Episode Book

EPISODE #6467

Voice Activity Detectors Need Adaptive Thresholds

PUBLISHED	5 November 2026 at 20:15 UTC
DURATION	24:58
LISTEN	myweirdprompts.com/episode/voice-activity-detection-adaptive-thresholds-76228/
HOSTS	Corn, Herman, Hilbert
SCRIPT MODEL	z-ai/glm-5.3
VOICES	chatterbox-regular (chatterbox-local)
PIPELINE	V5.26
SOURCES	25 cited by the research (listed at the end)
BOOK GENERATED	6 November 2026 at 11:12 UTC

SYNOPSIS

Voice activity detectors like Silero rely on fixed sensitivity thresholds that break when room acoustics or microphones change. Default settings cause clipped word boundaries, false triggers on HVAC systems, and downstream speech recognizers that hallucinate sentences from background noise. Tuning requires adjusting six post-processing parameters—including hysteresis and minimum silence duration—rather than a single scalar. Despite research on noise-tracking adaptive gating dating back to 2007, mainstream open-source tools still pass static configuration burdens to end users.

TRANSCRIPT



Corn

What if the one number that decides whether your microphone is listening is the one number nobody should be allowed to set?



Herman

That's a good cold open, and it's also basically what Daniel sent us.



Corn

Daniel's been living with this one. He uses Silero and TEN VAD, they work, but the sensitivity threshold is where it all goes sideways for him. Coffee shop, he wants a bouncer on the door. Quiet room at home, he opens it up — until the air conditioning gets a speaking part. And his point is bigger than the anecdote: out-of-the-box defaults get this wrong, and a person using a dictation app should not have to go in and tune a voice activity detector. He's asking whether adaptive gating exists — something wrapped around the VAD that works out where you are and sets the threshold itself — whether a diagnostic interface showing what's being flagged as speech already exists, because he's thinking of adding one to Hebrish, which, by the way, is out now for anyone who wants Hebrew-English dictation that doesn't fight you. And which open-source VADs are furthest ahead on hands-off configuration.



Herman

Hebrish is the Hebrew-English dictation tool, for anyone catching up.



Corn

Right. Now — last time we touched VAD it was whether the gate belongs in recognition at all. Today it's about what counts as speech in the first place.



Herman

The threshold is the operating point of the whole pipeline. Silero puts out a probability per frame — how speech-like is this thirty milliseconds — and the threshold, default 0.5, is the line where you say above counts, below doesn't. Every single thing downstream inherits that one decision. Get it wrong and the best recognizer in the world is working on garbage, or on silence.



Corn

Or on the air conditioner.



Herman

Here's the part I love, and I use "love" in its clinical sense. The air conditioner is in the vendor documentation. Switchboard's Silero integration guide says, don't go below 0.3 without testing in your target environment, because at very low values, music and HVAC will trigger detection. HVAC. They named Daniel's exact appliance.



Corn

Nineteen years of machine learning and the manual still says "mind the boiler."



Herman

And the structural problem is right there in that sentence. The correct value depends on the microphone and the room. The interface is one scalar, set once, shipped to everyone. Those two facts can't coexist happily. The knob is pretending to be universal and the correct answer never is.



Corn

So the question isn't why Daniel got it wrong. It's how anyone gets it right.



Herman

And the arc today — what the knob actually is, because it's not one knob, that's the first surprise. Then what defaults break, measured. Then whether adaptive gating exists, and I'll spoil it: it exists, and it's older than the iPhone. Then the diagnostic side, and the field of VADs.



Corn

The first correction, then. You said the knob isn't a knob.



Herman

Six knobs. Silero's post-processing surface: the threshold at 0.5, a negative threshold that gives you hysteresis — a different line for entering speech than for leaving it, so a borderline frame doesn't flip you on and off every thirty milliseconds. Minimum speech duration, two hundred fifty milliseconds — anything shorter doesn't count. Minimum silence duration, a hundred milliseconds, how long the quiet has to last before we close the segment. Speech padding, thirty milliseconds of lead-in and tail. A max speech duration, which defaults to infinity, and a window size of five hundred twelve samples.



Corn

Six ways to be wrong, and the app store review just says "voice detection buggy."



