

PROMPT ENGINEERING

WORKSHOP

Unlock the Power of AI Communication



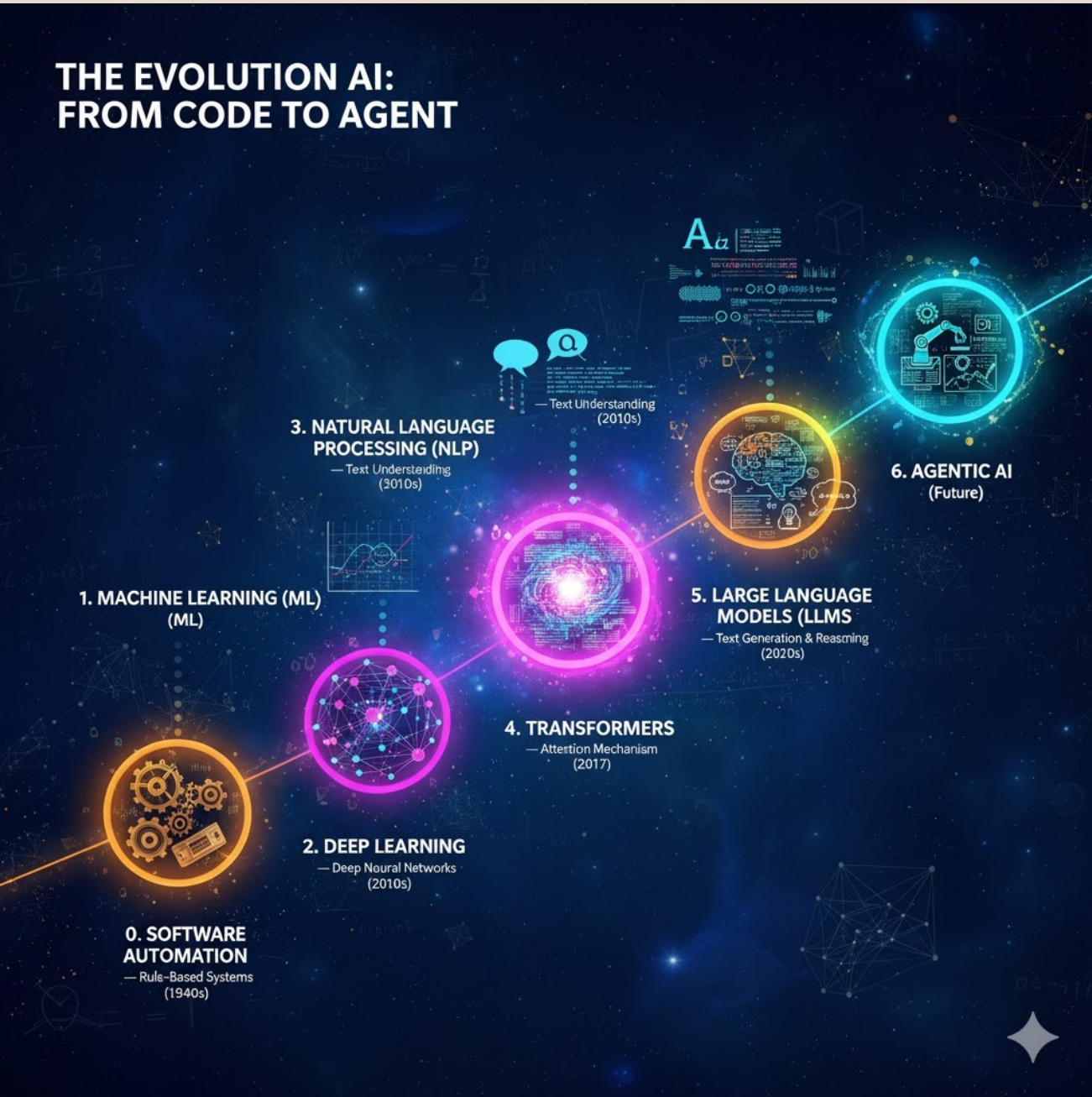
Presented by Sridhar Pai,
Data Scientist and AI Transformation Consultant
MS, Data Science.
Bengaluru, Karnataka, India
Contact number : +91-9886715008
Email : srinay2007@gmail.com

