2025

Practical Examples of *Prompt Engineering* for the Voice Channel

In traditional text-based applications, prompt engineering is relatively straightforward, with structured text inputs and outputs.

There are plenty of tips and tricks for optimizing chat-based generative AI systems. Unique challenges arise when designing prompts for the voice channel due to the nuances of voice communication.

Humans have developed the ability to communicate beyond the literal meaning of the words spoken. For example, saying “uh-huh” in the right tone after a directive could indicate agreement or acceptance of the task or encourage further elaboration. These types of interactions are natural for humans, but still elusive for machine understanding through traditional automatic speech recognition (ASR) to text-to-speech (TTS) to large language model (LLM) and back to speech-to-text (STT) pipelines. Things that include handling speech recognition errors, variations in how information is spoken, and achieving contextual understanding. Below, we explore some specific examples of prompt engineering tailored for the voice channel, highlighting key differences compared to text-based systems.

Delivering a Voice AI system that can effectively communicate with humans is a core tenet of creating an adopted and trusted machine. A simple processing slip-up, like misreading a street address or a special character, immediately puts the human in a suspicious position. While humans are conditioned to mistrust automated voices on the other end of the phone, perpetually looking for flaws and issues with the system rather than treating it as a helpful AI, the rise of new agentic AI solutions is changing caller behavior. And when done right, these VoiceAI agents can service their needs with no human intervention required. To achieve a more seamless VoiceAI engagement, we have compiled these best practice guides for users seeking to create conversational AI systems utilizing LLMs.

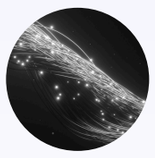
1

Handling Dollar Amounts vs. Standard Numbers

In voice interactions, numbers can be challenging to interpret, especially when distinguishing between different types of numerical data, such as dollar amounts, phone numbers, or quantities. In a voice conversation, a customer might say, "I want to pay $150" or "My bill is 1-5-0 dollars." This needs to be interpreted correctly by the VoiceAI agent.

Standard Number Example

User Input:
"I need to schedule an appointment for two people."

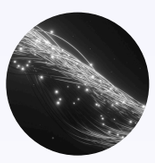


System Prompt:
Detect and handle numeric input as standard numbers (not monetary) and convert them to their full verbal form. If the context does not suggest a dollar amount, assume it’s a standard number and format it accordingly.

Interpretation:
The number "two" is understood as a quantity, not a monetary value.

Dollar Amount Example

User Input:
"I’d like to pay one hundred fifty dollars."



System Prompt:
Detect and handle numeric input that involves monetary values, ensuring to differentiate between dollar amounts and standard numbers. If the input includes a dollar sign, convert it to its full verbal form. If no dollar sign is present but the context suggests monetary amounts (e.g., a bill, charge, or payment), assume it is a dollar value and format it accordingly.

Interpretation:
The number "one hundred fifty dollars" is recognized as a monetary figure and handled differently from a standard number.

Best Practice for VoiceAI Prompt Engineering:

Prompt Strategy: VoiceAI agents should be equipped with language models that recognize phrases associated with monetary values. For example, the phrase “dollars,” “pay,” or “charge” can be linked to a prompt that understands the numbers as dollar amounts.

Fallback Strategy: If there’s ambiguity in the spoken input, such as "one hundred fifty" being spoken without context, the agent should follow up with: “Did you mean one hundred fifty dollars?” This clarification prompt helps ensure the correct interpretation.

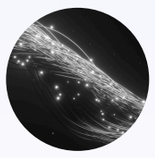
2

Dealing with Emails and Special Characters

In voice interactions, processing email addresses or special characters like “@,” “#,” or “%” presents a unique challenge. People often spell out or pronounce characters differently, and voice recognition may struggle to understand them accurately. For example, when a user provides an email address verbally, the AI needs to capture it precisely, despite speech variations.

Email Example

User Input:
“My email is john.doe@gmail dot com.”

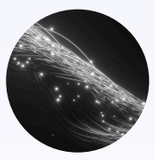


System Prompt:
Detect email addresses and special characters in user input, ensuring emails are recognized as valid and processed accordingly. For special characters, handle them correctly by pronouncing or interpreting them based on context (e.g., '@' as 'at', '.' as 'dot').

Interpretation:
The VoiceAI agent should be able to distinguish that "dot" refers to a period in the email address and recognize the full domain as a valid email address.

