Episode Book

EPISODE #6510

Voice Isolation Before Speech Recognition

PUBLISHED	9 November 2026 at 15:14 UTC
DURATION	20:45
LISTEN	myweirdprompts.com/episode/voice-isolation-speech-recognition-71312/
HOSTS	Corn, Herman, Hilbert
SCRIPT MODEL	z-ai/glm-5.3
VOICES	chatterbox-regular (chatterbox-local)
PIPELINE	V5.26
BOOK GENERATED	10 November 2026 at 10:05 UTC

SYNOPSIS

Voice isolation models strip out competing speakers before speech reaches voice activity detection and speech-to-text engines. While models like VoxLock, Scicom, Hush, and ai-coustics' Quail Voice Focus dramatically improve transcriptions in noisy multi-speaker settings, running them on clean audio consistently degrades transcription accuracy. From automated enrollment heuristics and decoy recordings to real-world latency tradeoffs across streaming frameworks, this look at modular audio pre-processing evaluates whether isolation filters belong in front of a lone dictation microphone.

TRANSCRIPT



Corn

Daniel was wandering the audio-to-audio section of Hugging Face last night, which is apparently a thing people do, and found a model category we've never touched: voice isolation. Not noise suppression, not diarization. Models that sit in front of your recogniser and strip out every voice except one.



Herman

And he put the interesting ones in a collection. VoxLock, Scicom's voice-isolator, ai-coustics' Quail Voice Focus, and Weya's Hush. Most of them are weeks old.



Corn

His own words: we did episodes recently on getting ASR to focus on one speaker, through target-speaker conditioning inside the recognition model, "and we completely missed this approach." He's right, we did. A separate isolation stage in front of the recogniser never came up.



Herman

Which stings, because it's the more modular of the two ideas.



Corn

He's also thinking about his own chain. He dictates to a fine-tuned Parakeet through a laptop mic, and he wants to know how these things decide which voice is "you," how the four compare, and whether one can actually run alongside a VAD in front of his model. Let's dig in.



Herman

Start with what the category actually is, because the name undersells it. A voice isolation model takes raw audio in and puts cleaned audio out, containing only the target speaker. Nobody downstream ever sees the other voices. Your VAD doesn't see them, your recogniser doesn't see them.



Corn

As opposed to the conditioning approach we covered, where the recogniser itself is told who to listen to.



Herman

Right. With target-speaker conditioning, you feed the ASR model an embedding, a voiceprint, and the model learns to attend to that voice while it transcribes. Google's Personal VAD is the frame-level version of that. It's tiny, about a hundred and thirty thousand parameters, and it just gates the audio into target, non-target, and non-speech. It never cleans anything. It just decides what gets through the gate.



Corn

And this new category cleans first.



Herman

It cleans first, and the whole chain after it is dumb again. Any VAD, any recogniser. That's the appeal. You can swap the isolator without retraining your Parakeet, swap the Parakeet without touching the isolator. Plug and play.



Corn

Where does it sit relative to plain denoising? Because DeepFilterNet and RNNoise have been around for years.



Herman

Denoisers have no notion of whose voice. They remove traffic, keyboards, fans, anything that isn't speech. If two people are talking, both stay. Then there's blind separation, which takes a mixture of N speakers and outputs N separate streams, but it doesn't tell you which one is yours. Target speaker extraction is the one that uses an enrollment embedding to pull out one specific voice, and that's really what most of these new models are, with a twist on the enrollment part.



Corn

Which is exactly the twist Daniel's asking about. How do they know who "you" are.



Herman

And that's where it gets good. VoiceFilter, the old Google paper from twenty eighteen, assumed you already had a clean sample of the target's voice. These new ones mostly don't assume that. VoxLock enrolls from the call itself. Its isolate mode takes the first fifteen seconds of audio, or a sample you supply of three seconds or more. Its live mode is the honest one: it passes audio through completely unprocessed until it has heard three seconds of the caller, and only then starts isolating.



Corn

So the first three seconds of every call go out raw.



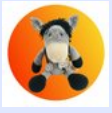
Herman

Everything. Whoever's talking, whatever's on in the background. Then the lock clicks in. Scicom's version is similar but more aggressive about it. From three seconds into the call, it keeps the loudest three seconds it has heard so far, and that choice is fixed once the call is fifteen seconds old. Whatever was loudest in that window is now the caller, permanently.