15 October 2025 4 PM - 5:30 pm

Prompt Engineering Workshop

- Artificial Intelligence Evolution
- Software 3.0 — A New Era in Programming, in Plain English.
- What is Prompt Engineering ?
- Key Skills in Prompt Engineering
- Controlling the LLM Output
- System, Contextual and role prompting
- Step-back prompting
- Essential Prompting Techniques
- Key Challenges and Solutions in Prompt Engineering
- Transliteration, Translation and Localization
- Konkani Proverbs Digitisation: Digitising Handwritten text in Konkani, Hindi and English.
- Major Freelance Marketplaces
- Summary & Q&A

THE EVOLUTION AI:
FROM CODE TO AGENT



Stage	What It Is (Simple Analogy)	Approximate Timeline	What It Does
0. Software Automation	The Handwritten Instructions 📝 : Writing rigid code that executes the exact same steps every time.	1950s – Present	Automates manual, repetitive tasks using explicit rules (e.g., a simple data migration script).
1. Machine Learning (ML)	The Basic Student 🎓 : Algorithms learn rules and patterns from data to make decisions without being explicitly programmed.	1980s – 1990s (Modern Resurgence)	Uses structured data to make predictions (e.g., credit scoring, simple classifications).
2. Deep Learning	The Advanced Brain 🧠 : Uses large, multi-layered digital networks to automatically find complex patterns.	2010 – 2012 (Breakthrough Era)	Excels at handling raw, unstructured data (e.g., recognizing faces in photos, understanding speech).
3. NLP (Natural Language Processing)	The Language Translator 🗣️ : Focuses on teaching computers to read, understand, and write human language.	2000s (Statistical Models)	Powers tasks like grammar checking, sentiment analysis, and basic machine translation.
4. Transformers	The Master Key 🔑 : A revolutionary architecture that lets AI process all words in a sentence at once, understanding their relationships instantly.	2017 (Architectural Shift)	Enabled the creation of truly massive models by allowing parallel training on huge datasets.
5. Large Language Models (LLMs)	The Expert Communicator 🗨️ : Massive Transformer models pre-trained on nearly the entire internet's text.	2020 – 2023 (Mainstream Adoption)	Can generate coherent text, summarize, answer complex questions, and write code.
6. Multi-modal Models	The Digital Senses 👁️ 👂 : LLMs that can handle and connect information from multiple data types (text, images, audio) at the same time.	2023 – 2024 (Current State-of-the-Art)	Allows you to ask questions about a photo or generate complex visual content.
7. Agentic AI	The Autonomous Assistant 🤖 : The ultimate step—models that can reason, plan, and take a series of independent actions to achieve a goal.	2024 – Future (Evolving Frontier)	Will be able to manage a project, execute complex coding tasks, or resolve customer issues entirely on its own.

Software 3.0 — A New Era in Programming, in Plain English.



