu, manusia harus memahami bahasa mesin yang rumit berupa kode biner. Kemudian, lahir bahasa pemrograman yang lebih manusiawi seperti C, Python, atau Java. Kini, kita berada di era baru di mana manusia tidak lagi menyesuaikan diri dengan mesin, melainkan mesin yang beradaptasi dengan bahasa manusia. Inilah titik lahirnya prompt engineering.

Prompt engineering adalah seni sekaligus ilmu dalam menyusun instruksi yang efektif agar kecerdasan buatan, khususnya model bahasa besar (Large Language Models/LLM), mampu memahami, merespons, dan menghasilkan output sesuai harapan. Pada tahap awal kemunculannya, banyak orang menganggap prompt hanya sekadar “mengetik perintah ke chatbot”. Namun, seiring waktu, terbukti bahwa kualitas prompt sangat menentukan kualitas hasil AI.

Saat ini, prompt engineering masih berada pada fase awal perkembangan. Banyak praktisi yang bekerja secara trial and error: mencoba berbagai variasi instruksi, menguji hasil, lalu memperbaiki ulang. Meski demikian, peran prompt engineer semakin krusial karena mereka mampu “menerjemahkan” kebutuhan manusia ke dalam bahasa yang dimengerti AI.

Namun, bab ini tidak berhenti pada pemahaman dasar. Kita akan membahas masa depan prompt engineering: bagaimana profesi ini akan berubah, teknologi pendukung apa saja yang akan muncul, tantangan yang perlu diantisipasi, serta

bagaimana dunia akan beradaptasi dengan kehadiran sebuah keterampilan baru yang dapat menentukan arah peradaban digital.



10.1 Ke Depan, Ada Beberapa Arah Perkembangan Prompt Engineering:

1. Dari Teks ke Multimodal

Prompt tidak lagi terbatas pada teks. Instruksi bisa berbentuk suara, gambar, video, atau bahkan gerakan tubuh yang ditangkap sensor. Seorang desainer bisa “menggambar” idenya di udara, lalu AI langsung menerjemahkannya menjadi prototipe.

2. Profesi atau Otomatisasi?

Pertanyaan besar muncul: apakah prompt engineering akan menjadi profesi permanen seperti software engineering, atau justru lenyap ketika AI sudah cukup pintar memahami maksud manusia tanpa instruksi detail? Ada kemungkinan, prompt engineer berubah fungsi menjadi **AI strategist** atau **interaction designer** yang lebih fokus pada tujuan, bukan sekadar teknis perintah.

3. Integrasi dengan Dunia Nyata

Prompt akan terhubung langsung dengan sistem kehidupan sehari-hari: asisten rumah pintar, kelas digital di sekolah, layanan pemerintahan,

hingga pengambilan keputusan bisnis. Misalnya, seorang guru cukup berkata: *“Buatkan rencana pembelajaran untuk anak usia 10 tahun dengan tema energi terbarukan”*, dan sistem otomatis menghadirkan kurikulum interaktif yang sesuai.

4. **Tantangan Etis dan Regulasi**

Semakin kuat peran prompt engineering, semakin besar pula tantangan etis yang muncul. Bagaimana jika prompt disalahgunakan untuk manipulasi informasi, propaganda, atau eksploitasi data pribadi? Di sinilah dibutuhkan aturan main yang jelas agar masa depan prompt engineering tetap memberi manfaat bagi manusia.

5. **Bahasa Bersama antara Manusia dan Mesin**

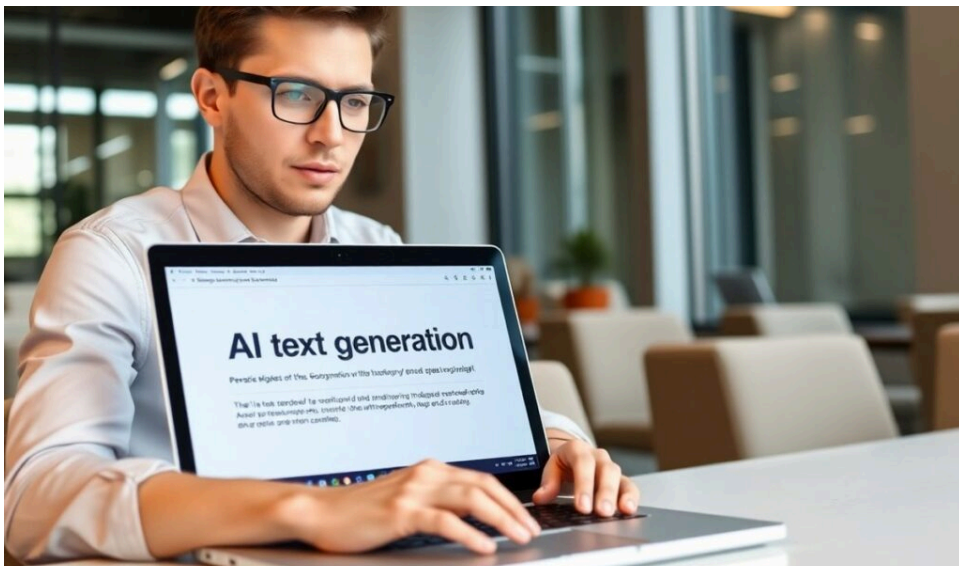
Lebih dari sekadar keterampilan teknis, prompt engineering adalah tonggak baru dalam sejarah komunikasi. Masa depan tidak lagi soal siapa yang lebih cerdas manusia atau mesin tetapi bagaimana keduanya bisa membangun **bahasa kolaboratif**. Inilah yang akan menentukan arah peradaban digital berikutnya.