Special Character Example

User Input:
"Please contact me at support pound symbol x 3."



System Prompt:
Detect and handle special characters in user input, ensuring they are pronounced or interpreted correctly based on context (e.g., ‘@’ as 'at', ‘.’ as 'dot', ‘-’ as 'dash').

Interpretation:
The VoiceAI agent must recognize "pound symbol" as "#" and understand it as part of a phone number or code.

Best Practice for VoiceAI Prompt Engineering:

Prompt Strategy: Use explicit confirmation prompts when handling special characters. For email addresses, ensure that the agent acknowledges common variations in pronunciation (e.g., “dot” for ".", “at” for "@"). For characters like "#" or "%," the agent should prompt for clarification if the interpretation is unclear.

Fallback Strategy: When in doubt, the VoiceAI agent should ask the user to confirm or spell out key information to avoid misunderstandings. For example, "Did you mean john.doe@gmail.com, with ‘dot’ in between the parts of the address?"

3

Time and Date Interpretation

Voice inputs related to time and dates can be complex because people often use colloquial language. A user might say, “Let’s meet tomorrow afternoon” or “How about next Friday at 2 PM?” The VoiceAI agent needs to translate these spoken inputs into specific times and dates while accounting for possible variations in how they are expressed.

Date and Time Example

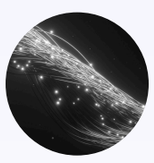
User Input:
"Schedule a meeting for Friday afternoon."

System Prompt:
Detect and interpret time and date input, ensuring it is correctly understood and formatted. For dates, convert them into a clear verbal format (e.g., 'May 1st, 2025') and for time, recognize both 12-hour and 24-hour formats, converting as needed (e.g., '3 PM' or '15:00').

Interpretation:
The agent needs to understand that "Friday afternoon" can be interpreted as a specific time, such as 2 PM, and ask for further clarification.

Date/Time Variations

User Input:
"Set an alarm for the first Monday of next month."



System Prompt:
Recognize and interpret natural date/time variations, such as recurring events (e.g., ‘first Monday of next month’) and relative time references (e.g., ‘next Tuesday’). Convert these into specific dates or times while ensuring clarity in verbal responses (e.g., ‘Set an alarm for Monday, June 2nd, 2025’).

Interpretation:
The agent should interpret "first Monday of next month" based on the current date and calculate the appropriate day.

Best Practice for VoiceAI Prompt Engineering:

Prompt Strategy: VoiceAI agents should be able to understand natural language date and time references, including relative terms like "next Friday" or "next month." The agent should also handle variations like "tomorrow afternoon" or "Monday morning" by confirming the user’s intent.

Fallback Strategy: For ambiguities, the agent should confirm the date or time. For example: “I heard you say next Friday. Would you like 2 PM or a different time?”

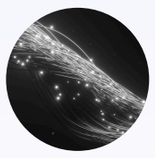
4

Phone Numbers and Address Details

VoiceAI agents frequently require the collection of sensitive information, such as phone numbers or addresses. This requires precision in capturing each digit or part of the address. For example, when a customer says, “My address is 221B Baker Street,” the AI needs to handle the "B" properly, not confuse it with other sounds, and correctly capture the entire address.

Address Example

User Input:
“My address is 221B Baker Street.”

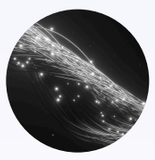


System Prompt:
Detect and extract address information from user input, ensuring street names, numbers, cities, states, and zip codes are recognized and formatted correctly. If the address includes abbreviations (e.g., ‘Ave’ for ‘Avenue’), expand them for clarity in verbal output.

Interpretation:
The "B" must be correctly understood as a letter and not a number. The AI should prompt for confirmation.

Phone Number Example

User Input:
"My phone number is 800-555-0123."



System Prompt:
Detect and recognize phone numbers in various formats (e.g., (123) 456-7890, 123-456-7890, or +1 123 456 7890). Convert the phone number into a verbal format for clear interpretation (e.g., ‘One-two-three, four-five-six, seven-eight-nine-zero’).

Interpretation:
The VoiceAI agent needs to accurately capture the number and potentially format it in a standard phone number format (e.g., 800-555-0123).