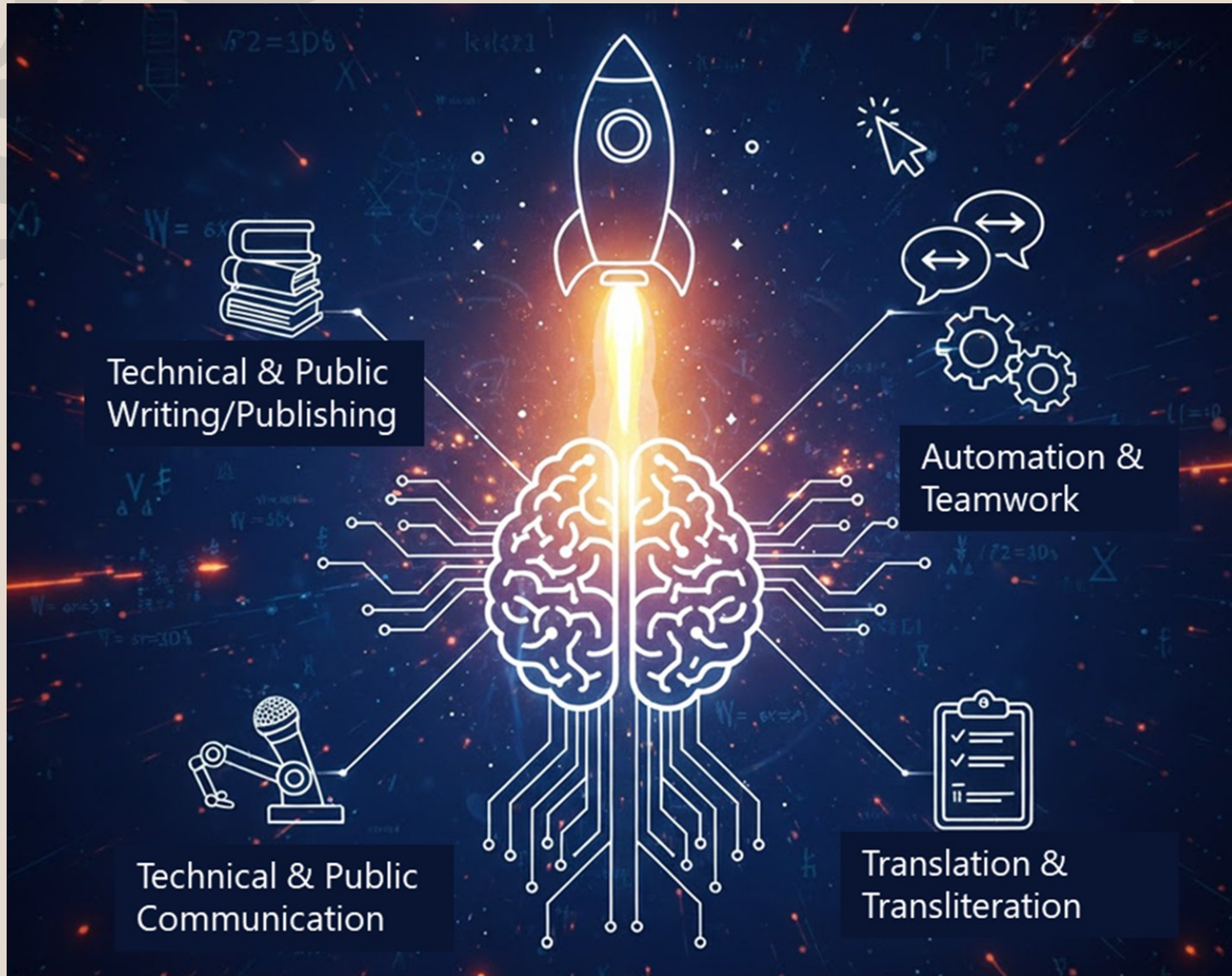
Andrey Karpathy - AI researcher; ex-Director of AI at Tesla; Cofounder of OpenAI; expert in deep learning & vision

We're living through a historic shift in software development. As he explains, software has evolved from 1.0 (human-written code) to 2.0 (neural networks trained on data), and now to 3.0: prompting large language models (LLMs)—where we program in natural language, not code.

LLMs like ChatGPT aren't just smart APIs—they're emerging as operating systems, central to a new computing stack. Apps like Cursor or Perplexity show us the future: partial autonomy, human-in-the-loop verification, custom GUI interfaces, and an "autonomy slider" that lets users decide how much control to delegate to the AI.

💡 Everyone who speaks English is now a "programmer." Tools like menu.app and vibe coding culture show how low the barrier has become. But as Karpathy warns, we're still in the "1960s of AI computing"—centralized, experimental, and full of quirks. Now is the time to build fast, build smart, and build with LLMs in the loop

Introduction to Prompt Engineering : Talking to AI



What is a **Large Language Model (LLM)**?

LLMs are Prediction Engines. They take text input (a prompt) and predict the next most likely token (word/part of a word) in a sequence. **They don't think, they predict.**

Eg: ChatGPT - <https://chatgpt.com/>

Gemini - <https://gemini.google.com/>

Meta AI - <https://web.whatsapp.com/>

- What is Prompt Engineering?

Prompt Engineering is the **process of designing high-quality prompts to guide LLMs to produce accurate and desired outputs.** It's an **Iterative Process** — **tinkering is key!**

Everyone can write prompt. No coding skills are required.