10.2 Masa Depan Prompt Engineering: Dari Karier hingga Vibe Coding



Sejarah teknologi adalah kisah panjang tentang cara manusia berbicara dengan mesin. Awalnya, komunikasi hanya dimungkinkan melalui bahasa mesin yang keras, dingin, dan penuh logika biner. Kemudian, bahasa pemrograman hadir sebagai jembatan agar manusia bisa lebih “berbicara” dengan komputer, meski tetap penuh dengan sintaks yang ketat. Kini, kita memasuki era baru: era **prompt engineering**, di mana manusia tidak lagi dipaksa untuk menyesuaikan diri dengan mesin, melainkan mesin lah yang belajar memahami bahasa manusia.

10.2.1 Prompt Engineering sebagai Karier Baru



Prompt engineering muncul sebagai keterampilan yang sebelumnya tidak pernah dibayangkan. Jika software engineer dulu berfokus pada logika dan struktur kode, prompt engineer lebih fokus pada **bahasa, konteks, dan imajinasi**. Mereka merancang instruksi yang mampu mengarahkan AI untuk menghasilkan jawaban, karya, bahkan solusi nyata.

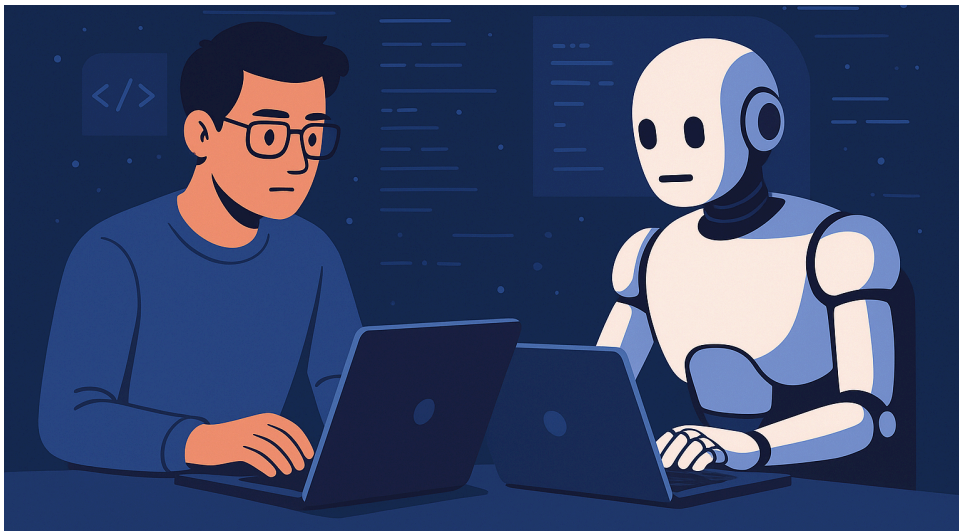
Karier ini masih sangat muda, tetapi potensinya sangat besar. Di masa depan, prompt engineer diperkirakan akan dibutuhkan di berbagai sektor:

1. **Pendidikan:** Guru dan dosen dapat mengandalkan prompt engineer untuk merancang bahan ajar interaktif, simulasi pembelajaran, atau kurikulum berbasis AI.

