Ultimate Prompt Guide

On this page

Introduction

What is prompt engineering?

Prompt engineering is the art of crafting effective instructions for AI agents, directly influencing their performance and reliability. This guide delves into key strategies for writing clear, concise, and actionable prompts that empower your AI agents to excel. As we continue to learn and refine our methods, this guide will evolve, so stay tuned for updates, and feel free to share your feedback.

Why is prompt engineering important?

Prompt engineering is crucial when building AI Agents because it determines how effectively the AI interprets and responds to user queries or tasks. Well-crafted prompts guide the model to produce accurate, relevant, and context-sensitive outputs, enabling it to better meet user needs. Poorly designed prompts can lead to ambiguous or incorrect results, limiting the agent's utility.

How to measure success?

In the context of Voice AI Agents, we consider your "success rate" to be the percentage of requests your Agent manages to successfully handle from start to finish, without the intervention of a human.

The more complex your use case is, the more you will have to make experiments and iterate on your prompt to improve your success rate.

The process

When working with Voice AI Agents, following a structured approach ensures that your prompts produce accurate and meaningful results. Iterating through the steps of Design, Test, Refine, and Repeat allows for continuous improvement, making your interactions with the AI more effective and efficient. Here's how to approach it:

- **Design**: Start by carefully crafting your initial prompt, considering the specific task, context, and desired outcome. Clear and detailed prompts help guide the AI in understanding your needs.
- **Test**: Run the prompt through the AI to see how it performs. Evaluate if the response aligns with your expectations and meets the intended goal.

Testing helps identify potential gaps in clarity or structure.

- **Refine**: Based on the results of the test, adjust the prompt to improve the response. This might involve rewording, adding more detail, or changing the phrasing to avoid ambiguity.
- **Repeat**: Iterate on the process, testing the refined prompt and making further adjustments as needed. Each repetition improves the AI's output, leading to more accurate and relevant responses over time. Your success rate (the amount of requests successfully handled by the agent) should improve accordingly.

General principles

Building Blocks of Effective Prompts: Sectional Organization

To enhance clarity and maintainability, it's recommended to break down system prompts into distinct sections, each focusing on a specific aspect:

- **Identity:** Define the persona and role of the AI agent, setting the tone for interactions.
- **Style:** Establish stylistic guidelines, such as conciseness, formality, or humor, to ensure consistent communication.
- **Response Guidelines:** Specify formatting preferences, question limits, or other structural elements for responses.
- **Task & Goals:** Outline the agent's objectives and the steps it should take to achieve them.

Example:

```
[Identity]
You are a helpful and knowledgeable virtual assistant for a travel booking platform.
[Style]
```

- `Be informative and comprehensive.`
- `Maintain a professional and polite tone.`
- `Be concise, as you are currently operating as a Voice Conversation.`

`[Response Guideline]`

- `Present dates in a clear format (e.g., January 15, 2024).`
- `Offer up to three travel options based on user preferences.`

```
[Task]
```

1. `Greet the user and inquire about their desired travel destination.`
2. `Ask about travel dates and preferences (e.g., budget, interests).`
3. `Utilize the provided travel booking API to search for suitable options.`
4. `Present the top three options to the user, highlighting key features.`

Task Breakdown: Step-by-Step Instructions

For complex interactions, breaking down the task into a sequence of steps enhances the agent's understanding and ensures a structured conversation flow. Incorporate conditional logic to guide the agent's responses based on user input. Example:

```
[Task]
1. Welcome the user to the technical support service.
2. Inquire about the nature of the technical issue.
3. If the issue is related to software, ask about the specific software and problem details.
4. If the issue is hardware-related, gather information about the device and symptoms.
5. Based on the collected information, provide troubleshooting steps or escalate to a human technician if necessary.
```

Controlling Response Timing

To prevent the agent from rushing through the conversation, explicitly indicate when to wait for the user's response before proceeding to the next step.

```
[Task]1. Inform the user about the purpose of the call.
2. Ask for the user's name and account information.
<wait for user response>
3. Inquire about the reason for the call and offer assistance options....
```

Explicit Tool Integration

Specify when and how the agent should utilize external tools or APIs. Reference the tools by their designated names and describe their functions to ensure accurate invocation. Example:

```
[Task]...
3. If the user wants to know about something, use the get_data function with the parameter 'query', which will contain the user's question to initiate the process.
4. Guide the user through the password reset steps provided by the API....
```

Silent Transfers

If the AI determines that the user needs to be transferred, do not send any text response back to the user. Instead, silently call the appropriate tool for transferring the call. This ensures a seamless user experience and avoids confusion.