The art of crafting clear instructions for AI tools to deliver specific business or communication outcomes

Used in automation, translation, support, technical workflows

Used for text summarization, information extraction, question and answering, text classification, language or code translation, code generation, and code documentation or reasoning

Key Skills in Prompt Engineering

- Clarity and structure in instructions
Emphasize the need for clarity—avoiding vague or ambiguous language.
- Defining output format (summary, table, code, translation)
Output format (bulleted list, table, code snippet) affects usefulness.
- Specifying language and audience
the impact of specifying languages, persona, in multilingual environments.
- Iterative prompt refinement
Advise tailoring prompts for intended audience: technical team, support staff, management



Controlling the LLM Output

LLM Configuration Besides the prompt itself, you control the LLM's output using configuration settings. We control randomness and vocabulary.

Note: The AI chatbots (chatgpt, gemini) are designed for the best possible general-purpose conversation. This is useful when you want control LLM using APIs. For eg. You are building custom application and using Open AI api or Gemini AI API thru the code.

Temperature (Creativity) : Controls the degree of randomness in token selection.

Example :

explain sunrise over a city with

“Use a high temperature (be creative and free-flowing)”

and

“Use low temperature (be concise and factual).”

Low Temp (e.g., 0 or 0.1): More deterministic, factual, and predictable. Great for math or summarization.

High Temp (e.g., 0.8-1.0): More diverse, unexpected, and creative results. Great for brainstorming or fiction.

Top-K and top-P (Top K most likely tokens from the model's predicted distribution. Top P the top tokens whose cumulative probability does not exceed a certain value (P))

Example: “Once upon a time, there was a brave knight who ...”

Top-K (K = 3):

Once upon a time, there was a brave knight who fought the dragon.

Top-P (p = 0.80):

Once upon a time, there was a brave knight who loved a princess.



System, contextual and role prompting

System prompting: sets the overall context and purpose for the language model

Contextual prompting: provides specific details or background information relevant to the current conversation or task

Role prompting: assigns a specific character or identity for the language model to adopt

System prompting Example:

Classify movie reviews as positive, neutral or negative. Only return the label in uppercase.

Review: "Her" is a disturbing study revealing the direction humanity is headed if AI is allowed to keep evolving, unchecked. It's so disturbing I couldn't watch it.

Sentiment:

Output NEGATIVE

Role prompting Example:

I want you to act as a travel guide. I will write to you about my location and you will suggest 3 places to visit near me. In some cases, I will also give you the type of places I will visit.

My suggestion: "I am in Amsterdam and I want to visit only museums."

Travel Suggestions:

Output

1. ...
2. ...
3. ...

Contextual prompting Example:

Context: You are writing for a blog about retro 80's arcade video games.

Suggest 3 topics to write an article about with a few lines of description of what this article should contain.

Output

1. ...
2. ...
3. ...

Step-back prompting

A technique for improving the performance by prompting the LLM to first consider a general question related to the specific task at hand, and then feeding the answer to that general question into a subsequent prompt for the specific task.

Example Prompt for self consistency:

Based on popular first-person shooter action games, what are 5 fictional key settings that contribute to a challenging and engaging level storyline in a first-person shooter video game?

Output :

1. ****Abandoned Military Base****: A sprawling, post-apocalyptic military complex crawling with mutated soldiers and rogue robots, ideal for challenging firearm combat.
2. ****Cyberpunk City****: A neon-lit, futuristic urban environment with towering skyscrapers and dense alleyways, featuring cybernetically enhanced enemies and hacking mechanics.
3. ****Alien Spaceship****: A vast alien vessel stranded on Earth, with eerie corridors, zero-gravity sections, and extraterrestrial creatures to encounter.
4. ****Zombie-Infested Town****: A desolate town overrun by hordes of aggressive zombies, featuring intense close-quarters combat and puzzle-solving to find safe passage.
5. ****Underwater Research Facility****: A deep-sea laboratory flooded with water, filled with mutated aquatic creatures, and requiring stealth and underwater exploration skills to survive.

Output:

In the heart of a murky abyss, lies a dilapidated underwater research facility, standing as a testament to human ambition and its disastrous consequences. Shrouded in darkness, pulsating with the hum of malfunctioning machinery, and haunted by the echoes of drowned souls, the facility is now a breeding ground for a nightmare borne from the deep sea. The player, an elite marine equipped with advanced diving gear and experimental weaponry, is tasked with venturing into this aquatic realm of terror. Armed with resilience and resourcefulness, they must navigate treacherous corridors, solve cryptic puzzles, and confront gruesome sea monstrosities that lurk in the depths. From grappling with the crushing pressure of the deep sea to outsmarting cunning aquatic predators, every moment in this uncharted underworld tests the limits of human endurance and courage.