2. **Bisnis:** Perusahaan akan membutuhkan prompt engineer untuk menciptakan asisten virtual, riset pasar berbasis data AI, hingga strategi pemasaran kreatif yang digerakkan mesin cerdas.
3. **Pemerintahan:** Dalam pelayanan publik, prompt engineering bisa membantu menciptakan sistem birokrasi yang lebih cepat dan mudah dipahami oleh masyarakat.
4. **Kreativitas & Hiburan:** Dari penulisan naskah film hingga pengembangan game interaktif, prompt engineer bisa menjadi penghubung antara imajinasi manusia dengan kemampuan generatif AI.

Seiring waktu, posisi prompt engineer mungkin tidak lagi berdiri sendiri. Ia bisa bertransformasi menjadi **AI Interaction Designer**, **AI Strategist**, atau bahkan **AI Ethics** profesi yang lebih luas, strategis, dan menyentuh aspek sosial serta etika penggunaan kecerdasan buatan.

10.2.2 Dari Coding ke “Vibe Coding”



Dunia coding yang selama ini identik dengan baris-baris kode rumit sedang bergeser menuju **vibe coding**. Istilah ini merujuk pada pengalaman berinteraksi dengan AI tanpa harus menulis kode panjang. Cukup dengan perintah sederhana, deskripsi ide, atau bahkan sebuah contoh gambar, mesin akan “memahami nuansa” dan mewujudkannya dalam bentuk aplikasi, desain, atau sistem kerja.

Jika dulu seorang programmer perlu menguasai ribuan baris syntax Python, Java, atau PHP, kini dengan **vibe coding**, seseorang bisa membangun prototipe aplikasi hanya dengan berkata:

“Buatkan aplikasi pencatat keuangan sederhana dengan tampilan minimalis, ikon berwarna biru, dan fitur ekspor ke Excel.”

Dan dalam hitungan menit, AI akan menyajikan kerangka aplikasi yang bisa langsung dipakai atau dimodifikasi. Konsep ini akan mengubah cara kita memandang coding. Coding tidak lagi dipandang sebagai **teknik** semata, tetapi juga sebagai **ekspresi**. Sama seperti seorang musisi yang tidak lagi harus menulis notasi rumit untuk menciptakan musik, tetapi cukup bermain dengan nada hingga tercipta harmoni. Demikian pula vibe coding memungkinkan manusia “bermain” dengan ide hingga tercipta karya digital yang utuh.

10.2.3 Evolusi Hubungan Manusia dan Mesin

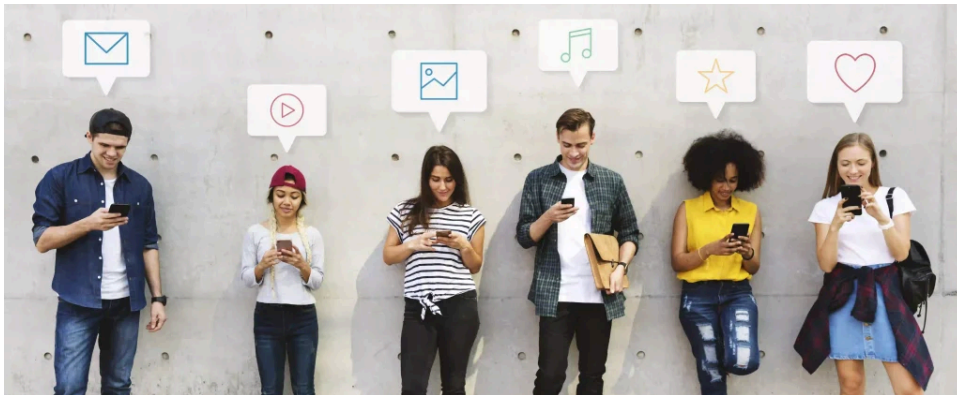


Masa depan prompt engineering bukan sekadar tren teknis, melainkan perubahan paradigma dalam hubungan manusia dan teknologi. Dengan berkembangnya AI multimodal yang mampu menerima teks, suara, gambar, dan gerakan tubuh prompt engineering tidak lagi terbatas pada kalimat, tetapi akan menjadi interaksi penuh.

Bayangkan seorang arsitek yang cukup menggambar sketsa kasar di udara menggunakan gesture tangan, lalu AI segera membangun model 3D yang detail. Atau seorang anak yang menceritakan dongeng dengan kata-kata polos, lalu AI

menghidupkannya dalam bentuk animasi interaktif. Di sinilah prompt engineering berfungsi sebagai **bahasa perantara** antara intuisi manusia dan logika mesin.

10.2.4 Tantangan Etis dan Sosial



Meski menjanjikan, perkembangan ini tidak lepas dari tantangan. Ada risiko **bias algoritmik**, manipulasi informasi, hingga hilangnya keterampilan tradisional karena manusia terlalu bergantung pada AI. Dunia akan membutuhkan prompt engineer yang bukan hanya mahir secara teknis, tetapi juga memiliki **integritas etis** dan kesadaran sosial yang tinggi.