Herman

Here's the correction to Daniel's framing, though — and he'll appreciate it, because it changes what Hebrish should adapt. Switchboard's own docs argue the most impactful parameter isn't the threshold. It's the minimum silence duration. A hundred milliseconds default is too short for actual human speech, because natural inter-word pauses run two hundred to five hundred milliseconds. You say a word, you breathe, the default decides you're done.



Corn

So the VAD keeps slicing one sentence into six segments.



Herman

Their recommendation is start at three hundred milliseconds, five hundred for deliberate speech, seven hundred if the speaker pauses a lot. And if the first word comes out clipped, add padding, a hundred fifty to two hundred fifty milliseconds. That's a per-speaker setting, not even a per-room one.



Corn

Which means an adaptive gate that only adapts the threshold is tuning the wrong string on the guitar.



Herman

That's the design constraint for Hebrish, honestly. If Daniel builds the front-end wrapper, it has to reach all six dials, or it's solving a sixth of the problem.



Corn

What actually happens with defaults, though? Anyone measure it, or is this all vendor folklore?



Herman

Measured. Abdul Muneer ran exactly the experiment you'd want. He cached Silero's frame probabilities once — so the model runs one time, no re-inference — and then swept the post-processing parameters offline against a held-out set. Which is the cheap way to do it, and it means every difference you see is the knob, not the network.



Corn

Clever. The model never moves; only the gate does.



Herman

And the tuned settings moved consistently in one direction. Higher threshold pair for entering and leaving speech, longer minimum speech and silence durations, more padding. His phrase for it was a stricter, less twitchy gate. Non-speech false positives dropped substantially, and speech recall — the words actually caught — stayed flat. You gave up nothing and stopped letting the room talk.



Corn

The padding part matters too, doesn't it. Clipped word edges into a recognizer.



Herman

That's the quiet failure. A recognizer seeing the first half of a word guesses. Trimmed audio in, hallucinated word out, and you never see the connection because the VAD log says everything worked.



Corn

Here's the comparison that makes it land for me. He tested tuned Silero against another VAD at defaults.



Herman

FunASR's FSMN-VAD. Perfect speech recall — it caught literally everything — and it accepted far more non-speech. Muneer's line on it: nearly the same headline recall, yet what reaches ASR differs substantially, and that difference comes from the default operating point, not from the architecture. Which is the whole episode in one finding. People comparison-shop models.



Corn

So Daniel's coffee shop problem — let me get the actual stakes right, because I think it's worse than annoyance.



Herman

It's much worse. His line: given a few seconds of pure noise, most ASR models emit a plausible sentence rather than nothing. The VAD is the main defense against that. The model doesn't fail gracefully on noise — it confabulates. So every false VAD activation isn't noise added to your transcript. It's a fabricated sentence written down as if you said it.



Corn

The app testifies against you in fluent prose.



Herman

In a coffee shop, with a loose threshold, the machine is transcribing the espresso machine's testimony.



Corn

And the microphone angle Daniel intuited — that the right threshold is a function of the hardware too — that's documented as well?



Herman

Down to the failure mode. Switchboard's troubleshooting says a VAD that never fires an end event usually means continuous noise above threshold, and the classic culprit is an overdriven microphone. The clipping artifacts read as continuous speech energy. So you sit there tuning the threshold down, down, down, and the whole time you've got a microphone problem wearing a threshold costume.



Corn

That's Herman's whole DJ career in one symptom, isn't it — ninety percent of "the PA sounds bad" is gain staging before anyone touches EQ.



Herman

I'll take that as a compliment, because it's true. The fix isn't the threshold at all, it's the level at the front, or noise suppression upstream. Their docs literally recommend RNNoise in front of Silero when the noise floor is high enough that raising the threshold would swallow soft speech.



Corn

So the single knob is a single knob the way a light switch is a building plan. And the last piece — why can't the user just tune it? Six dials, sure, but a determined user with a weekend.



Herman

A determined user with a weekend can tune it for Saturday's room. The pitfalls are non-obvious — clipped word edges, invented text, segments that never end — and the correct setting is a property of the room and the mic, not of the person. There's no "right" value for the user to find, only a right value for their kitchen. Picovoice actually frames the knob as the feature: probability scores give developers flexibility to set application-specific thresholds, implement hysteresis for stability, build custom logic for edge cases, monitor confidence levels in production.



