A Practical Guide to Voice AI



By Jennifer Marsh
Conversational AI Product Leader

Contents

Introduction 3

What This Guide Covers and Who Would Benefit 3

Part I: Should You Use Voice AI? 4

- Why Businesses Are Looking at Voice AI 4
- Start With the Task, Not the Technology 5
- What Conversational AI Actually Is 5
- Where Voice AI Works Well 6
- Where to Use Caution 7

Part II: Choosing and Designing the System 8

- Types of Voice AI Vendors 8
- What to Evaluate 9
- Design the System Before You Write the Prompts 9
- Give It the Information It Needs 10
- Decide What Happens When It Cannot Help 10
- Think Beyond the Call 11

Part III: Designing the Conversation 11

- How Prompting Fits into the System 11
- Prompt for Behavior, not a Script 12
- The First 30 Seconds 12
- Keep the Conversation Moving 12
- Interruptions, Pauses, and Timing 13
- Confirmation Without Making the Call Painful 13
- Know When to Stop 14

Part IV: Evaluating and Improving Performance 14

- Common Failure Modes 14
- The Agent Makes Something Up 14
- The Conversation Goes Nowhere 15

The Agent Cannot Recover from an Unexpected Answer 15
The Agent Doesn't Transfer Fast Enough or Transfers Too Quickly 15
Evaluate Calls, Not Just Prompts 15
Change One Thing for a Reason 16
Watch for Regressions 16
Use Different People 17
Part V: How to Prompt for Success 17
Define the Role 17
Define the Goal 18
Provide Context 18
Set Behavioral Expectations 18
Define Boundaries 18
Compliance Considerations: When to Seek Legal Guidance 19
Conclusion 20

Introduction

For the past couple of years, I have worked on conversational AI products in the legal, home services, and healthcare industries. During that time, I have seen these systems improve considerably, but I have also seen how quickly a seemingly simple use case can become complicated once you put an AI agent on the phone with real people.

Answering a phone call sounds straightforward. In practice, callers interrupt, change subjects, give incomplete answers, ask unexpected questions, speak unclearly, and do not necessarily follow the path you designed for them. The AI must understand what the caller wants, know what information matters, decide what to do next, and recover when the conversation does not go as planned. How well it does those things depends heavily on how the system is designed, what information it has, and what you have told it to do.

This guide is based on what I have learned from working with these systems in practice. My goal is to help businesses understand where conversational AI can be useful, where it can go wrong, and what to think about before putting an AI agent in front of their own callers.

What This Guide Covers and Who Would Benefit

This is not a technical manual, and it is not a vendor pitch. I am not sharing anything proprietary or specific to a particular system. Instead, I am drawing on my experience working with conversational AI and evaluating how these systems behave in practice.

This guide is for businesses considering conversational AI to answer phones or handle some portion of inbound calls, as well as those already using these systems who want to improve performance. It may also be useful to product managers, conversation designers, implementation teams, and consultants responsible for designing, configuring, or evaluating voice AI solutions. While these groups approach conversational AI from different perspectives, they face many of the same questions: What can the technology reliably accomplish? How should the system be designed? What occurs when something goes wrong? And how do you know whether it is working?

A lot of the discussion around conversational AI focuses on prompts: how to write them, how to make the agent sound more natural, and how to get the model to follow instructions. Those things matter. But in my experience, most problems start earlier: with what you are asking the system to do.

If the use case is poorly defined, the system lacks the information it needs, or the workflow requires judgment the AI is not equipped to exercise, better prompts will only get you so far.

Whether you are selecting a vendor, implementing a system for a client, or building a conversational AI product, understanding those limitations is important.

That is why this guide starts with the use case rather than the prompt.

Part I: Should You Use Voice AI?

Why Businesses Are Looking at Voice AI

Phones still matter. Many customers, patients, and clients prefer calling when they have a question, need help, or want to buy something. Calls may come in at inconvenient times, and unexpected circumstances can create volume spikes.