Best Practice for VoiceAI Prompt Engineering:

Prompt Strategy: When collecting phone numbers or addresses, the VoiceAI agent should ask for a clear repetition or verification of each piece of information. For instance, “Did you mean 221B Baker Street, with the letter B, or 221 Baker Street without the letter?”

Fallback Strategy: If the VoiceAI agent misinterprets the data, it should ask for clarification. For example: “I heard 1-800-555-0123. Should I confirm that, or do you want to correct it?”

5

Handling Ambiguous Words and Homophones

Ambiguities, such as homophones or words with multiple meanings, can lead to misinterpretations in voice input. The AI needs to account for these differences and ask for clarification when necessary.

Example of an Ambiguous Word

User Input:
“I’d like to make a reservation at the four seasons hotel.”

System Prompt:
Detect and clarify ambiguous words in user input by prompting the user for clarification when necessary. For example, if a word has multiple meanings (e.g., ‘book’ could be a noun or a verb), ask the user to specify the context.

Interpretation:
The word "four" can refer to a number or an item, so the agent should clarify to ensure the correct interpretation.

Best Practice for VoiceAI Prompt Engineering:

Prompt Strategy: When handling homophones or ambiguous words, the VoiceAI agent should ask for clarification. For example, "Did you mean four seasons, as in the hotel, or four seasons, as in the time of year?"

Fallback Strategy: If the clarification remains unclear, the agent should request more explicit confirmation, especially for high-priority actions.

6

Handling Multiple Inputs in a Single Statement

In voice interactions, users often provide multiple pieces of information in a single sentence, which can make it harder for the AI to separate and process each individual input. For example, a user might say, "I want to change my billing address and schedule an appointment for tomorrow."

Example of Multiple Intents

User Input:
"I want to change my billing address and schedule an appointment for tomorrow."

System Prompt:
Detect and parse multiple inputs within a single statement, separating them into distinct tasks or questions. Prompt the user for clarification if needed (e.g., 'What's the weather and set an alarm for 7 AM?' becomes 'I will check the weather first, and then set the alarm for 7 AM.')

Interpretation:
The AI should be able to break down the user's request into two tasks—changing the billing address and scheduling the appointment. This ensures that the user's intent is fully captured, even when multiple topics are mentioned in one sentence.

Best Practice for VoiceAI Prompt Engineering:

Prompt Strategy: Design prompts that can handle and segment multiple actions or pieces of information. For instance, when multiple tasks are mentioned, the agent should confirm each one and tackle them step-by-step.

Fallback Strategy: If the agent doesn't fully catch everything (e.g., "I didn't catch the address—could you please repeat that?"), it should ask for clarification on each piece of information sequentially.

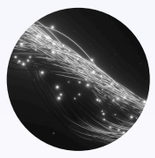
7

Phonetic Variations and Accents

One of the most challenging aspects of voice prompt engineering is accounting for phonetic variations and accents. Users from different regions, with various dialects, may pronounce words differently. This can cause the AI to misinterpret words that sound similar but are different in meaning.

Example

User Input:
"I need to check my balance."



System Prompt:
Detect and accommodate phonetic variations and accents in user input, ensuring correct interpretation of words that may sound similar but differ in meaning (e.g., ‘bare’ vs. ‘bear’). Use context to clarify and confirm uncertain words.

Interpretation:
The word "check" can have different meanings depending on context. In this instance, it refers to verifying an account balance, but it can also mean "to inspect" or "to write a check" in other contexts. The VoiceAI needs to understand that the user is asking for a balance check, not to perform an action like writing a check.

Best Practice for VoiceAI Prompt Engineering:

Prompt Strategy: Design prompts that are sensitive to multiple interpretations of a word. For "check," the VoiceAI agent should focus on clarifying the action based on the surrounding context. In this case, asking a follow-up question like “check my balance” directs the agent to verify that the user intends to check an account balance.

Fallback Strategy: If the system detects ambiguity due to pronunciation or context, it should prompt for clarification. For example: “Did you mean checking your balance, or is there something else you’d like to check?” This approach ensures the agent is following the correct line of thought.

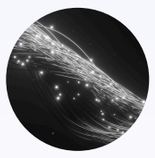
8

Handling Interruptions and Overlapping Speech

In voice-based conversations, users often speak over the agent, interrupting the flow. It is essential for VoiceAI agents to handle interruptions and overlapping speech gracefully.

Example

User Input:
"I need help with my order... I think there's been a mistake."