Traditional Prompt Example :

Write a one paragraph storyline for a new level of a first-person shooter video game that is challenging and engaging.

Output :

The level begins with the player's squad being ambushed in a dense urban area by a relentless enemy faction. Players must fight their way through a maze of back alleys and dilapidated buildings, utilizing stealth and precision to eliminate enemy forces and avoid detection. The player must use their environment to their advantage, setting traps, utilizing cover, and making calculated moves to survive. Along the way, they uncover intel suggesting the enemy is planning a devastating attack, and the player must race against time to gather evidence and relay it back to their headquarters before the enemy can carry out their plans.

Example Prompt:

Context: 5 engaging themes for a first person shooter video game:

1. ****Abandoned Military Base****: A sprawling, post-apocalyptic military complex crawling with mutated soldiers and rogue robots, ideal for challenging firearm combat.
2. ****Cyberpunk City****: A neon-lit, futuristic urban environment with towering skyscrapers and dense alleyways, featuring cybernetically enhanced enemies and hacking mechanics.
3. ****Alien Spaceship****: A vast alien vessel stranded on Earth, with eerie corridors, zero-gravity sections, and extraterrestrial creatures to encounter.
4. ****Zombie-Infested Town****: A desolate town overrun by hordes of aggressive zombies, featuring intense close-quarters combat and puzzle-solving to find safe passage.
5. ****Underwater Research Facility****: A deep-sea laboratory flooded with water, filled with mutated aquatic creatures, and requiring stealth and underwater exploration skills to survive.

Take one of the themes and write a one paragraph storyline for a new level of a first-person shooter video game that is challenging and engaging.

Essential Prompting Techniques

Types of Prompts & Approaches

Zero-shot: No prior example

Prompt:

Summarize this support ticket in English:
"Unable to connect to the office Wi-Fi since this morning.
Tried restarting my laptop and router, but nothing worked."

Output:

The user cannot connect to the office Wi-Fi today, and restarting both their laptop and router did not resolve the issue.

Few-shot: With examples

Translate the following helpdesk queries from English into Hindi and Kannada.

Examples:

Input (English): My internet is running very slow.

Output:

English: My internet is running very slow.

Hindi: मेरा इंटरनेट बहुत धीमा चल रहा है।

Kannada: ನನ್ನ ಇಂಟರ್ನೆಟ್ ತುಂಬಾ ನಿಧಾನವಾಗಿ ನಡೆಯುತ್ತಿದೆ.

Input (English): The printer is not working.

Output:

English: The printer is not working.

Hindi: प्रिंटर काम नहीं कर रहा है।

Kannada: ಪ್ರಿಂಟರ್ ಕೆಲಸ ಮಾಡುತ್ತಿಲ್ಲ.

Input (English): The laptop keeps restarting repeatedly.

Output:

English: The laptop keeps restarting repeatedly.

Hindi: लैपटॉप बार-बार रीस्टार्ट हो रहा है।

Kannada: ಲ್ಯಾಪ್ಟಾಪ್ ಮತ್ತೆ ಮತ್ತೆ ರೀಸ್ಟಾರ್ಟ್ ಆಗುತ್ತಿದೆ.

Chain of Thought prompting

Why Chain of Thought prompting will be used:

Chain of Thought prompting involves breaking down a **complex question** or **task into a series of interconnected prompts and responses**. This method is suitable for scenarios where there is a need to generate explanations or reasoning behind various options or decisions.

By using a series of interconnected prompts, the model can provide detailed explanations for each quiz option, both correct and incorrect. This allows learners to understand the rationale behind each answer choice, enhancing their comprehension and learning experience.

Chain of Thought Example

Prompt :

When I was 3 years old, my partner was 3 times my age. Now, I am 20 years old. How old is my partner? Let's think step by step.