If nobody answers, the caller may leave a message. More often, especially when they are looking for a new product or service, they may simply move on.

The alternatives are not always particularly good. Hiring enough people to provide consistent phone coverage is expensive. Voicemail is inexpensive but cannot answer questions or book a consultation. Interactive voice response systems can route calls, but callers generally do not enjoy navigating menus. External call centers provide human coverage, but the person answering may know relatively little about your business.

That creates an obvious opportunity for conversational AI. An AI agent can answer immediately, handle multiple calls, provide information it has been given, collect information from callers, and potentially perform tasks such as scheduling or routing. It does not need breaks and can provide coverage outside normal business hours.

But the capabilities also make it easy to expect too much. People sometimes approach these systems as if they can replace a knowledgeable employee who understands the business and can figure out what to do in almost any situation. That is still a very different proposition from automating a defined set of tasks.

If you can afford to have a knowledgeable person answer every call and help every caller, that is generally a very good caller experience. Most businesses cannot do that. The more useful comparison is often between conversational AI and the realistic alternatives: voicemail, an IVR, unanswered calls, or a call center representative with limited information.

In that context, Voice AI can be quite useful. The next question is where.

Start With the Task, Not the Technology

When businesses consider Voice AI, they often ask, "Can we replace a person with AI?" I think that is the wrong place to start. The better question is: **What does the caller need to accomplish, and can an AI system reliably help them do it?**

Once callers get beyond the initial discomfort some people have with speaking to the AI, they tend to care less about who answered and more about whether they can get what they need without needless friction. And a human being isn't automatically helpful just because they are human.

One call that stuck with me was to a dermatologist's office. My insurance had changed, and I wanted to know whether the practice accepted my new plan. I could not find the answer on the website, so I called. A live call center representative answered. She was perfectly polite, but she could not answer my question. She also could not transfer me to someone in the office who could. Her only option was to take a message and have someone call me back.

The frustrating part was not that I had spoken to a call center. It was that the entire interaction took several minutes before I realized the person answering the phone couldn't actually help me. That task did not necessarily require a person. It required accurate information about accepted insurance plans and the ability to communicate that information to me.

I have had similar experiences with hotels. A call center representative may be able to book a room but be unable to answer a relatively simple question about the physical layout. If that information matters to my decision, the fact that someone answered doesn't solve the problem.

In both cases, the real issue is capability. The system answering the phone, whether it consists of people, automation, or some combination of the two, needs to be capable of helping with the reason people are calling. That is a much more useful way to think about Voice AI: can it help the caller as well as or better than a human could? For well-defined tasks, when the AI has access to the right information and capabilities, the answer can absolutely be yes.

What Conversational AI Actually Is

A conversational AI agent is an automated system that lets a caller speak naturally rather than follow a fixed phone menu or script. Traditional IVR systems require the caller to adapt to the system: press one for sales, press two for support, and so on. A conversational system attempts to interpret what the caller is saying and reply based on the conversation.

It is also different from a script. Imagine that you wanted someone to interview customers about a new product. You could give the interviewer a list of questions and require them to ask those questions in the same order every time. Or you could tell the interviewer what you are trying to learn, give them the necessary background, and let them have a conversation that naturally gets to the relevant information. Conversational AI is closer to the second model.

The system has a goal and context, and its responses can change depending on what the caller says. It can be given a list of questions to ask, but it can also be provided with the parameters and context to have a natural conversation with the caller. That is one reason these systems can feel considerably more natural than older forms of phone automation. That flexibility also makes them harder to design. Some systems combine this conversational flexibility with more structured workflows that control which steps must be completed or which actions the agent can take. This allows the AI to respond naturally to callers while still following defined business processes.

The flexibility is not unlimited. The system is still operating within the information, goals, instructions, and capabilities it has been given. It doesn't understand your business the way an experienced employee does. I sometimes think of it as closer to a new employee than an experienced one.

