Comparing speech-to-text (STT) quality is not as simple as asking each voice agent the same question and deciding which answer sounds better. A voice agent can misunderstand a caller because its speech recognition missed a word, because the audio was poor, or because a later step—such as retrieval or response generation—used the transcript incorrectly. If you evaluate only the final answer, those failure points blur together.

A useful comparison separates the voice pipeline into stages, tests each platform under realistic call conditions, and records what happens when the agent is interrupted, delayed, or unsure. That includes testing phone lines as well as a website widget: audio quality, timing, and caller behavior can differ substantially between them.

Here's a practical way to compare speech recognition accuracy across voice agent platforms, including what to measure, how to build a fair test, and how to tell STT errors apart from knowledge-base problems.

Start by separating STT from the rest of the voice agent

A typical voice agent pipeline contains several steps:

1. **Audio capture:** A phone line or website widget receives the caller's voice and sends audio to the platform.
2. **Speech recognition:** The STT system turns the audio into text.
3. **Turn detection:** The platform decides whether the caller has finished speaking or is pausing to think.
4. **Interpretation and retrieval:** The agent identifies the request and may search a knowledge base or other connected system.
5. **Response generation:** The agent forms an answer.
6. **Speech output:** Text is converted into audio and played to the caller.

A failure at any step can look like a recognition problem. For example, a caller says, "I want to cancel the second order," but the transcript says "I want to cancel the order." That is an STT error. If the transcript is correct but the agent cancels the wrong order, the failure is more likely in interpretation, retrieval, or the action logic.

When you compare platforms such as ElevenLabs, ElevenAgents, or an agent connected to Zendesk, don't assume the whole experience is powered by one model or configured the same way in every deployment. Ask what speech recognition component is being used, whether you can inspect transcripts, and what control you have over turn detection and timeouts. Product names alone won't tell you where an error occurred.

Build a test set from real calls, not ideal sentences

Studio-quality demos are poor predictors of production performance. Real callers speak over background noise, change their minds, use unfamiliar names, and leave long pauses in the middle of a sentence. A fair test set should include these behaviors rather than a small collection of carefully read prompts.

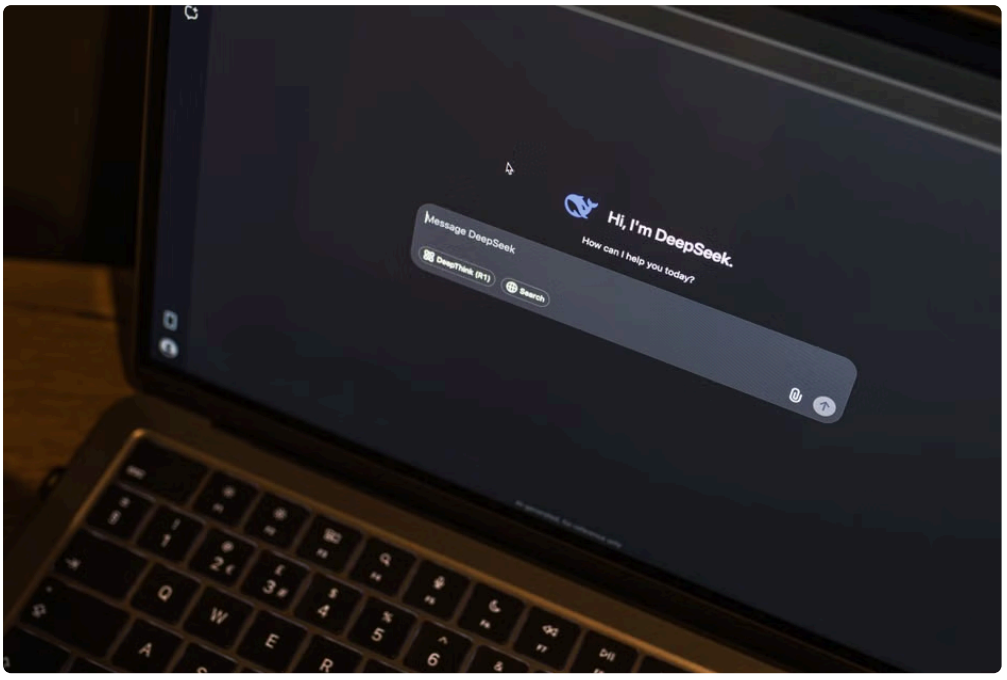
Include the calls that are hardest for your business

Collect representative audio or scripts, with appropriate consent and privacy safeguards, from the situations your support team actually handles. If you cannot use recordings, have people role-play them. Include

variations such as:

- Fast talkers and callers who run several thoughts together.
- Background noise from traffic, a kitchen, an open office, or a poor mobile connection.
- Accents, speaking styles, and microphone distances common among your customers.
- Product names, street names, email addresses, order IDs, and other uncommon terms.
- Corrections such as “Friday—sorry, I mean Thursday.”
- Hesitations, filler words, silence, and callers who speak while the agent is talking.
- Negations, quantities, dates, and other details that change the meaning of a request.

Keep the audio and its reference transcript consistent across platforms. If one platform hears a clean recording and another hears a noisy one, you are not comparing STT quality—you are comparing test conditions. For live phone tests, use the same phone line, handset, location, and script where practical. Then repeat the test with background noise. I listen once in a quiet room and once with noise in the environment because a system that succeeds in only one condition is not ready for every caller.



Run each sample multiple times if the platform’s output can vary. Record the platform version, configuration, connection, and test date. A result without those details is difficult to reproduce after a vendor changes a model or setting.