Corn

Loudest. Not closest, not first, not the voice that responds when the agent asks a question. Loudest.



Herman

It's a heuristic, and it's cheap, and most of the time on a phone call the caller is the loudest thing. But you can see the edges of it. They also have a voicebot mode, opt-in, for agents that speak first, because an agent that opens with "Hi, how can I help?" will enroll on its own greeting. So the mode plays a decoy voice to hold the model until a real caller shows up.



Corn

A decoy voice. The model is fobbed off with a hostage recording.



Herman

And the cost is printed right on the card: a caller who starts speaking in the first second loses twenty-five to thirty-one percent of that first turn. The model is still holding onto the decoy when the caller starts, so their opening words get eaten.



Corn

Nothing for free in that trade.



Herman

None of it. Hush, the small one, its card just says it isolates exactly one speaker, and doesn't say how it picks. My read is there's no enrollment at all, it just keeps the dominant primary speaker. And ai-coustics' Quail Voice Focus isolates "the foreground speaker" with no enrollment described anywhere, mechanism undisclosed. It's a closed binary, you get it through their SDK.



Corn

So on the spectrum of who-decides-you're-you: VoxLock and Scicom look at the audio and make a bet, Hush probably just grabs whoever is front and center, and ai-coustics won't say.



Herman

That's the landscape. And the enrollment mechanism is really the whole design problem in miniature.



Hilbert

Hilbert's Mid-Show Haiku. Enrollment. Three seconds of a stranger, then the room goes quiet. The machine keeps one voice and lets the rest fall. It does not ask twice.



Corn

It really doesn't.



Herman

The models themselves are wildly different in size, which matters for what Daniel can run. VoxLock is twenty-seven million parameters, a fine-tune of a public causal target-speaker extraction model called REAL-TSE, band-split RNN architecture, using WeSpeaker ECAPA embeddings for the speaker side. The whole thing was fine-tuned on a single NVIDIA L4. One GPU. Which tells you the heavy lifting was done by the base model and the training data was simulated LibriSpeech calls plus real crosstalk from AMI meetings.



Corn

Simulated calls. They built fake conference calls to teach it who to keep.



Herman

Mixed clean recordings with background talkers and simulated room acoustics, then patched it with real meeting audio. Scicom is the opposite philosophy. Five hundred and fifty-five million parameters, a causal transformer bolted onto a causal WavLM-Large. They trained smaller masking transformers first, a hundred and forty million, then two-forty-three, then three-seventy-four, and only when they made WavLM itself causal and trained the entire business jointly did the error drop further.



Corn

And the little one?



Herman

Hush is eight megabytes. Roughly one point eight million parameters, built on the DeepFilterNet3 design, the same filterbank and gated recurrent setup. Trained on over ten thousand hours of mixed audio, with a competing speaker in about sixty percent of the samples. It's Apache-licensed, CPU only, under a millisecond of compute per ten milliseconds of audio.



Corn

Eight megabytes. My leaf-drying playlist is bigger than that.



Herman

Latency follows the same spread. VoxLock has twenty-four milliseconds of algorithmic latency plus whatever chunk size you feed it, but the measured numbers in actual frameworks are higher, around a hundred to a hundred and eighty milliseconds in Pipecat. Scicom has no look-ahead beyond its twenty-millisecond analysis window, and in LiveKit with thirty-two concurrent calls it runs about forty milliseconds behind at the median. Hush is about twenty milliseconds of algorithmic latency. ai-coustics quotes thirty.



Corn

Those framework numbers are the interesting ones. Twenty-four milliseconds of algorithmic latency is a spec. A hundred and eighty in Pipecat is what you actually get.



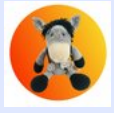
Herman

The plumbing dominates. Buffering, handoffs between components. Same model, wildly different lived experience depending on the framework.



Corn

Benchmarks. Daniel will want the actual comparison.



Herman

The Krisp Voice Isolation Benchmark is the shared reference point. Two hundred sixty-five real recordings, no synthetic mixing, offices and homes and call centres and street calls, each transcribed by hand, and the recogniser is Parakeet TDT 0.6B v2, which is nice for us because that's Daniel's family of model.