A new employee understands the general job but may not know what to do when something unusual happens. They can handle the common path. When the conversation moves outside of that path, they may need extra guidance or help.

However, there is an important difference. A person can often recognize when they don't know something. AI systems may continue trying to answer even when they lack the necessary information. That is where incorrect or invented information can creep in.

A well-designed agent therefore needs more than information about what it should do. It also needs guidance about what it should **not** do and what happens when it reaches the limits of its role. If the bot is built well and monitored, especially at the start, these risks can be minimized.

Where Voice AI Works Well

The strongest Voice AI use cases share a few things in common: the task is reasonably well defined, the required information is available, and there is a recognizable successful outcome. Some tasks well-suited to conversational AI are:

- Answering common questions is an obvious example. If callers regularly ask about hours, locations, services, accepted insurance, basic pricing information, or how to

schedule an appointment, an AI agent can answer those questions if it has reliable information.

- Information collection is another strong use case. Taking messages, gathering contact information, collecting basic intake details, or asking a defined set of qualifying questions are all relatively structured tasks.
- Scheduling can also work well when the underlying system supports it. The objective is clear, and an external source of truth defines available appointments.
- Routing and transferring can be useful for the same reason. If the agent can determine what the caller needs and clear rules define where different calls should go, conversational routing can be less frustrating than forcing the caller through a phone tree.

These uses of AI are not necessarily glamorous. That is partly why they work for you. If your AI phone system can handle a lot of low-hanging fruit, it frees you or your staff to handle the harder questions.

Where to Use Caution

Problems tend to appear when a task requires information the system doesn't have, judgment it is unable to reliably exercise, or an outcome that has never been clearly defined. Suppose someone asks a question that is not covered by the information available to the agent. Depending on the system and its instructions, it may not simply say, "I don't know." Instead, it may attempt to construct an answer that sounds plausible but is untrue.

That is obviously dangerous in some settings and merely annoying in others, but neither is desirable. This risk can be mitigated, however, by giving the AI agent extensive context and instructing it to tell the caller it doesn’t know the answer if the answer isn't in its context. In my experience, AI agents can follow these types of instructions well, including avoiding legal or medical advice. However, these guardrails are not foolproof, and their effectiveness needs to be tested and monitored.

Ambiguous goals create a different problem. If the system has been told broadly to "help the caller," there may be no obvious point at which it has finished helping. The conversation can continue without moving toward an outcome.

Complex judgment is another limitation. An important difference exists between collecting information and deciding what it means. There is also a difference between providing a known policy and determining whether an unusual situation qualifies for an exception. The deeper you move into nuance, exceptions, or discretionary decision-making, the more

carefully you should consider whether the AI should be handling that portion of the interaction at all.

Callers themselves create another source of complexity. Real people do not behave like test scripts. They may speak softly and unclearly. They change their minds. They answer questions indirectly. They interrupt. They tell long stories. They provide three pieces of information when you asked for one. They ask a question in the middle of answering yours. Sometimes they misunderstand what you asked.

That is normal caller behavior. A useful system must tolerate a reasonable amount of this caller behavior without losing track of what it is trying to accomplish. It must know when to redirect the caller back to the goal and when to clarify the caller's answer. It may also need to filter background noise or enhance the voice.

This is also why expanding an agent's scope is not automatically an improvement. A system that does three things reliably may provide a better caller experience than one that nominally does fifteen things but handles several of them poorly. Start narrower than you think you need to. Expand based on what the system shows it can handle reliably.

Part II: Choosing and Designing the System

Types of Voice AI Vendors

Once you have a use case, you can start thinking about the system itself. Voice AI products vary considerably. At a high level, one useful distinction is between managed solutions and platforms that give the customer more direct control.

With a managed solution, the vendor may design and maintain much of the experience. You explain what you need, provide information about your business, and the vendor handles much of the configuration. That can reduce the internal work required to launch.