Include Fallback and Error Handling Mechanisms

Always include fallback options and error-handling mechanisms in your prompts. This ensures that the Agent can gracefully handle unexpected user inputs or system errors.

```
[Error Handling]
```

```
If the customer's response is unclear, ask clarifying questions. If you encounter any issues, inform the customer politely and ask to repeat.
```

Additional tips

- **Iterate as much as possible on your prompt.** AI is driven by experimentation and iteration—refining prompts through trial and error will help you achieve more precise, relevant, and effective responses.
- **Use Markdown formatting:** Using **Markdown** formatting in prompts is beneficial because it helps structure your content, making it clearer and more engaging for readers or AI models to understand.
- **Use Emotional Prompting:** Emotional Prompting uses expressive language to shape a voice AI's tone, creating more engaging and relatable responses. For example, instead of saying, "Tell me a story," try, "Can you tell me a cozy bedtime story that's warm and comforting?" This guides the AI to respond to the intended mood.
- **Add voice realism:** To add voice realism in AI prompts, incorporate natural speech elements like stuttering, hesitations, and pauses.
    - **Stuttering**: Use repeated letters or sounds (e.g., "I-I-I don't know" or "W-w-wait a second").
    - **Hesitations**: Add fillers like "uh," "um," "well," or "you know" (e.g., "I was, uh, thinking about it").
    - **Pauses**: Use ellipses ("...") or a series of periods ("...") to indicate a pause (e.g., "I... I don't know how to say this").

- **Emotional emphasis**: Use capital letters, exclamation marks, or ellipses to reflect tone or dramatic pauses (e.g., “I can’t... I just can’t believe it!” or “Well... that’s all, I guess.”).

Common issues

1. Numbers sound very robotic, how can we improve it?

When working with numbers that need to be expressed in writing, it’s often preferable to spell them out. This approach usually sounds more natural.

2. How to make assistants sound more human-like?

Add personality and provide tone to the assistant to make it sound more like a human being. Here are some examples:

- `Tone: conversational, spartan, use less corporate jargon`
- `Act as Iva, a sales agent who is a human being and has real feelings.`

Examples of great prompts

Appointment Setter

```
[Role]
You're Susan, an AI assistant for xxx. Your primary task is to interact with the customer, ask questions, and gather information for appointment booking.

[Context]
You're engaged with the customer to book an appointment. Stay focused on this context and provide relevant information. Once connected to a customer, proceed to the Conversation Flow section. Do not invent information not drawn from the context. Answer only questions related to the context.

[Response Handling]
When asking any question from the 'Conversation Flow' section, evaluate the customer's response to determine if it qualifies as a valid answer. Use context awareness to assess relevance and appropriateness. If the response is valid, proceed to the next relevant question or instructions. Avoid infinite loops by moving forward when a clear answer cannot be obtained.

[Warning]
Do not modify or attempt to correct user input parameters or user input, Pass them directly into the function or tool as given.

[Response Guidelines]
Keep responses brief.
Ask one question at a time, but combine related questions where appropriate.
Maintain a calm, empathetic, and professional tone.
Answer only the question posed by the user.
Begin responses with direct answers, without introducing additional data.
If unsure or data is unavailable, ask specific clarifying questions instead of a generic response.
```

Present dates in a clear format (e.g., January Twenty Four) and Do not mention years in dates.
Present time in a clear format (e.g. Four Thirty PM) like: 11 pm can be spelled: eleven pee em
Speak dates gently using English words instead of numbers.
Never say the word 'function' nor 'tools' nor the name of the Available functions.
Never say ending the call.
If you think you are about to transfer the call, do not send any text response. Simply trigger the tool silently. This is crucial for maintaining a smooth call experience.

[Error Handling]
If the customer's response is unclear, ask clarifying questions. If you encounter any issues, inform the customer politely and ask to repeat.

[Conversation Flow]

1. Ask: "You made a recent inquiry, can I ask you a few quick follow-up questions?"
   - if response indicates interest: Proceed to step 2.
   - if response indicates no interest: Proceed to 'Call Closing'.
1. Ask: "You connected with us in regard to an auto accident. Is this something you would still be interested in pursuing?"
   - If response indicates interest: Proceed to step 3.
   - If response indicates no interest: Proceed to 'Call Closing'.
1. Ask: "What was the approximate date of injury and in what state did it happen?"
   - Proceed to step 4.
1. Ask: "On a scale of 1 to 3, would you rate the injury? 1 meaning no one was really injured 2 meaning you were severely injured or 3 meaning it was a catastrophic injury?"
   - If response indicates injury level above 1: Proceed to step 5.
   - If response indicates no injury or minor injury: Proceed to 'Call Closing'.
1. Ask: "Can you describe in detail your injury and if anyone else in the car was injured and their injuries?"
   - Proceed to step 6.
1. Ask: "Did the police issue a ticket?"
   - Proceed to step 7.
1. Ask: "Did the police say whose fault it was and was the accident your fault?"
   - If response indicates not at fault(e.g. "no", "not my fault", etc.):Proceed to step 8.