Measure recognition errors, not just whether the agent got the gist

A transcript can be “close enough” for a simple FAQ and still be unsafe for a payment, cancellation, or account change. Compare each transcript against a reference transcript, then inspect errors by type. Common measures include word error rate (WER), which combines substitutions, deletions, and insertions, and exact-match accuracy for critical fields such as an order ID or date.

What to measure What it tells you Why it matters Word error rate How often words are substituted, omitted, or added Useful for comparing overall transcription, but it does not show the consequence of each error. Critical-field accuracy Whether names, numbers, dates, and identifiers were captured correctly A single wrong digit or negation may cause a costly action. Intent-changing errors Whether the transcript changed what the caller was asking for Helps identify errors that affect routing, policy, or resolution. Condition-based

results How accuracy changes with noise, speed, pauses, or connection quality Reveals weaknesses hidden by an overall average.

Don’t let a single average score decide the winner. A platform may perform well on common words but poorly on the names, product codes, or short commands your callers use. Break results out by scenario. Also check whether punctuation or capitalization differences are being counted as meaningful errors in your evaluation; they usually matter less than a wrong amount or a missed “not.”

Evaluate latency and turn-taking, not only transcript quality

Fast, accurate recognition is useful only if the agent takes turns naturally. In a live call, callers notice the time between finishing a thought and hearing a response. They also notice when the agent responds to a pause as if the caller had finished, or sits silent after the caller has clearly stopped.



Measure the interval from the end of the caller’s speech to the agent’s first audible response. Count dead-air seconds out loud during test calls if you have to; it makes the wait concrete. Record the median and the slowest cases rather than relying on one best run. A short delay may be acceptable for a complex request, but repeated long silence can make callers think the line has dropped.

Test turn-taking with realistic pauses. Ask a question, pause briefly, then continue with the important detail. Does the platform start answering too early? Does it wait too long after a complete sentence? Try the same scenarios through a phone line and a website widget. A browser microphone may capture different noise and timing than a telephone connection, so success on one channel does not guarantee success on the other.

Stress-test interruptions and timeout settings

Interruption handling is where polished demos often stop being convincing. In a real support conversation, a caller may say “no, that’s not what I meant” while the agent is speaking. The agent should stop or yield promptly, listen to the correction, and use it. If it continues talking over the caller, a strong transcript score will not rescue the experience.

Test barge-in with short, ordinary interruptions and longer corrections. Try speaking over the agent at different points in its response. Note whether it stops, whether it captures the interruption, and whether it

changes course. Also test pauses that are not interruptions: callers may be searching for an account number or thinking through a question.

Timeouts and end-of-turn thresholds influence these results. A short silence threshold can make the agent jump in early; a long one can create awkward dead air. Ask the vendor where these settings live, what can be configured, and whether you can see when the system decided a turn was complete. Hidden settings make it harder to diagnose whether a problem is recognition, turn detection, or a platform default.

Include awkward caller scripts in your evaluation: “I need to change my delivery—actually, wait, not the delivery, the billing address.” These are more revealing than a clean, uninterrupted question because they test recognition, correction handling, and turn-taking together.

Separate STT errors from knowledge-base and RAG failures

When an agent gives a wrong answer, inspect the evidence in order:

1. **Listen to the audio.** Confirm what the caller actually said and whether the audio was understandable.
2. **Read the transcript.** Look for missed words, wrong numbers, substitutions, or a lost correction.
3. **Check the interpreted request.** Did the agent understand the intent that the transcript expressed?
4. **Inspect retrieval.** Did the knowledge base return the right document or answer?
5. **Review the final response or action.** Did the agent follow the available information and your business rules?

This distinction matters for any retrieval-augmented generation (RAG) setup. If the transcript is accurate but the agent uses an outdated Zendesk article, that is not an STT failure. If the caller says “I haven’t received a refund” and the transcript loses “haven’t,” the knowledge base may be perfectly correct while the agent answers the opposite question.

Use paired tests to isolate the cause. First, run the call through the live audio path. Then provide the verified reference transcript directly to the agent, if the platform supports it. If the agent answers correctly from the reference transcript but not from the audio, investigate speech recognition or turn detection. If it still answers incorrectly, examine retrieval, instructions, connected systems, and response generation.

Use one scorecard across every platform

For a useful comparison, keep the same criteria and assign importance based on the risk of your use case. A support team handling account changes may value critical-field accuracy and interruption handling more than a small improvement in general WER. A team answering basic opening-hours questions may weight overall recognition and response time differently.

- **Recognition:** Overall transcript quality and accuracy on important words and fields.
- **Robustness:** Performance with fast talkers, background noise, pauses, and less common vocabulary.
- **Turn-taking:** Appropriate response timing and reliable handling of interruptions.
- **Observability:** Access to transcripts, timestamps, failure logs, and relevant configuration settings.
- **Resolution:** Whether the complete interaction answers the request or completes the intended action correctly.

Keep raw examples alongside scores. A spreadsheet that says one platform had a lower WER is less useful than a note showing that it repeatedly dropped the word “second” from order requests. Score both the

individual pipeline stages Look at more info and the end-to-end experience so you can make a purchasing decision without confusing one kind of failure for another.

A practical decision rule

Before choosing a platform, define the errors you cannot tolerate, such as changing a negation, mishearing an account number, or interrupting a caller who is correcting an important detail. Set minimum acceptable results for those cases, then compare platforms using the same recordings, channels, and conditions. Include a live pilot on phone lines and, if relevant, a website widget.

The best STT choice is not necessarily the one with the most impressive demo or the lowest average word error rate. It is the platform that reliably captures the speech your customers actually produce, makes its behavior diagnosable, and gives the rest of the voice pipeline enough accurate information to respond safely. Test the noisy calls, the fast talkers, and the awkward corrections before rollout—not after your support team starts hearing about them.