With a more configurable platform, you may have much greater control over the agent's behavior, prompts, workflows, integrations, and information. That flexibility can be valuable, but it also means you take on more responsibility. Neither model is inherently better. The right choice depends partly on how unusual your use case is, how much control you want, whether you have people internally who can manage the system, and how frequently you expect to make changes. In other words, what matters is understanding what you are buying.

What to Evaluate

Voice quality is the easiest thing to notice in a demo, but it should not be the only thing you evaluate. A beautiful voice attached to a system that is unable to reliably do the job you need isn't particularly useful. So instead, start with the actual use case.

Can the system do the things you identified as necessary? If callers need to be transferred, understand how transfers work. If the agent needs to schedule appointments, understand the appointment scheduling capabilities. If information needs to move into another system after the call, determine whether the required integration exists.

You should also pay close attention to audio and transcription performance.

A conversational agent cannot respond correctly to something it did not hear correctly. Background noise, poor connections, accents, mobile phones, interruptions, and speech recognition all affect the experience. What seems like a reasoning problem could have started as an audio issue.

Then consider what occurs when things go wrong. Can the agent recognize when it cannot help? Can the caller reach a person? What happens if a transfer fails? What happens if an integration is unavailable? What does the system do when it cannot understand the caller?

Those situations are not unusual edge cases. If the system handles enough calls, they will happen. You should also understand how observable the system is after launch. Can you review recordings and transcripts? Can you identify outcomes? Can you see what happened when a call failed? How quickly can behavior be changed? This is important because you should assume the system will need adjustment after it encounters real callers.

Finally, understand what you control and what the vendor controls. If you discover a problem tomorrow, can you change it yourself? Do you submit a request? How long does that take? What parts of the behavior can be modified easily? Those questions are generally more useful than asking how human the demo sounds.

Design the System Before You Write the Prompts

Once you have selected the technology, resist the temptation to start by writing dialogue.

First define the job. "Answer our phones" is not a job description. Should the agent act primarily as a receptionist? Is it collecting intake information? Scheduling appointments? Answering common questions? Routing callers? Taking messages? All the above? It may do several of these things, but each one should be intentional.

Then define what a successful call looks like. For one caller, success may mean answering a question. For another, it may mean booking an appointment. For another, it may mean collecting enough information for someone to follow up. If you cannot describe what successful completion looks like, it will be difficult to tell the system when it has finished its job.

Next, map the major paths a conversation can take. This does not mean scripting every line. Think in terms of outcomes instead. If a caller wants information, what should happen? If a caller needs an appointment, what should happen? If a caller needs something the agent cannot provide, what should happen? If a caller wants a person, what should happen?

The purpose is not to predict every conversation. You cannot. It is to make sure the system has somewhere sensible to go when common situations occur.

Give It the Information It Needs

If the agent is expected to answer questions about your business, it needs reliable information about your business. That sounds obvious, but lack of context is responsible for a surprising number of apparent "AI problems."

Depending on the use case, the agent may need information about hours, locations, services, pricing, service areas, policies, insurance, scheduling, employees, products, or other subjects callers routinely ask about. To make sure the AI agent has what it needs, think about what a new receptionist would need to know to do the same job. If you wouldn't expect a receptionist to know an answer without training, you shouldn't expect the AI to know it either.

It is equally important to keep that information accurate and up to date. Giving an agent outdated information confidently is worse than giving it no information at all.

Decide What Happens When It Cannot Help

Every system needs an exit. Sometimes the appropriate response is to transfer the call. Sometimes it is to take a message. Sometimes it is simply to explain that the agent cannot answer that question. The important part is deciding this intentionally.

An AI agent that continues trying to help after it has reached the limits of what it can do creates one of the worst Voice AI experiences: the caller becomes trapped in a conversation with something that cannot solve the problem but will not get out of the way.

The opposite can also happen. If the system transfers at the first sign of difficulty, there may be little point in having it answer the phone. The proper boundary depends on the use case. What matters is that one exists and is thought through.