Output :

1. Let's assume my current age is "x" years.
2. When I was 3 years old, my age was 3 years.
3. My partner's age at that time was 3 times my age, which means my partner's age was $3 * 3 = 9$ years.
4. Now, I am 20 years old, which means my age has increased by $20 - 3 = 17$ years since I was 3 years old.
5. Since my partner's age was 9 years when I was 3 years old, and my age has increased by 17 years, my partner's age must have also increased by 17 years.
6. Therefore, my partner's current age is $9 + 17 = 26$ years.
So, my partner is 26 years old.

Chain Of Thought Example

*You are part of a large organization training and development team. You have to create an assessment for a topic in the form of a Quiz. A few rules to follow - the quiz must have at least 10 questions, each question should have one correct answer. Use Chain-of-Thought prompting to get explanation for the correct and incorrect quiz options. Also, make sure to create a balanced quiz in terms of difficulty. Here pdf file attached on **prompt engineering techniques**. Prepare questions such that 25% complex, 40% moderately difficult 35% simple. Provide answers and reasoning for the answers.*

Instructions

- Open Gemini AI.
- Attach prompt engineering pdf.
- Run the above prompt in Gemini AI.



Some more Examples

Prompt : EMAIL:

...

Hi,

I have seen you use Wordpress for your website. A great open source content management system. I have used it in the past too. It comes with lots of great user plugins. And it's pretty easy to set up.

I did notice a bug in the contact form, which happens when you select the name field. See the attached screenshot of me entering text in the name field. Notice the JavaScript alert box that I inv0k3d.

But for the rest it's a great website. I enjoy reading it. Feel free to leave the bug in the website, because it gives me more interesting things to read.

Cheers,

Harry the Hacker.

...

Classify the above email as IMPORTANT or NOT IMPORTANT.

Let's

think step by step and explain why.

Instructions

- Open Gemini AI
- Type the prompt
- Review the output



Key Challenges and Solutions in Prompt Engineering

Challenge 1: Hallucination (Factual Errors) 🤪

The Problem	The Solution
<p>LLMs invent plausible-sounding but false information. (The #1 threat to trust and reliability)</p> <p>How it happens: Model overconfidence or lack of specific training data.</p>	<p>Implement Grounding & Verification</p> <p>1. Retrieval-Augmented Generation (RAG): Connect the model to a verified, external knowledge base.</p> <p>2. Chain-of-Verification (CoV): Instruct the model to check its own facts step-by-step.</p> <p>3. Explicit Constraints: Tell the model: <i>"Answer only with the provided text. If you don't know, say 'I cannot find that information.'"</i></p>

Prompt (Input)	Model Output (Hallucination)	The Reality
"List three research papers by Dr. Emily Carter on quantum computing, with titles and URLs."	"1. 'Qubit Entanglement in Hyper-Dimensional Space,' published in <i>Journal of Physics</i> , 2023. [link to a non-existent URL: www.fakequantumjournal.com/carter/paper1]"	The papers, the journal, and the specific links do not exist. The model synthesized plausible-sounding but false bibliographic data.

Challenge 2: Ambiguity & Inconsistency ?

The Problem	The Solution
<p>Vague Prompts: Unclear instructions lead to irrelevant, general, or unfocused outputs.</p> <p>Inconsistent Tone: The model switches style (e.g., from professional to casual) between responses.</p>	<p>Prioritize Clarity and Role-Playing</p> <p>1. Specificity: Define the goal, format, length, and audience. (e.g., <i>"Write a 5-bullet summary for executives."</i>)</p> <p>2. Few-Shot Prompting: Provide high-quality input/output examples to set a clear pattern.</p> <p>3. Persona-Driven Prompting: Instruct the model to adopt a specific role: <i>"Act as a lead data scientist."</i></p>

Prompt (Input)	Model Output (Problem)	Problem Description
"Explain inflation."	Output 1 (Too Technical): "Inflation is defined as the sustained increase in the general price level of goods and services in an economy over a period of time, measured by indices such as the CPI and PCE price index, and primarily driven by demand-pull or cost-push factors."	The response is too academic and uses jargon. The user didn't specify the target audience (e.g., a child, an economist).
Solution (Specific Prompt):	"Explain inflation in three concise bullet points, targeting a small business owner who needs to understand its immediate impact on inventory costs."	The specific prompt forces a targeted, useful response by defining the format (3 bullet points), length (concise), and audience/context (small business/inventory costs).

