Episode Book

EPISODE #5995

NVIDIA ASR: Parakeet, Canary, and Nemotron Explained

PUBLISHED	15 October 2026 at 13:30 UTC
DURATION	32:05
LISTEN	myweirdprompts.com/episode/nvidia-asr-parakeet-canary-nemotron-17221/
HOSTS	Corn, Herman, Dorothy, Hilbert
SCRIPT MODEL	deepseek-flash
VOICES	chatterbox-regular (chatterbox-local)
PIPELINE	V5.4
BOOK GENERATED	19 October 2026 at 17:42 UTC

SYNOPSIS

NVIDIA's speech recognition lineup isn't one model — it's a portfolio, and the names don't tell you what you're getting. Parakeet, Canary, and Nemotron Speech are brands, not architectures; underneath, nearly all of them share the same FastConformer encoder. This episode walks the whole family tree one model at a time: why Parakeet v3 is not a straight upgrade over v2 (it's slightly worse on English), how the Parakeet unified model makes the streaming tax visible in a single page of numbers, why Canary's forty-second clip limit catches people out, and where the non-commercial license hides. Plus a look at a new fine-tune of Parakeet v2 that handles Hebrew words mixed into English sentences, shipped with on-device GGUF builds. Less benchmark, more map — the goal is knowing which model fits which workflow.

TRANSCRIPT



Corn

NVIDIA's speech recognition lineup is a maze. That's Daniel's starting point, and he's right. Whisper is one family with sizes, tiny through large, so "which Whisper" is a single question. NVIDIA is a portfolio. Parakeet, Canary, Nemotron Speech, all of it, plus the pieces around the edges that do adjacent jobs, and within Parakeet alone there's v2, v3, a unified model, a hundred-and-ten-million-parameter one, single-language variants. Daniel wants one clean pass through the whole family tree, one by one, and then the part he actually cares about, which aren't the architectures but the workflows. When do you reach for streaming, when do you reach for the offline long-form model, batch transcription versus real-time voice typing. That's the shape of the episode.



Herman

And the framing he gave it is the right one. A dummies guide, in the good sense. Not a benchmark episode, a map.



Corn

One thing at the top, before we get into the guide. This morning Daniel published a fine-tune of Parakeet v2 for English with Hebrew code-switching. Base model is the English-only v2. He taught it to handle Hebrew words mixed into English sentences, written in one fixed romanized spelling, because v2's tokenizer has no Hebrew characters at all, so the Hebrew comes out transliterated. Casing and punctuation preserved. Roughly two hours of his own read-aloud recordings plus LibriSpeech, trained for a single speaker, so it's a proof of concept rather than a product.



Herman

And the example he gives is a good one. Stock v2 hears "makolet," the corner grocery, as "Micollette." The fine-tune writes "makolet." That's the whole point in one word.



Corn

He also shipped GGUF builds for on-device use. Q8_0 at seven hundred and thirty megabytes, which he says matched the full-precision output word for word on five held-out clips. There's a smaller four-bit one at four hundred and seventy-five megabytes that got one word wrong.



Herman

Noted, and we'll come back to why that file format matters when we get to the NeMo section. So. Where do we start the guide?



Corn

With the part that actually confuses people. Whisper gives you one dial. NVIDIA gives you five, and nobody labels them.



Herman

Name the axes, then, so we have a spine. Axis one, which family, Parakeet or Canary or Nemotron Speech. Axis two, which series inside that family, v2 or v3 or flash. Axis three, what kind of decoder is doing the writing. Axis four, streaming or offline. Axis five, the licence, which bites people commercially more often than any of the others.



Corn

That's the grid. And the first trap is the names themselves. Parakeet, Canary, Nemotron, those are brands. They aren't architectures. Underneath, every current model shares the same FastConformer encoder, so the branding is doing work the engineering doesn't.



Herman

Which is exactly backwards from how people read it. They see three families and assume three different machines. It's one machine wearing three name badges.