Think Beyond the Call

A call can sound successful and still fail operationally. If the agent collects a message, where does the message go? If it qualifies a potential customer, what happens next? If it schedules an appointment, where is that appointment recorded? If it promises follow-up, how is that follow-up triggered? The conversation is only the first part of a bigger workflow.

This matters especially when evaluating products and vendors because a persuasive demonstration often focuses on the interaction itself. In production, what happens after the caller hangs up can be just as important.

Part III: Designing the Conversation

How Prompting Fits into the System

Only after the system has a defined job, definite outcomes, usable information, and boundaries does prompting become particularly useful. Prompting in a conversational system is not simply writing lines for the AI to say.

Instead, provide instructions that help it understand its role, what it is trying to accomplish, what information it can rely on, and how it should behave while getting there.

The first is the role: A receptionist, intake assistant, scheduling assistant, and a customer service agent have different responsibilities. Giving the agent a specific role helps establish the context for interpreting the conversation.

Then there is the goal: What should be accomplished by the end of this interaction? An agent without a clear goal can be perfectly pleasant and still accomplish very little.

The next piece is context: the information the system needs to perform the job and answer the caller’s questions.

Finally, there are behavioral instructions. These govern things such as how much the agent should say at once, what information it should confirm, what to do if it does not understand, how to respond when the caller changes subjects, and when it should stop trying to handle the interaction itself. These pieces all work together. If the behavior is wrong, changing a sentence's wording may not fix it. Identify which part of the system is causing the problem before you try to fix it.

Prompt for Behavior, not a Script

One of the easiest mistakes is writing a conversational AI prompt like a call center script: "If the caller says X, say Y." Sometimes an exact response is appropriate, but trying to anticipate every phrase a caller might use quickly becomes unmanageable because callers express the same intent in countless ways.

A better approach is to tell the system what the caller's intent means and what it should accomplish in response. This keeps the conversation flexible without leaving the agent entirely on its own. This distinction matters because a conversational system shouldn't require the caller to use exactly the right words.

The First 30 Seconds

The beginning of the call does a surprising amount of work. Before the agent has solved anything, the caller is already forming an impression of what they are dealing with and how they should interact with it. The greeting should identify the business and move into the conversation without gratuitous delay.

The business name must sound accurate. If the system mispronounces the company name, the caller immediately knows something is off. Sometimes the correct written spelling does not produce the correct spoken pronunciation, and the system needs additional guidance.

You also need to decide how the agent identifies itself. I generally favor being clear that the caller is interacting with an AI assistant. Aside from any applicable legal requirements, trying to make the caller believe the system is human can create unnecessary distrust once they figure it out. Disclosure does not need to become a speech, however. A short identification can set the expectation and then get out of the way.

The same applies to explaining capabilities. If callers need to know that the agent can schedule an appointment or transfer them to someone, say so if it helps. But avoid turning the greeting into an instruction manual. People called because they want something. Let them get to it.

Keep the Conversation Moving

Natural conversation isn't the same as a verbose conversation. An agent can sound friendly without acknowledging every statement with a paragraph. In fact, excessive acknowledgments can make an interaction seem less natural because they slow down something the caller is trying to accomplish.

Responses generally need to be short enough for a phone conversation. The agent should also avoid asking several unrelated questions at once. People may answer only one, forget part of the question, or respond in a way that makes it unclear which piece they were addressing.

Where possible, collect information conversationally and in manageable pieces. At the same time, the system should not rigidly ask for information the caller has already provided.

For example, if someone begins the call by giving their name and explaining exactly why they are calling, a system that immediately asks, "May I have your name?" feels like it wasn't listening. The goal is not simply to execute the next prompt. It is to keep track of what has already happened in the conversation.

Interruptions, Pauses, and Timing

Voice adds problems that do not exist in a text interface. People interrupt each other. Sometimes the caller starts talking while the agent is still speaking. Sometimes the system mistakes background noise for speech and stops. Sometimes the caller pauses because they are thinking, looking something up, or speaking to someone else in the room.