Tantangan terbesar dari masa depan prompt engineering bukan hanya bagaimana menciptakan instruksi yang efektif, tetapi bagaimana memastikan bahwa teknologi ini tidak menjadi alat yang merugikan masyarakat. Dalam konteks global, muncul beberapa pertanyaan yang harus terus dijawab:

1. **Bagaimana memastikan bahwa AI tidak disalahgunakan melalui prompt yang menyesatkan?**
Sebuah prompt sederhana bisa digunakan untuk kebaikan, misalnya menciptakan materi edukasi. Namun, prompt yang sama bisa dipelintir untuk menciptakan hoaks, propaganda politik, atau bahkan serangan siber. Maka, diperlukan regulasi, transparansi, dan edukasi yang kuat agar AI tidak menjadi senjata informasi.
2. **Bagaimana menjaga orisinalitas karya manusia ketika AI semakin dominan?**

Dunia seni, literasi, dan desain kini menghadapi era di mana karya AI bisa menyaingi, bahkan melampaui, manusia. Pertanyaan kritis muncul: apakah karya tersebut masih dianggap sebagai ekspresi kreatif manusia, atau sekadar hasil manipulasi mesin? Di sinilah prompt engineer dituntut untuk menjaga keseimbangan antara memanfaatkan kecerdasan buatan dan mempertahankan ciri khas manusiawi dalam setiap hasil ciptaan.

3. **Bagaimana memastikan akses teknologi ini merata, bukan hanya untuk segelintir orang atau korporasi besar?**

Jika hanya perusahaan raksasa yang memiliki sumber daya untuk menguasai AI dan prompt engineering, maka kesenjangan digital akan semakin lebar. Akses yang inklusif sangat penting agar masyarakat luas, termasuk pelajar, UMKM, dan komunitas kecil, bisa ikut merasakan manfaatnya. Prompt engineering harus dikembangkan sebagai keterampilan bersama, bukan monopoli segelintir pihak.

Lebih jauh lagi, masa depan prompt engineering juga menyentuh aspek **identitas manusia**. Di satu sisi, AI bisa membantu manusia mengekspresikan ide lebih cepat dan lebih besar skalanya. Namun di sisi lain, jika tidak hati-hati, manusia bisa kehilangan jati dirinya karena terlalu larut dalam kenyamanan otomatisasi. Prompt engineering seharusnya menjadi **alat pemberdayaan**, bukan pengganti kreativitas manusia.

10.2.5 Masa Depan yang Kolaboratif



Masa depan prompt engineering adalah masa depan **kolaborasi**. Bukan sekadar mesin membantu manusia, tetapi manusia dan mesin saling melengkapi dalam menciptakan dunia baru. Karier ini akan terus berkembang, vibe coding akan menjadi arus utama, dan bahasa manusia akan menjadi alat paling kuat dalam berinteraksi dengan teknologi.

Pada akhirnya, masa depan prompt engineering bukan hanya tentang AI, melainkan tentang hubungan manusia dengan teknologi. Profesi ini bisa menjadi jembatan antara kreativitas manusia dan kecerdasan mesin membuka kemungkinan baru yang hari ini bahkan belum bisa kita bayangkan.

DAFTAR PUSTAKA

- [1] Ainsworth, M. (2025, July 22). A beginner's guide to AI image prompts & generation. Maxweb Solutions. <https://maxwebsolutions-co-uk.translate.goog/blog/beginners-guide-to-ai-image-prompts>
- [2] Anthropic. (2023). Prompt engineering guide. Anthropic Developer Docs. <https://docs.anthropic.com>
- [3] Anthropic. (2024). Claude 3 model card. <https://www.anthropic.com/news/claude-3>
- [4] Baltrušaitis, T., Ahuja, C., & Morency, L.-P. (2019). Multimodal machine learning: A survey and taxonomy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(2), 423–443. <https://doi.org/10.1109/TPAMI.2018.2798607>
- [5] Chen, B., Zhou, Z., Huang, Y., & Jiang, Y. (2023). Unleashing the potential of prompt engineering for large language models. *arXiv preprint arXiv:2310.14735*. <https://doi.org/10.48550/arXiv.2310.14735>
- [6] Cohere. (2024). Command R+ overview. <https://docs.cohere.com>
- [7] Google Cloud. (2024). What is prompt engineering? Google Cloud AI Documentation. <https://cloud.google.com/discover/what-is-prompt-engineering>
- [8] Google DeepMind. (2024). Gemini 1.5 technical overview. <https://deepmind.google/technologies/gemini>
- [9] 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 preprint arXiv:2307.12980*. <https://doi.org/10.48550/arXiv.2307.12980>
- [10] Hewing, M., & Leinhos, V. (2024). The prompt canvas: A literature-based practitioner guide for creating effective prompts in large language models. *arXiv preprint arXiv:2412.05127*. <https://doi.org/10.48550/arXiv.2412.05127>
- [11] Kurbanov, I. (2025, July 27). 5 key elements of an effective AI music prompt. Sendfame. <https://sendfame.com/blog/ai-music-prompt>
- [12] Meta AI. (2024). LLaMA 3 model card. <https://ai.meta.com/llama>
- [13] Micael. (2025, July 24). How to write effective text prompts to generate AI videos? FlexClip Help Center. <https://help-flexclip-com.translate.goog/en/articles/10326783-how-to-write-effective-text-prompts-to-generate-ai-videos>