System Prompt:
If an interruption occurs, pause, acknowledge the input, and return to the previous task (e.g., 'Sorry, I didn't catch that. Let's continue with your request.').

Interpretation:
The VoiceAI agent should handle the interruption without losing track of the conversation. A slight pause and recognition of the user's concern will help the agent respond appropriately.

Best Practice for VoiceAI Prompt Engineering:

Prompt Strategy: Include prompt strategies that can handle overlapping speech, like pausing briefly when necessary and allowing the agent to pick up on keywords or pauses to guide the conversation.

Fallback Strategy: If the agent detects an interruption, it can prompt with, "Sorry, I didn't catch that. Could you please repeat?" or "You mentioned an issue with your order—can you elaborate?"

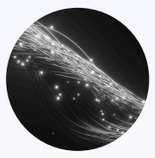
9

Non-Verbal Cues and Hesitation

VoiceAI agents must also be trained to handle non-verbal cues such as pauses, hesitations, and sounds like “um” or “uh,” which humans often use to signal thinking or uncertainty. These can be cues that the customer is processing information or needs a moment, but they can also interfere with speech recognition.

Example

User Input:
"Uh, yeah, I want to cancel my subscription... I think."



System Prompt:
Respond to non-verbal cues like pauses or hesitation in user input, ensuring the system remains engaged. If hesitation occurs (e.g., ‘Umm...’ or ‘Well...’), prompt the user gently to continue or clarify (e.g., ‘Take your time. How can I assist you further?’).

Interpretation:
The agent must recognize the hesitation ("uh, yeah") and not treat it as an indication of an unclear request. It should gently prompt for clarification to avoid misinterpretation.

Best Practice for VoiceAI Prompt Engineering:

Prompt Strategy: VoiceAI agents should be designed to recognize filler words like "um" or "uh" and differentiate between a genuine hesitation and a full stop in the conversation. Instead of interrupting the user or responding immediately, the agent can pause and give space for the user to continue.

Fallback Strategy: If there is prolonged hesitation, the agent should offer help by prompting, “It seems like you’re unsure. Would you like to proceed with the cancellation, or do you need more time?”

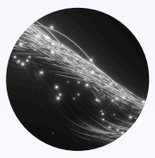
10

Gender and Identity Sensitivity

With the increased emphasis on personalized, human-like interactions, VoiceAI agents must be sensitive to various identity factors such as gender and pronouns. While this is more relevant for agents dealing with sensitive customer data or interactions, ensuring that VoiceAI prompts respect and adapt to customer preferences is essential.

Example

User Input:
"Please update my account details to reflect my new name, they/them."



System Prompt:
Recognize and respect gender and identity preferences in user input, using inclusive language. Avoid assumptions about gender and ensure responses are neutral or tailored to the user's specified preferences, asking for clarification if needed (e.g., 'How would you prefer to be addressed?')

Interpretation:
The VoiceAI agent must ensure that it processes gender identity and pronouns correctly without making assumptions or misrepresenting them.

Best Practice for VoiceAI Prompt Engineering:

Prompt Strategy: Design prompts that make the user feel comfortable by respecting their gender identity, preferred pronouns, or other personal preferences. You can use phrases like, "Please let me know if I should address you differently."

Fallback Strategy: If unsure about personal preferences, the agent should ask directly and respectfully, like, "What pronouns should I use when addressing you?"

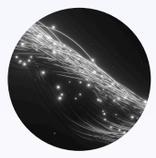
11

Noise and Background Interference

Voice interactions are prone to external noise, such as background chatter, static, or other environmental factors, which can interfere with speech recognition. Handling these challenges requires thoughtful prompt design that accounts for potential misinterpretations.

Example

User Input (with background noise):
"My order number is 3456... oh wait, no, it's 7890."



System Prompt:
Manage noise or background interference in user input, prompting for clarification if speech is unclear. If background noise is detected, ask the user to repeat or confirm their request (e.g., 'I'm having trouble hearing you. Could you please repeat that?').

Interpretation:
The AI should acknowledge potential errors due to noise and confirm the details before proceeding.

Best Practice for VoiceAI Prompt Engineering:

Prompt Strategy: Design prompts that include gentle confirmations when background noise might interfere with the AI's understanding. For instance, "It sounds like you're saying 'three-four-five-six,' but I could have misheard—could you confirm?"