A technically correct response can still feel terrible if the timing is wrong.

If the agent jumps in too quickly, the caller feels interrupted. If it waits too long, the caller may think the call has disconnected. If the agent and caller repeatedly speak over one another, the conversation becomes frustrating even if every individual response is reasonable.

Some of this behavior is controlled by the underlying technology rather than prompts, which is important when diagnosing a problem. Not every bad conversation is a prompting problem. If the agent repeatedly responds late, fails to hear the caller, or behaves strangely around interruptions, you may need to determine whether you are looking at an instruction problem, a speech recognition problem, an audio problem, or a system timing problem. That distinction can save a great deal of unnecessary prompt editing.

Confirmation Without Making the Call Painful

Some information deserves confirmation. Names, phone numbers, email addresses, appointment times, addresses, and other details used later may need to be checked before the agent moves on. But confirmation can also become tedious, and AI agents can struggle to understand the caller.

Repeating everything the caller says makes the system feel mechanical and unnecessarily lengthens the call. The goal should be to confirm information where an error matters, not to prove that the system heard every utterance.

The same principle applies to spelling. If a name is common and was clearly understood, spelling every letter back may add little value. If the system is uncertain, or the information will be used in an important downstream process, confirmation becomes more useful. The appropriate amount depends on the consequences of getting the information wrong. If it doesn't really matter, don't feel compelled to confirm it.

Know When to Stop

A good Voice AI experience is not one in which the AI handles the longest possible portion of the call. Sometimes success means getting out of the way. If the caller needs something outside the agent's role, continuing to ask questions or repeatedly trying to help can turn an otherwise useful system into a barrier.

This is why escalation should be part of conversation design rather than treated as a failure. The system completed its job if it recognized that the caller needed something else and moved them to the appropriate next step.

Part IV: Evaluating and Improving Performance

Common Failure Modes

Once real people start calling, problems appear quickly. That does not necessarily mean the system is bad. It means real conversations contain far more variation than the scenarios used during setup.

Fortunately, many problems fall into recognizable categories.

The Agent Makes Something Up

You notice that a caller asked a question and the AI agent gave a bad or untrue answer. This can happen when the agent lacks the necessary information, misinterprets what it has, or tries to answer beyond what its context supports. The instinct may be to rewrite the response. That is often treating the symptom rather than the problem.

Instead, ask why the system thought it should answer. Is the information missing? Is it ambiguous? Does the agent need a clearer instruction not to answer outside the information it has? Sometimes the fix is supplementary context. Sometimes it is a stronger boundary.

The Conversation Goes Nowhere

The call continues, everyone is civil, and nothing useful happens. That usually means the goal is too vague, or the system doesn't know how to guide the caller toward an outcome.

Look at what the agent thinks its job is. "Be helpful" is not enough. Instead, understand the use case, define the goal, and design when the agent should end the call.

The Agent Cannot Recover from an Unexpected Answer

The system asks a question expecting one kind of response and receives something completely different. This happens constantly in real conversations. The caller may answer a different question, provide only part of the information, change the subject, or ask a question of their own. If the agent can only function when the caller follows the expected path, the system is too brittle.

The answer is usually not to write hundreds of additional instructions. Instead, give the system better guidance on what information it needs and how to return to the goal when the conversation strays.

The Agent Doesn’t Transfer Fast Enough or Transfers Too Quickly

This is especially frustrating. The caller needs a person, but the AI keeps trying to help.

Clear escalation guidance may assist, but you should also evaluate what is happening in the conversation. If callers repeatedly reach the same point and ask for a human, that may be telling you something about where the useful scope of the automation ends.

Do not assume every transfer is a failure. Instead, investigate why the callers are asking for a person. Do they not believe the AI can help? Perhaps the introduction needs to clarify what tasks the AI can handle.