Corn

Same species as his fine-tune. Good.



Herman

Raw unprocessed audio comes in at twenty-three percent word error rate. VoxLock takes that to four point nine five in its best mode, with about one background word leaking through every nine minutes. Run live, streaming, it's five point six three. Scicom's GPU model, live, gets five point seven four, its CPU distillation five point six six. For context, Krisp's own commercial filter is around five point three. So the open models are in the same room as the incumbent.



Corn

And these are weeks old. VoxLock was fine-tuned on a single GPU.



Herman

On messier audio the gap holds. AMI meeting recordings, raw is nearly forty-three percent, VoxLock twelve, Scicom ten. A call-centre floor with a neighbouring agent as the competing voice, raw is seventy percent, and even after isolation it's still around fifty. That one's not solved. But then there's the number that should stop Daniel cold.



Corn

Caller alone. No competing voice.



Herman

Raw is zero point zero four percent word error. Essentially perfect. VoxLock turns that into two point nine two. Scicom into five point five eight. Isolation makes clean audio worse.



Corn

It damages a signal it was supposed to be improving.



Herman

It's a filter running on a signal that didn't need filtering. And Krisp's own numbers show the same thing on their side: their default filter takes clean phone audio from three point four eight percent to three point nine one. One file went from nine percent to twenty-one. They list it as an open problem.



Corn

So the trade is real and nobody has solved it, including the people selling it.



Herman

There's a deeper version too, from Scicom's VAD test. They ran a hundred and twenty one-minute calls with a background voice and measured what happens downstream. Speech leaking into the caller's pauses fell from about half the pauses to under a quarter. False starts by the agent dropped from three point three per minute to under two. Those are real wins for a voice agent. But missed caller speech rose from a tenth of a percent to eight point four.



Corn

Eight percent of the caller's own words, gone.



Herman

Some of the suppression lands on the caller. And Scicom says so themselves, which I respect: the two stated failure modes are losing the caller when someone else talks for a while, and degrading a caller who talks alone, and fixing one makes the other worse.



Corn

That's the whole category in one sentence. Now, Daniel's chain.



Herman

His chain today is presumably microphone, VAD, Parakeet. The question is whether you can insert isolation at the front. Mechanically, yes. Scicom explicitly recommends placement before VAD, before turn detection, before STT. VoxLock ships as a Pipecat audio input filter, a LiveKit noise-cancellation option, sherpa-onnx, and plain offline Python. Scicom has LiveKit plugins for GPU and CPU, plus a WebSocket server. Hush is ONNX with prebuilt libraries for all three desktop operating systems. So the plumbing exists.



Corn

The plumbing exists and most of it is aimed at voice agents, not dictation.



Herman

Which is the honest caveat. Everything here is tuned for sixteen-kilohertz phone audio and agent-caller conversations. Nobody's card says a word about a lone user dictating on a laptop microphone. None of them.



Corn

For a lone dictating user, what does enrollment even look like? He's not a caller arriving cold.



Herman

This is where it gets practical. The live self-enrollment modes are the wrong fit. If VoxLock's live mode passes everything through for three seconds, then on a dictation session it enrolls on whatever it hears first, which is Daniel, fine, but if Ezra wanders in talking during those three seconds, the model may lock onto the wrong voice and spend the whole session filtering Daniel out of his own dictation.



Corn

Dictating father, transcribing son.



Herman

The supplied sample is the sensible route. VoxLock's stream mode takes a three-second recording of the target voice and isolates from the very first chunk. Scicom accepts a supplied caller clip too. You record three seconds of yourself once, keep the file, feed it in at the start of every session.



Corn

So the chain becomes microphone, isolation, VAD, Parakeet. Does the VAD survive the insertion?



Herman

That's the part he'd have to retune. The VAD numbers say isolation changes what the VAD sees in both directions. Background voices in his pauses mostly vanish, which is good if he's dictating in a room where Hannah's on a call. But the VAD may start missing the ends of his own utterances, because the isolation has shaved them. His threshold would need adjusting after the change, not before.



Corn

Compute. He's on a laptop.