Corn

And the second trap, the same model carries different names depending on where you buy it. The streaming English Nemotron model on NGC is sold as "Nemotron 3 ASR." On Hugging Face it's nemotron-speech-streaming-en-0.6b. Same weights.



Herman

So with those axes in hand, let's start with the family Daniel already knows. Parakeet. And the v2 versus v3 question, which trips up almost everyone.



Corn

This is the one I see people get wrong constantly. They assume v3 is v2 plus a version number. Go on.



Herman

v2 is parakeet-tdt-0.6b-v2. Six hundred million parameters, released May first, 2025, CC-BY-4.0, English only. Six point zero five percent average WER on the Open ASR Leaderboard. RTFx around three thousand three hundred and eighty. Handles up to twenty-four minutes of audio in a single pass. And it emits its own punctuation, its own capitals, and word, segment and character timestamps, no second pass needed.



Corn

Which for batch work is the whole story. You feed it a recording, you get back a punctuated, timestamped transcript.



Herman

Now v3, released August fourteenth, 2025. Same parameter count, same licence. But it's multilingual. Twenty-five European languages with automatic language detection built in. And on English it scores six point three four percent WER.



Corn

Worse than v2.



Herman

Slightly worse. Six point three four against six point zero five. So v3 is not a straight upgrade. The axis is English-only versus multilingual. If you only ever transcribe English, v2 is still the better model. That's the single most useful correction in this episode.



Corn

People have been waiting for v3 like it's a firmware update and quietly degrading their own English transcriptions.



Herman

And what v3 buys you is the languages. Hebrew isn't in that list of twenty-five, incidentally, which is why Daniel's fine-tune had to exist at all.



Corn

So which one do I actually download, if all I have is English meetings?



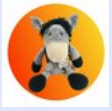
Herman

v2. Full stop. It's more accurate on English, it's the same size, and it punctuates.



Corn

And the long-audio trick on v3, the one you mentioned?



Herman

v3 handles twenty-four minutes with full attention, or up to three hours if you switch it to local attention. So the multilingual model is actually the one with the longer ceiling, which is a detail most people miss.



Corn

What else is on the Parakeet shelf?



Herman

The hundred-and-ten-million one, parakeet-tdt_ctc-110m. About a hundred and fourteen million parameters, English, punctuation and capitals, up to twenty minutes, seven point four nine percent WER, and RTFx around five thousand three hundred. That's the weak-device pick. Runs on almost nothing.



Corn

Phones, Raspberry Pis, that class of hardware.



Herman

Then there's the older generation still floating around. parakeet-ctc-1.1b and parakeet-tdt-1.1b, both English, both lowercase with no punctuation. The tdt one scores seven point zero two percent. They're superseded, but they're still in every tutorial from two years ago.



Corn

Lowercase and no punctuation means you're post-processing, right? Running it through something to restore the capitals.



Herman

Or accepting a wall of lowercase text. Which is fine if you're feeding it into a search index. Bad if a human has to read it.



Corn

And the interesting one. The unified model.



Herman

parakeet-unified-en-0.6b, April seventh this year. One set of English weights that does both offline and streaming. That's the novel part. Usually you pick a model for one job or the other, and here it's the same weights, you're just choosing a latency point.



Corn

Give me the ladder. This is the one to slow down on.



Herman

Offline it's five point nine one percent WER. Set it to one point one two seconds of latency, six point two nine. At zero point one six seconds, eight point four four. At zero point zero eight seconds, fifteen point six three.



Corn

That's not a degradation, that's a different product.



Herman

At eighty milliseconds of latency you're paying more than double the error rate of the offline mode. Same weights, same model, you just asked it to make up its mind before it had heard the sentence.



Corn

So the streaming tax is visible on one page, in one model, with no other variables.



Herman

Which is why it's worth more than any architecture diagram. You can see exactly what speed costs you.



Corn

And there's no Parakeet v4, before anyone asks.