For example, perhaps callers believe they need to speak to a human to make an appointment and therefore ask for a person immediately after the AI greeting. If the AI doesn’t transfer the call, the caller will get frustrated. If the AI transfers a call when it could have booked the appointment itself, it is less useful. At the start of the call, the AI could tell the caller it can answer questions, book consultations, or transfer them to the billing department. That way, the caller knows to stay with the AI if they need an appointment and ask to be transferred to the billing department if they need to pay a bill.

Evaluate Calls, Not Just Prompts

Testing before launch is important, but the people who designed the system make unusually cooperative callers. They know what the agent is supposed to do. Even when

they deliberately try to create problems, they are still approaching the conversation with knowledge a real caller does not have.

Real callers are much more informative. That is why I think call evaluation needs to continue well after you launch your AI bot.

In addition to listening to calls, it helps to define what you are measuring. A call may sound perfectly natural but fail to accomplish its purpose. Depending on the use case, you might look at successful appointment bookings, completed intake forms, accurate transfers, or calls that required human intervention. The metrics should reflect what the system was supposed to accomplish, not simply how many calls it handled.

Listen to successful calls, mediocre calls, and failed calls. If you only review obvious failures, you may miss smaller patterns that degrade the experience across many more calls. When something feels wrong, identify the point at which the interaction changed.

Then ask why. Was the agent missing information? Did it misunderstand the caller? Was the goal unclear? Did it fail to recognize that it should transfer? Was the response correct but badly timed? Was there a technical issue rather than a conversational one? The more precisely you identify the failure, the less likely you are to make a change that doesn't address the real problem.

Change One Thing for a Reason

Once you find a bad call, you may be tempted to start improving everything around it. That makes it difficult to know what will improve the caller experience. A better approach is to look for trends and create focused changes.

If several calls show the same behavior, form a hypothesis about why it is happening. Make a change intended to address that problem. Then evaluate whether the behavior improves. This sounds obvious, but it is surprisingly easy to skip.

You can make a prompt cleaner, more detailed, or more sophisticated without improving the caller experience. The call is the evidence, so make one change at a time, listen to calls, see if they improve, then move on to the next change.

Watch for Regressions

Changes can have consequences elsewhere. An instruction added to solve one situation may influence how the agent interprets another. Making a transfer rule more aggressive may solve calls where the agent held on too long but cause premature transfers elsewhere. More thorough confirmation may improve data accuracy but make ordinary calls painfully slow.

This is why evaluating only the scenario you just fixed is not enough. After a significant change, check that the first problem improved, but also evaluate whether common interactions still behave as expected.

As the system matures, this becomes increasingly important. Early problems are often obvious. Later improvements involve finer distinctions, and a change that looks better in one call may make the overall system worse.

Use Different People

Another useful form of evaluation is simply putting the system in front of people who did not help build it. The same group of testers tends to develop habits. They know what questions work. They know where the system is going. Their conversations begin to resemble one another.

Someone unfamiliar with the system does not have those assumptions. They will phrase things differently, misunderstand things you thought were obvious, interrupt at awkward times, and ask questions nobody on the project thought to ask.

That is useful information. The goal is not to make the agent capable of handling absolutely anything someone can throw at it. The goal is to understand the range of normal behavior it will encounter and make it reliable within the scope you have chosen.

Part V: How to Prompt for Success

The following examples are intentionally simple. They are not production prompts and are not meant to be copied into a system without adapting them to your specific business, technology, and use case. The point is to show how the concepts in this guide can translate into instructions.

Define the Role

Instead of: "You are a helpful AI assistant."

Try something more specific: "You are a virtual receptionist for a professional services business. You answer general questions using the information provided, collect contact information when follow-up is needed, and help callers reach an appropriate next step."

The exact wording is not particularly important. The specificity is.

Define the Goal

For an appointment-oriented use case: "Your goal is to understand why the caller is contacting the business and, when appropriate, help the caller schedule an available appointment."

For an information-collection use case: "Your goal is to collect the information needed for a staff member to understand the caller's request and follow up."