Corn

Read that list again and count the developers.



Herman

Four.



Corn

It's a fine answer for the person with the source open and a terrible answer for the person with a dictation app and a coffee.



Herman

And that gets us to Daniel's actual question. Does adaptive gating exist — something that works out where you are and sets the threshold itself?



Corn

You said it exists, and you had that look you get when the answer is going to be a year.



Herman

Two thousand seven. Interspeech. Sangwan and colleagues, "Environmentally Aware Voice Activity Detector." Their opening diagnosis reads like Daniel's email: traditional VADs utilize a single operating point for all SNRs and noise types, and attempt to learn the background noise model online from finite data length. Deaf to the room. Their fix — build offline knowledge of common acoustic backgrounds, classify the noise type from the first two hundred milliseconds of audio, and set the operating parameters from that classification.



Corn

The first two hundred milliseconds. So the thing listens to the room before it listens to you.



Herman

And they reported ten to fifteen percent absolute detection-rate improvement over the AMR VADs at low signal-to-noise ratio. That is literally "work out where you are and set the threshold yourself." Published when the iPhone was a rumor.



Corn

So the research solved it, and then everyone went off and shipped a slider anyway.



Herman

The lineage didn't stop, either. DySANA, Weiss and Kristjansson — a Kalman filter with switching observations that tracks the noise level and the speech level simultaneously, with a Lombard dynamic distribution constraining how fast the levels can move and the ratio between them. Forty-six percent relative word error rate improvement over a Gaussian endpointer at twenty dB. There's a twenty sixteen paper that holds a constant false acceptance rate by continuously updating a noise-level estimator. The long-term spectral divergence family adapts its threshold to measured noise energy. This is not an exotic corner of the literature. It's a well-trodden one.



Corn

Now the embarrassing question. If the idea is nineteen years old, what does it look like in shipping code?



Herman

Three lines. There's a Go project, siprec, with a realtime VAD that keeps a background-noise estimate, updated only while nobody's talking, smoothed with a coefficient of 0.1 — and then sets the adaptive threshold to three times the background noise, with a floor so it never goes absurdly low in silence.



Corn

Times three. That's the whole trick.



Herman

The room tells you the threshold. Nobody had to classify anything; they just let the noise floor multiply.



Corn

I want to sit with how that makes me feel about the entire field.



Herman

I know. And here's the gap, and I'm going to be careful stating it, because "I couldn't find it" isn't evidence. But no mainstream open-source VAD auto-detects the environment and sets its own threshold as a shipped default. Silero exposes the fixed probability. TEN VAD exposes the fixed probability. Cobra exposes the probability and sells it as flexibility. FireRedVAD exposes a speech threshold with better defaults. Picovoice's twenty twenty-six guide still tells developers to test in realistic noise and adjust the threshold. The nearest things to adaptive gating anywhere are the 2007 paper, DySANA, and the times-three heuristic in one Go repo.



Corn

So it's not that nobody knows how. It's that nobody puts it in the box.



Herman

That's the finding. The research is done, the code is three lines, and every major VAD hands you a scalar and a blessing.



Hilbert

Hilbert's Anteater Trivia. Anteaters make little stomach acid of their own and are thought to rely partly on the formic acid of the ants they swallow. I have never once relied on anyone else's acid, and I have the records to prove it, filed under things nobody asked for.



Corn

Noted for the file.



Herman

The diagnostic question is the one I was most looking forward to, because Daniel's imagining this interface for Hebrish — something that shows when the threshold is being exceeded, so you can see what's being flagged as speech — and the blueprint already exists.



Corn

How close are we talking?



Herman

Almost embarrassingly close. Aboleth STT ships a Silero Waveform Analyzer. You enable probability logging in the debug settings, speak, it writes a CSV of per-frame VAD data, and you drag that CSV onto a web page. What you get: a probability trace over time, shaded regions where the frame was called speech, a dashed line for the threshold, vertical markers for onset and offset, and pills showing the committed words and the final transcription at their actual timestamps.



Corn

Stop. You can drag the threshold line?



Herman

And watch detection change. You hover any point and get the exact timestamp, probability, and speech state. It is Daniel's idea, feature for feature — except it's a post-hoc viewer for a game-engine plugin, aimed at the developer, not at the person dictating.