Herman

The CPU variants all fit. Scicom's distilled model is twenty-two million parameters and needs about half a core at ten-millisecond frames, a third of a core in int8. Hush needs almost nothing. VoxLock int8 is the heavy one, over a hundred percent of a core at twenty-millisecond chunks, dropping to about sixty at eighty-millisecond chunks. Feasible, but you're spending real battery on a stage that might be hurting him.



Corn

Which is the question the benchmarks already half-answer.



Herman

They do. On clean caller-alone audio, isolation moved word error from essentially zero to around three or five percent. Daniel dictates alone, usually, on a laptop mic. Absent a competing speaker, every data point we have says the filter is a net negative for exactly his case.



Corn

So my recommendation writes itself. He tests it, and he tests it against his own baseline first.



Herman

And the test is cheap, that's the good news. VoxLock and Hush are Apache-2.0, so he can drop either into his chain for an afternoon, run his usual dictation set through it, and compare against his fine-tune's output directly. Scicom's licence is listed as other with no terms visible, which for a personal experiment is probably fine but worth a look before anything ships.



Corn

One more thing the benchmarks hide. Who built them.



Herman

Krisp built the benchmark, and Krisp sells a voice isolator. The open models were tuned and evaluated against it. VoxLock's claims are self-evaluated, the card doesn't even have a limitations section. It's not fraud, the numbers are consistent across independent test sets, but it's the whole industry grading its own homework because until recently there was barely any public benchmark for background speech.



Corn

Which is presumably why the category sprouted all at once. Somebody finally put a scoreboard up.



Hilbert

The loudest-three-seconds rule is the part you're both being polite about. Herman, you said "heuristic" like it was a compliment. I ran one of these for a week last month, the Scicom one, on my own dictation. It enrolled on my cat. The cat is louder than I am and more consistent in the opening seconds, so the model concluded the cat was the primary speaker and dutifully removed me from my own audio. The transcript was the word meow, rendered variously, for three sessions before I worked out what was happening. I re-recorded the enrollment sample in the kitchen with the door shut on her and that fixed it.



Corn

Three sessions of meow.



Hilbert

The third one I kept. It's a clean recording of her, better than the one I made on purpose. Now I dictate with a supplied sample of my own voice, three seconds, re-recorded every Monday morning, because voices change. There's a folder of them on the desktop already. Four files, dated. Claws on a laptop keyboard being what they are, I record it on the phone, which sits in a clamp.



Herman

The Monday re-record is actually more rigorous than any of these cards ask for.



Hilbert

The cards ask for nothing. That's my point. And the cat has not volunteered to change her volume, so the arrangement stands. She's a good animal, for all that. Better at her job than half the people I've worked with.



Corn

Monday morning, a dated folder of voice samples. That man has a cleaner enrollment protocol than two of the four models.



Herman

He does. A supplied, recent, single-speaker sample beats loudest-three-seconds every time, and the models that accept a supplied clip are exactly the ones worth using.



Corn

So the practical advice to Daniel might be: don't add one at all.



Herman

For his default case, probably not. Dictating alone on a known mic, the clean-audio penalty is a tax with no benefit. The case where it pays is the case he didn't describe: dictating in a room where other people talk. Then the raw numbers swing hugely. If he ever works from a cafe or a shared office, that's when a three-second Monday sample and an isolator earn their keep.



Corn

And the misconception to leave people with, since it's the one I held walking in.



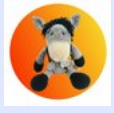
Herman

Say it.



Corn

That voice isolation and noise suppression are the same product. They're not. Your denoiser removes the world. Your isolator removes the other people. One has no idea whose voice it's keeping, the other's entire job is choosing, and everything hard about it lives in that choice.



Herman

And the second one people miss: that they always help. The best case for these models is a polluted signal. On a clean one, they're the one making it dirty.



Corn

Where this goes next is worth watching. If the voice-agent frameworks keep standardising these filter slots, isolation becomes a default stage in front of every recogniser, the way VAD already is. The enrollment problem and the clean-audio penalty get solved or the whole category stalls there. The Apache models make trying it nearly free. Producer credit to Hilbert Flumingtop, who has the best-labeled folder in Jerusalem. If you want more of this, try episode eight, Building Your Own Whisper; and episode seven, Building Custom ASR Tools. This has been My Weird Prompts. See you tomorrow, Herman.



Herman

See you tomorrow, Corn.