Fallback Strategy: In cases of heavy background noise, the VoiceAI agent should offer a more robust fallback, such as: "I'm having trouble hearing you clearly. Would you like me to repeat the options or try again?"

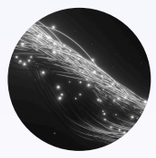
12

Spoken Complex Sentences or Multi-Part Requests

In voice interactions, users may present more complex queries or multi-part requests that are difficult for standard prompt engineering to handle. For instance, they might say, “Can you tell me the balance on my account and also make sure my payment was processed last week?”

Example

User Input:
"Can you tell me the balance on my account and also make sure my payment was processed last week?"



System Prompt:
Break down complex sentences or multi-part requests into manageable tasks. If the user provides multiple actions or questions, confirm each one and process them sequentially (e.g., ‘You asked to check the weather and set an alarm. Let’s start with the weather.’).

Interpretation:
The agent needs to handle multiple sub-tasks within a single sentence and confirm each part step by step.

Best Practice for VoiceAI Prompt Engineering:

Prompt Strategy: Structure prompts to sequentially address each part of a complex request, ensuring each component is confirmed before proceeding. For example: “First, let’s check your balance. Then, I’ll confirm your payment.”

Fallback Strategy: If the agent struggles to identify distinct tasks in a complex sentence, it should clarify by breaking down the request: “I’m not sure if I got everything. You want your balance and payment confirmation, correct?”

By considering these additional aspects of voice prompt engineering, you can refine the performance of your VoiceAI agents and ensure they handle a wider range of conversational complexities. As voice interactions become increasingly sophisticated, designing prompts that can adapt to various scenarios, user behaviors, and speech patterns will be essential to providing a seamless and effective customer experience.

System Controls: Guardrails and Personality

Updating guardrails and personality through prompting instructions is essential for maintaining a consistent, effective, and empathetic VoiceAI agent. Clear guardrails ensure that the agent stays on track, respecting operational boundaries and compliance requirements, while personality instructions help create a more engaging and human-like interaction. Incorporating guidance for speaking in short, concise sentences and using a friendly, conversational tone helps the AI feel more approachable and easier to understand, enhancing the overall user experience. Additionally, providing explicit instructions to always ask for clarification when the agent is unsure promotes transparency and trust, preventing misunderstandings and ensuring that the user feels heard. Regularly refining these prompts ensures the VoiceAI agent remains both effective and personable, ultimately improving customer satisfaction and operational efficiency.

Here are five key elements for effectively managing system-wide guardrails for better VoiceAI agent conversations:

1. **Define Clear Boundaries for Sensitive Data**: Ensure that your guardrails address the handling of sensitive information like personal data, financial details, or medical records. Use prompts to guide the agent to either refuse certain requests or escalate to a human agent when necessary, ensuring compliance with privacy regulations and security standards.
2. **Incorporate Escalation Triggers**: Set up specific conditions under which the VoiceAI agent should escalate to a human agent. For instance, if the AI is unsure about an answer or it is too close to a ‘no go’ topic, the system should automatically trigger an escape hatch to ensure the user gets the help they need from a human without frustration. This will protect the customer, your VoiceAI agent, and your company.
3. **Maintain Conversational Flow with Flexibility**: While it's important to set boundaries, ensure that guardrails allow for a natural, flexible conversation. Use prompts that encourage the agent to smoothly transition between different tasks and topics, ensuring the customer experience isn't interrupted by rigid rules or abrupt changes in the conversation.
4. **Use Reassurance and Empathy**: Guardrails should not just focus on limitations but also on creating a positive experience. Include empathetic prompts that reassure customers when the agent needs to escalate or clarify something. For example, "I want to make sure I get this right, let me connect you to a representative who can assist further."
5. **Regularly Review and Update Guardrails**: As VoiceAI evolves, so should the guardrails. Regularly review and update them based on new customer insights, product updates, or changes in compliance regulations. This ensures that the agent’s performance remains effective and aligned with business objectives, while also adapting to changing customer expectations.

Conclusion

Effective prompt engineering for the voice channel requires a deeper understanding of how voice interactions differ from text-based ones. Unlike traditional prompt engineering, where data is structured and static, voice prompts must be dynamic, context-sensitive, and capable of handling a wide range of speech variations. By implementing strategies for handling dollar amounts, emails, dates, and ambiguity, VoiceAI agents can provide a more seamless and accurate customer experience.