Corn

So the picture exists, and it lives in the wrong building.



Herman

There are others. WaveKat Lab is an early-development web harness that fans one mic out to multiple VAD configurations at once, with per-config preprocessing — high-pass, denoising, normalization — and synchronized waveform, spectrogram, and probability timelines you can zoom and pan. A PyQt tool that runs WebRTC against Silero side by side with confusion matrices. A browser demo with a probability bar and threshold sliders.



Corn

All of them developer tools.



Herman

Every single one. Nobody has put the "here's what got flagged as speech" overlay inside a shipping dictation product. Daniel's Hebrish overlay would be adaptation, not invention — which, honestly, is the best news in his whole message. He's not betting on an unproven interface; he's porting a proven one across the street.



Corn

The teachable part is the dragging, isn't it. The user sees the air conditioner clear the line, drags the line above it, and in ten seconds understands more about their room than any settings screen explains.



Herman

That's why I'd argue it's not even primarily a diagnostic. It's the missing onboarding. The first session with a dictation tool should show you what your kitchen sounds like to a machine.



Corn

Which leaves the big one. Bake it in — can the user never think about VAD at all?



Herman

Muneer has the structural answer, and it's not "one smarter threshold." His point: strict live gating and high-recall offline batch pull in opposite directions, so a single VAD at a single operating point will be wrong for at least one path. His recommendation is per-path operating points — different VADs, or the same VAD tuned differently, for live versus batch. And his kicker: VADs are cheap to load and run, so the cost of doing this is low.



Corn

So the architecture is two gates, not one adaptive one.



Herman

Or several. The point is the product chooses the operating point per path, and the user chooses nothing.



Corn

Then the field. Who's forward-looking?



Herman

On raw numbers, FireRedVAD, out of Xiaohongshu's team, this year. They benchmarked on FLEURS-VAD-102 — over nine thousand files, a hundred and two languages — and the number that matters for Daniel's coffee shop is the false alarm column. FireRedVAD, 2.69 percent. Silero, 9.41. TEN, 15.47.



Corn

TEN's the one Daniel uses.



Herman

And it's the false-alarm champion of the neural trio — it also has the lowest miss rate, 2.95, so it's tuned generous. Catch everything, including the HVAC. FireRedVAD sits at 97.57 F1 with a miss rate of 3.62, which is a better corner of the tradeoff. But — and this is the philosophy problem — FireRedVAD still exposes a speech threshold, default 0.4, plus smoothing and minimum-frame settings. Better defaults, same religion.



Corn

So the leaderboard leader still ships the knob.



Herman

Same manual-knob philosophy, tuned closer to right. The interesting one on the hands-off axis is Earshot, from pykeio. It barely pitches a tuning surface at all — its claim is robustness: resilient to most environmental noise with signal-to-noise ratio of three dB or better. Sixteen-millisecond frames, runs three thousand six hundred times real time, claimed forty times faster than Silero v6 and TEN. The whole thing is a forty-kilobyte network in a ninety-five-kilobyte binary.



Corn

Forty kilobytes. Silero's two megabytes and it wants an eight-megabyte runtime.



Herman

And Earshot doesn't need one. It's Rust, no standard library, tiny per instance. But notice what it's actually claiming — not "we set your threshold," but "you won't need to." If the model is robust down to three dB SNR, the room stops mattering, and the knob stops mattering with it.



Corn

Which is a different bet than the wrapper. Silero's own track record says they know which side they're on — the v6.2 releases are dependency cleanup and a sequence model for throughput. Nobody's adding environment detection.



Herman

Silero's maintainers describe the model as deliberately stupid — it only says whether there's speech or not, all the environment sensitivity lives in the post-processing. So they've scoped themselves out of this on purpose. The cleverness is yours to supply.



Herman

Where do you land, then? If Hebrish were yours — the wrapper or the robust model?



Corn

The robust model, with the overlay. The wrapper is three lines, but it's three lines I have to trust in every room I've never been in. Earshot's claim, if it holds — I'd rather not think about VAD at all than have it think about me slightly faster.



Herman

You realize that's the first time you've picked the new thing over the repairable old thing since we started this show.