Challenge 3: Handling Complexity & Context 🧠

The Problem	The Solution
<p>Complexity: LLMs struggle with multi-step reasoning or tasks that require deep logic.</p> <p>Overwhelming Context: Too much input data (exceeding token limits) causes the model to lose focus or error out.</p>	<p>Enforce Structured Reasoning</p> <p>1. Chain-of-Thought (CoT): Force the model to think step-by-step by including the instruction: "Show your work and explain your reasoning."</p> <p>2. Iterative Prompting: Break large tasks or context into smaller, manageable chunks across multiple turns.</p>

Challenge 4: Security (Injection & Jailbreaking) 🛡️

The Problem	The Solution
<p>Prompt Injection: Malicious user input attempts to <i>override</i> the system's core, hidden instructions.</p> <p>Jailbreaking: User input attempts to <i>bypass</i> the model's safety/ethical guardrails to generate prohibited content.</p>	<p>Implement Security Layers</p> <p>1. Input-Output Separation: Use clear delimiters (e.g., <<<) to strictly separate trusted system instructions from variable user data.</p> <p>2. Reinforced Prompts: Place critical, explicit safety instructions in a system prompt and tell the model to never ignore them.</p> <p>3. Output Monitoring: Use external filters to scan the model's output for keywords or patterns that indicate a security breach.</p>

1. Transliteration 🗒️

What it is: The process of converting text from one script (or writing system) to another, based on sound rather than meaning. It focuses on preserving the pronunciation.

In Prompt Engineering:

- **Goal:** To allow users who speak a language (e.g., Arabic, Hindi) but may not have the corresponding keyboard script to input their prompts using a more common script (e.g., Latin/English alphabet).
- **Example:** Converting the Hindi name "मोहन" into the English alphabet as "Mohan." The meaning is not changed, only the way the word is spelled based on how it sounds.
- **Prompt Use Case:** A user might write their prompt, "kya time hua hai?" (a Latin script representation of the Hindi phrase "What is the time?"). The prompt engineer might design a system that internally transliterates this back to the correct script or uses the phonetic spelling to help the LLM better understand the user's intent, especially for proper nouns or technical terms

2. Translation 🗣️

What it is: The process of converting the meaning of text from a source language to a target language. It focuses on linguistic equivalence.

In Prompt Engineering:

- **Goal:** To ensure the core instructions, context, and desired output of a prompt are accurately understood by the LLM when interacting in a language different from the one the prompt was originally written in (often English).
- **Example:** Converting the English phrase "Write a summary of the report." to the French phrase "Rédigez un résumé du rapport."
- **Prompt Use Case:** If you have a powerful "System Prompt" designed in English (e.g., defining the AI's role, safety guidelines, and output format), you would use translation to create equivalent prompts for users interacting in Spanish, German, or Japanese, ensuring all users receive the same quality of instruction-following

3. Localization (L10N) 🌐

What it is: The comprehensive process of adapting a prompt, product, or service to a specific regional market. It goes beyond mere translation to include cultural, legal, and functional appropriateness.

Prompt Engineering:

Goal: To make the LLM's responses feel natural, relevant, and culturally appropriate to the local user.

Example: A translation of the English phrase "Call me in the fall" to Spanish is "Lláname en el otoño." A proper localization for a user in Australia (where "fall" is "autumn" but is in a different month) or a user in Latin America (where different slang or terminology might be used) ensures the output is contextually accurate.

Prompt Use Case: This is the most complete method. It involves

- Translating the prompt text.
- Adjusting the tone (e.g., formal vs. casual address).
- Ensuring cultural sensitivity (e.g., avoiding certain idioms, using the correct local terms for holidays or sports).
- Using local units of measurement (Fahrenheit vs. Celsius, miles vs. kilometers)

Konkani Proverbs Digitisation: Digitising Handwritten text in Konkani, Hindi and English.

This project was envisaged to spread and preserve authentic Konkani Proverbs. Earlier days these proverbs were extensively used by our ancestors in their regular conversations. But slowly it has lost the place due to the extensive usage of English and other languages. But it is important to learn and preserve this knowledge for next generation.

Team of 3 people started worked in this project. Based on handwritten text of proverbs by Author, we digitized 210 proverbs in 2 weeks of time. This involved 3 languages ie Konkani, Hindi and English.