Again, these are starting points rather than magic phrases.

Provide Context

Context may include information such as:

"The business is open Monday through Friday from 9:00 AM to 5:00 PM. It provides [services]. It serves customers in [locations]. Appointments are available [describe general scheduling rules]."

The appropriate context will depend entirely on what callers need. Do not dump information into the system simply because you have it. Give it information that helps it perform the job.

Set Behavioral Expectations

Examples might include:

- "Keep responses succinct and focused on the caller's request."
- "Ask one question at a time when collecting information."
- "Do not guess when information is not provided."
- "Do not repeatedly ask for information the caller has already given."
- "Confirm important information when an error could affect follow-up or completion of the caller's request."

These instructions are useful because they describe how the agent should behave across many different conversations.

Define Boundaries

For example: "If the caller asks for information that is not available to you, do not invent an answer. Explain that you do not have that information and offer the appropriate next step."

Or:

"If the caller's request is outside your role, do not continue trying to solve it. Explain what you can do and follow the defined escalation process when appropriate."

The important point is not the wording.

You are telling the system what to do when it reaches the edge of its knowledge or authority. That is every bit as important as telling it what to do when everything goes according to plan.

Compliance Considerations: When to Seek Legal Guidance

Introducing an AI voice agent into a business workflow can create legal and compliance questions that should be considered before launch. This is not legal advice, and the requirements vary considerably depending on the business, jurisdiction, technology, and type of information involved.

One obvious issue is call recording. Voice AI platforms may record or transcribe calls, and consent-to-record laws differ by jurisdiction. If calls are being recorded, businesses should understand which laws apply, what disclosures may be required, and how recordings and transcripts are stored. Even if a business operates in only one state, it may receive calls from people located elsewhere, which can complicate determining which recording and consent requirements apply.

AI disclosure is another developing area. Requirements concerning when businesses must tell consumers that they are interacting with an automated or AI system are evolving. Even apart from specific AI laws, businesses should consider whether the interaction could be misleading if callers reasonably believe they are speaking to a person.

Confidentiality is another consideration. Voice interactions can involve names, phone numbers, addresses, financial information, health information, details about legal matters, or other personal information. Businesses need to understand what information the system collects, where it goes, which third parties receive it, how long it is retained, and what security protections apply.

The underlying business may also be subject to industry-specific requirements that have little to do with AI itself. Healthcare, financial services, legal services, and other regulated industries may already have rules governing the information being talked about or collected.

The important point is that "using AI" is not one legal issue. The risk comes from what the system is doing: communicating with people, making representations on behalf of a

business, recording conversations, collecting information, transferring data, and potentially taking actions. Identify and discuss those areas with appropriate legal or compliance professionals before implementation. Because this area is rapidly evolving, compliance should be ongoing.

Conclusion

Voice AI is getting better quickly. That does not mean every phone call should be automated. AI voice agents work best when they have a clear job, reliable information, a defined outcome, and sensible boundaries. They struggle when we ask them to exercise judgment they do not have, answer questions without the required context, or continue conversations after they have reached the limits of what they can usefully do.

That is why I would not start a Voice AI project by asking how to write the prompt. Start with the caller. Why are people calling? What are they trying to accomplish? Which of those tasks can an automated system realistically complete? What information would it need? What should happen when it cannot help? Only after you can answer those questions should you start worrying about how the agent should speak.

And once the system is live, listen to it. Real conversations will show you things no planning exercise will. They will expose missing information, ambiguous instructions, unexpected caller behavior, awkward timing, poor handoffs, and situations you didn't anticipate. Use those calls to improve the system, but make changes deliberately. Fix the problem you observed. Then make sure the fix did not create a new one somewhere else.

The goal is not to create an AI agent that can handle everything. The goal is to create one that reliably handles what it is supposed to handle and recognizes when it should get out of the way. If you can do that, you do not need perfect prompts. You have a useful system.