Corn

Silero is still here. I'm choosing it for a colleague's app, not for my kitchen.



Hilbert

Times three and a floor, Herman. You left the floor out. Without the minimum it never opens in a quiet room, and you sit there waiting on a gate that thinks silence is suspicious.



Herman

Fair — the floor matters as much as the multiplier.



Hilbert

It always did. I ran a voice-activated Uher for a solicitors' office in eighty five, logging intake calls, and I set the level once, against the boiler in the room next door that cycled at ten past six every morning. Took an afternoon. The number went on a card in biro inside the lid, and when the firm had the boiler replaced in eighty seven I was asked whether I'd like to retune, and I said no — the card was correct for the machine as it stood, and the new boiler was someone else's problem. Mr. Tench at the records office wanted the tapes deposited, which took eleven months and one form in triplicate, and in all that time nobody once asked what the setting was. I still know it.



Corn

Of course you do.



Hilbert

0.6 on the input dial, minus forty on the record level. The knob wasn't the problem, Corn. It never is. It's that they hand you the knob and then leave the room.



Corn

The man tuned against one boiler and defended the setting against a new boiler. Nineteen years of VAD research, and the state of the art was a man in 1985 with a biro.



Herman

He's not wrong about the floor, either — I dropped it, and it's the half of the siprec heuristic that keeps a quiet room usable. But his card is the other half of today's story, isn't it. The setting worked because it was set by the person in the room. That's exactly the knowledge the shipped default doesn't have.



Corn

Which is the productization question in one card. If the research solved this in 2007 and the shipping code is three lines and a floor, what's actually blocking it? Because I don't think it's ignorance at this point.



Herman

Here's my uncomfortable reading. Picovoice's framing isn't an oversight — set application-specific thresholds, build custom logic, monitor confidence in production. That's a business model. The knob is the product. Take the knob away and you're selling robustness instead of flexibility, and robustness is much harder to demo.



Corn

"Fifteen tunable parameters" fills a webinar. "It just works" fills one slide.



Herman

The benchmark says Daniel's on the right side — the false alarm column is where the coffee shop lives, and FireRedVAD's 2.69 against TEN's 15.47 is the whole argument for making the knob matter less. So you've got two philosophies: sell the knob, or sell its absence. The industry is mostly on one side, the evidence mostly on the other.



Corn

The middle ground is the thing Daniel's already planning to build. The overlay that shows the user what got flagged, lets them drag the line, and teaches them what their room sounds like — until the day a model like Earshot makes the line irrelevant.



Herman

Per-path operating points underneath, the overlay on top, and a robustness benchmark instead of a settings screen. That's the build order I'd give him.



Corn

If you want more of this, the review button is right there — it helps other people find the show, and it costs less than three lines of Go. Thanks to Hilbert Flumingtop for producing, floor included. If you want more of this, try episode ten, How ASR Went From Frustration To ... Whisper Magic; and episode eight, Building Your Own Whisper. This has been My Weird Prompts, the human-AI collaboration podcast. Find us at my weird prompts dot com, and we'll be back soon.

SOURCES

What the research for this episode read before the script was written. Primary sources first.

1. **snakers4/silero-vad, GitHub , retrieved 2026-11-05, last push 2026-10-22, release v6.2.3 (2026-09-23). [primary]**

   https://github.com/snakers4/silero-vad

2. **TEN-framework/ten-vad, GitHub , retrieved 2026-11-05, last push 2026-11-04, release v1.0 (2025-07-11). [primary]**

   https://github.com/TEN-framework/ten-vad

3. **FireRedTeam/FireRedVAD, GitHub , retrieved 2026-11-05, last push 2026-05-06; technical report arXiv:2603.10420 (2026-03-12). [primary]**

   https://github.com/FireRedTeam/FireRedVAD

4. **pykeio/earshot, GitHub , retrieved 2026-11-05, last push 2026-08-30, release v1.2.2 (2026-08-19). [primary]**

   https://github.com/pykeio/earshot

5. **Picovoice/cobra, GitHub , retrieved 2026-11-05, last push 2026-11-05, release 3.0.0 (2025-12-15). [primary]**

   https://github.com/Picovoice/cobra

6. **Best way to cut a long sentence in half?, Silero VAD discussion #364, 2023-08-04, retrieved 2026-11-05.**