Below are few examples.

एवढो विक्कुनु आयिलो दुड्डु खोर्जु खाया.
आवळो विक्कुनु आयिलो दुड्डु खोर्जु खाया.
Honesty earned money does not irritate.
The ill-gotten money always pricks the conscience.

रात्री पळैलेले बांस्तु दिसा वच्चुनु पडता.
रात्री पळैलेले बांस्तु दिसा वच्चुनु पडता.
To commit a mistake despite being aware of the consequences, often we commit such blunders knowingly.

बडिद दीन्तु पेट्टु खावचें.
बडिद दीन्तु पेट्टु खावचें.
Help others and in return get into problem or cause injury to oneself.

ಅವಳೊ ವಿಕ್ಕುನು ಆಯ್ಕಿಲೊ ದುಡ್ಡು ಖೊರ್ಜು ಖಾಯ್ನಾ.

आवळो विक्कुनु आयिलो दुड्डु खोर्जु खाया ।

Honestly earned money does not irritate. Ill-gotten money always pricks the conscience.

रात्री पळैलेले बांस्तु दिसा वच्चुनु पडता.

रात्री पळैलेले बांस्तु दिसा वच्चुनु पडता ।

To commit a mistake despite being aware of the consequences. We often commit such blunders knowingly.

ಬಡ್ಡಿ ದೀವ್ವು ಪೆಟ್ಟು ಖಾವ್ವೆ.

बडिद दीन्तु पेट्टु खावचें ।

Help others and in return get into problem or cause injury to oneself.

2 of our Team members will share some experience working in this project. I request them to introduce first and then share their experience.

Major Freelance Marketplaces

These platforms offer a wide variety of writing and language jobs, including translation and localization projects.

Platform	Key Features & Focus	Best For
Upwork http://www.upwork.com	A global marketplace with a vast number of clients. Jobs range from simple document translation to complex software localization contracts. You can set your own rates and submit proposals.	Freelancers with a strong portfolio and those seeking longer-term, hourly, or fixed-price contracts.
Fiverr http://www.fiverr.com	Focuses on "Gigs" (pre-packaged services with set prices). Translators and localizers create service offerings (e.g., "I will translate 500 words from Spanish to English in 24 hours").	Freelancers who want to quickly showcase their services and control their pricing and deliverables.
Freelancer.com	Similar to Upwork, offering a broad range of project types, often utilizing a bidding system where freelancers compete for jobs.	Beginners looking to build experience and those who prefer a wide variety of short-term projects.
PeoplePerHour	A UK-based platform known for filtering freelancer profiles to ensure quality. Offers both short-term projects and long-term contracts.	Experienced freelancers looking for high-quality, long-term translation or localization projects.

Here are few great references for Prompt Engineering learning:

Resource Name	Description	URL
Prompt Engineering Guide	A free, open-source, and highly comprehensive handbook covering foundational concepts, advanced techniques (like Chain-of-Thought, ReAct), and the latest research papers.	https://www.promptingguide.ai/
ChatGPT Prompt Engineering for Developers	A highly-rated, short course taught by DeepLearning.AI and OpenAI experts, focusing on practical best practices for building applications using LLM APIs (Summarizing, Inferring, Transforming).	https://www.deeplearning.ai/short-courses/chatgpt-prompt-engineering-for-developers/
OpenAI Prompt Engineering Best Practices	Official documentation from OpenAI detailing crucial strategies for interacting with their models, including principles like using delimiters and few-shot examples.	https://help.openai.com/en/articles/6654000-best-practices-for-prompt-engineering-with-the-openai-api
Vanderbilt University Prompt Engineering Specialization	A multi-course series on Coursera that provides a structured, academic approach to learning various prompt patterns and how to apply them to build LLM applications.	https://www.coursera.org/learn/prompt-engineering

Summary & Q&A

Prompt engineering is an art and a science—start simple, experiment, and pay attention to the results! Thank you

Feedback Survey:



Q & A

Thank you.



Linkedin: <https://www.linkedin.com/in/sridharpai/>

Data Scientist and AI Transformation Consultant

MS, Data Science,

Bengaluru, Karnataka, India

Contact number : +91-9886715008

Email : srinay2007@gmail.com