- [14] Mistral AI. (2023). Introducing Mistral and Mixtral models. <https://mistral.ai/news>
- [15] OpenAI. (2023). GPT-4 technical report. <https://openai.com/research/gpt-4>
- [16] OpenAI. (2024). Prompt engineering. OpenAI Documentation. <https://platform.openai.com/docs/guides/prompt-engineering>
- [17] OpenAI. (2024). Introducing GPT-4o. <https://openai.com/index/gpt-4o>
- [18] Schulhoff, S., et al. (2024). The prompt report: A systematic survey of prompting techniques. arXiv preprint arXiv:2406.06608. <https://doi.org/10.48550/arXiv.2406.06608>
- [19] White, J., Fu, Q., Hays, S., Sandborn, M., Olea, C., Gilbert, H., Elnashar, A., Spencer-Smith, J., & Schmidt, D. C. (2023). A prompt pattern catalog to enhance prompt engineering with ChatGPT. arXiv preprint arXiv:2302.11382. <https://doi.org/10.48550/arXiv.2302.11382>
- [20] White, J., Hays, S., Fu, Q., Olea, C., & Schmidt, D. C. (2023). Prompt engineering for everyone: Patterns and practices for interacting with AI language models. Addison-Wesley Professional.
- [21] xAI. (2024). Grok AI assistant. <https://x.ai/blog/grok>
- [22] YouTube. (2025, July). Trik prompt Suno oleh Agus Hardiman [Video]. ArtSonica Channel. <https://www.youtube.com/watch?v=fL1hSUixqL4>

Perkembangan teknologi kecerdasan buatan, khususnya large language models (LLM), telah menghadirkan tantangan sekaligus peluang yang signifikan bagi dunia akademik, industri, dan masyarakat luas. Dalam konteks ini, prompt engineering menjadi instrumen strategis yang menentukan kualitas interaksi antara manusia dan mesin. Keterampilan dalam merumuskan instruksi yang tepat bukan sekadar aspek teknis, melainkan fondasi epistemologis yang memastikan keluaran sistem kecerdasan buatan selaras dengan kebutuhan pengguna.

Urgensi penguasaan prompt engineering semakin terasa karena kualitas hasil yang dihasilkan LLM secara langsung dipengaruhi oleh struktur, kejelasan, dan konteks instruksi yang diberikan. Dengan demikian, prompt engineering dapat dipandang sebagai keterampilan literasi baru yang tidak hanya relevan dalam bidang teknologi, tetapi juga penting dalam ranah pendidikan, penelitian ilmiah, serta pengambilan keputusan berbasis data. Pengabaian terhadap aspek ini berpotensi menimbulkan bias, kesalahan interpretasi, maupun ketidakefisienan dalam pemanfaatan kecerdasan buatan.

Buku ini disusun untuk memberikan landasan teoretis sekaligus panduan praktis mengenai prompt engineering. Melalui pendekatan sistematis, diharapkan pembaca memperoleh pemahaman yang komprehensif mengenai konsep, prinsip, dan aplikasi prompt engineering sehingga mampu mengoptimalkan pemanfaatan kecerdasan buatan dalam berbagai bidang. Dengan demikian, buku ini diharapkan dapat berkontribusi pada pengembangan literasi digital yang lebih kritis, reflektif, dan adaptif terhadap dinamika teknologi masa kini.



PT Solusi Administrasi Hukum

Jl. Pedan Karangdowo, Klaten
publisher.sah.co.id
publisher@sah.co.id