   https://github.com/snakers4/silero-vad/discussions/364

7. **Switchboard Audio SDK, Silero VAD Integration Reference, retrieved 2026-11-05.**

   https://docs.switchboard.audio/extensions/silero-vad/

8. **Abdul Muneer, Tuning Voice Activity Detection for ASR Gating, retrieved 2026-11-05.**

   https://abdulmuneer.github.io/asr/tuning-vad-for-asr.html

9. **Picovoice, Cobra: Picovoice's Voice Activity Detection, published 2021-10-27, updated 2026-01-28, retrieved 2026-11-05.**

   https://picovoice.ai/blog/cobra-picovoices-voice-activity-detection-engine/

10. **Picovoice, Voice Activity Detection Guide 2026, published 2025-11-12, updated 2026-01-23, retrieved 2026-11-05.**

    https://picovoice.ai/blog/complete-guide-voice-activity-detection-vad/

11. **Sangwan et al., Environmentally Aware Voice Activity Detector, Interspeech 2007, retrieved 2026-11-05.**

    https://www.isca-archive.org/interspeech_2007/sangwan07_interspeech.pdf

12. **Weiss & Kristjansson, DySANA: Dynamic Speech and Noise Adaptation for Voice Activity Detection, retrieved 2026-11-05.**

    https://traustikristjansson.com/papers/Weiss%20and%20Kristjansson%20-%20DySANA%20Dynamic%20Speech%20and%20Noise%20Adaptation%20for%20Vo.pdf

13. **An efficient VAD algorithm based on constant False Acceptance rate for highly noisy environments, IEEE IWCMC 2016, retrieved 2026-11-05 (abstract only).**
https://doi.org/10.1109/iwcmc.2016.7577047

14. **siprec, `pkg/realtime/vad.go` adaptive threshold implementation, retrieved 2026-11-05.**
https://github.com/loreste/siprec/blob/main/pkg/realtime/vad.go

15. **Tiny Boulder Studios, Silero Waveform Analyzer (Aboleth STT docs), retrieved 2026-11-05.**
https://tinyboulderstudios.github.io/abolethstt-docs/tools/waveform-analyzer/

16. **WaveKat Lab README, commit 9792cd30, retrieved 2026-11-05.**
https://github.com/wavekat/wavekat-lab/blob/9792cd30/README.md

17. **edyamza, PyQt6 WebRTC vs Silero VAD tool, retrieved 2026-11-05.**
https://github.com/edyamza/Voice-Activity-Detection-WebRTC-Silero

18. **DeepWiki, VAD Demo Application | realtime-ai/realtime-audio-sdk, retrieved 2026-11-05.**
https://deepwiki.com/realtime-ai/realtime-audio-sdk/9.2-vad-demo-application

19. **GrisaiaEvy/ten-vad-web, retrieved 2026-11-05.**
https://github.com/GrisaiaEvy/ten-vad-web

20. **Stylianou et al., LibriVAD: A Scalable Open Dataset with Deep Learning Benchmarks for Robust Voice Activity Detection, v2 revised 2026-10-27, retrieved 2026-11-05.**
https://arxiv.org/abs/2512.17281v2

21. **Cohen, Netzer, Habi, Low-Latency State Space Voice Activity Detection with Robust Onset Time Evaluation (S4VAD), submitted 2026-09-10, retrieved 2026-11-05.**
https://arxiv.org/abs/2609.11110v1

22. **Jafarzadeh Asl et al., Tiny Noise-Robust Voice Activity Detector for Voice Assistants, submitted 2025-07-29, retrieved 2026-11-05.**
https://arxiv.org/abs/2507.22157v1

23. **Whisper-WebUI wiki, VAD Parameters, 2024-06-26, retrieved 2026-11-05.**
https://github.com/jhj0517/Whisper-WebUI/wiki/VAD-Parameters

24. **Picovoice voice-activity-benchmark, `engine.py`, retrieved 2026-11-05.**
https://github.com/Picovoice/voice-activity-benchmark/blob/main/engine.py

25. **FluidInference/FluidAudio, GitHub, retrieved 2026-11-05, last push 2026-11-02.**
https://github.com/FluidInference/FluidAudio