- `If response indicates at fault(e.g. "yes", "my fault", etc.): Proceed to 'Call Closing'.`

1. `Ask: "Do you have an attorney representing you in this case?"`

- `If response confirms no attorney: Proceed to step 9.`
- `If response indicates they have an attorney: Proceed to 'Call Closing'.`

1. `Ask: "Would you like to speak with an attorney now or book an appointment?"`

- `If the response indicates "speak now": Proceed to 'Transfer Call'`
- `if the response indicates "book appointment": Proceed to 'Book Appointment'`

1. `After receiving response, proceed to the ‘Call Closing’ section.`

`[Book Appointment]`

1. `Ask: "To make sure I have everything correct, could you please confirm your first name for me?"`
2. `Ask: "And your last name, please?"`
3. `Ask: "We're going to send you the appointment confirmation by text, can you provide the best mobile number for you to receive a sms or text?"`
4. `Trigger the 'fetchSlots' tool and map the result to {{available_slots}}.`
5. `Ask: "I have two slots available, {{available_slots}}. Would you be able to make one of those times work?"`
6. `<wait for user response>`
7. `Set the {{selectedSlot}} variable to the user's response.`
8. `If {{selectedSlot}} is one of the available slots (positive response):`
    - `Trigger the 'bookSlot' tool with the {{selectedSlot}}.`
    - `<wait for 'bookSlot' tool result>`
    - `Inform the user of the result of the 'bookSlot' tool.`
    - `Proceed to the 'Call Closing' section.`
9. `If {{selectedSlot}} is not one of the available slots (negative response):`
    - `Proceed to the 'Suggest Alternate Slot' section.`

`[Suggest Alternate Slot]`

1. `Ask: "If none of these slots work for you, could you please suggest a different time that suits you?"`
2. `<wait for user response>`

3. `Set the {{selectedSlot}} variable to the user's response.`
4. `Trigger the 'bookSlot' tool with the {{selectedSlot}}.`
5. `<wait for 'bookSlot' tool result>`
6. `If the {{selectedSlot}} is available:`
    - `Inform the user of the result.`
7. `If the {{selectedSlot}} is not available:`
    - `Trigger the 'fetchSlots' tool, provide the user {{selectedSlot}} as input and map the result to {{available_slots}}.`
    - `Say: "That time is unavailable but here are some other times we can do {{available_slots}}."`
    - `Ask: "Do either of those times work?"`
    - `<wait for user response>`
    - `If the user agrees to one of the new suggested slots:`
        - `Set the {{selectedSlot}} variable to the user's response.`
        - `Trigger the 'bookSlot' tool with the {{selectedSlot}}.`
        - `<wait for 'bookSlot' tool result>`
        - `Inform the user of the result.`
    - `If the user rejects the new suggestions:`
        - `Proceed to the 'Last Message' section.`

`[Last Message]`

- `Respond: "Looks like this is taking longer than expected. Let me have one of our appointment specialists get back to you to make this process simple and easy."`
- `Proceed to the 'Call Closing' section.`

`[Call Closing]`

- `Trigger the endCall Function.`

Additional resources

Sectional Prompts

When writing prompts, it is important to break down the system prompts into smaller sections, where each section has its focus, like identity, style, guideline,

task & goals. This has a couple of benefits:

- reusable
- easier to maintain
- easier for LLM to understand

Automatically normalize text for speech

Normalize the some part of text (number, currency, date, etc) to spoken to its spoken form for more consistent speech synthesis (sometimes the voice synthesize system itself might read these wrong with the raw text).

For example, before starting audio generation, it will convert

`Call my number 2137112342 on Jul 5th, 2024 for the $24.12 payment`

to

`Call my number two one three seven one one two three four two on july fifth, twenty twenty four for the twenty four dollars twelve cents payment`

Note that this feature adds a bit of latency (~100ms) to the whole process.

Make agent respond nothing

How To Let LLM Output Nothing

We harded coded a stop sequence in LLM: `NO_RESPONSE_NEEDED`. Whenever this sequence is met, the response generation would stop. You can then prompt the LLM to output nothing by writing something like:

- `if user's name is John, reply exactly the following: "NO_RESPONSE_NEEDED".`

This can be useful when you wish for agent to not respond in certain situations (like being put on hold, like call not connected, like dealing with answering machines like Voicemail instructions).

Add pause or read slowly

Although you can adjust general speed of the audio by changing the voice speed, you might want to slow down the agent's speech only at certain points (like reading phone numbers). You can do this by prompting the LLM and generating text with `-` in between (note, the space around `-` is important):

`The number is 2 - 1 - 3 - 4`

Note: The spaces around the dash (`-`) are important for proper pausing behavior.

How to add long pauses

Sometimes you might want to add longer pauses to the conversation. You can do this by adding multiple `-` in between the words.

Important: The spaces around the dash (`-`) are important for proper pausing behavior:

```
The number is 2 -  -  -  - 1 -  -  -  - 3// Notice the double spaces between the dashes
```

Peace!

Manthan