Herman

There is not. NVIDIA's newer releases carry the Nemotron name instead. So if you're waiting for v4, you're waiting for something that already shipped under a different label.



Corn

That's a naming decision that has cost real people real hours.



Herman

It has. Now Canary, which is a different animal, and not just by brand.



Corn

Different how?



Herman

Canary is an encoder-decoder model, which NVIDIA's own documentation abbreviates AED. And the practical difference is that you steer it with task tokens. You tell it the source language, the target language, whether to transcribe or translate, whether to punctuate. It's a model you give instructions to.



Corn

So it's the one that can translate.



Herman

That's the headline capability. Parakeet transcribes. Canary transcribes or translates, and toggles punctuation on demand. The cost is speed, and a short-clip design we'll get to.



Corn

Walk me through the lineup.



Herman

canary-1b, the original. English, German, French, Spanish. Transcription plus translation. But CC-BY-NC-4.0, which means non-commercial only. That's the licence outlier in the whole family. If you're building anything you intend to sell, that one's off the table.



Corn

The one model in the portfolio you can't put in a product.



Herman

canary-1b-flash came next. Eight hundred and eighty-three million parameters, same four languages, and crucially CC-BY-4.0, so commercial use is fine. RTFx around a thousand and forty-six. Timestamps are labelled experimental, so don't build a subtitle pipeline on them without checking. And it's built for clips under forty seconds.



Corn

Which is the constraint that catches everyone.



Herman

Longer audio needs chunking, and you have to do the chunking yourself. Segment the recording, feed it in pieces, stitch the transcripts. That's on you.



Corn

So someone points this at a ninety-minute meeting recording and gets forty seconds of it.



Herman

Or it fails in a way that takes an afternoon to diagnose. Then there's canary-180m-flash. A hundred and eighty-two million parameters, same tasks, same licence, and NVIDIA lists it for Jetson-class hardware. That's the embedded tier.



Corn

And then canary-1b-v2.



Herman

August fourteenth, 2025, same day as Parakeet v3, and it transcribes the same twenty-five European languages. About nine hundred and seventy-eight million parameters. It translates English to and from twenty-four languages, and here's the difference from flash, it chunks long audio automatically. You hand it a long file, it handles the segmentation itself.



Corn

So the version two of Canary solved the forty-second problem.



Herman

For that model, yes. What it didn't solve is speed. RTFx is about seven hundred and forty-nine, against Parakeet v2's three thousand three hundred and eighty. It's roughly four and a half times slower.



Corn

And then the headline model.



Herman

canary-qwen-2.5b. This is clever. NVIDIA took canary-1b-flash and bolted it to Qwen3-1.7B, a language model, and they call that design a SALM. English only, CC-BY-4.0, five point six three percent WER, RTFx around four hundred and eighteen. It was number one on the Open ASR Leaderboard as of October last year.



Corn

Best in class on accuracy, and the slowest thing in the catalogue.



Herman

Correct. And two caveats worth stating plainly. It was trained on clips of forty seconds or less, so it inherits flash's short-clip limitation. And its LLM mode operates on the transcript, not on the audio. So the language model is sitting after transcription, rewriting what came out.



Corn

Which is a different thing from a model that listens.



Herman

Substantially different. It can clean up a transcript beautifully. It did not hear the room.



Corn

So Canary's whole value proposition is translation and task control, and you pay for both in throughput.



Herman

Every time. If all you need is English transcription, Canary is strictly more expensive than Parakeet for the same job.



Corn

And that's the point I want to sit on for a second, before we get to streaming. These aren't quality tiers. Nobody's ranking them best to worst. They're workflow shapes. A model built for forty-second clips isn't worse than one built for three hours. It's built for a different job, and the only mistake is using it for the wrong one.



Herman

Which is exactly the framing Daniel asked for. Not the architecture, the workflow.



Corn

So streaming. Nemotron Speech.



Herman

The newest branding, and the streaming family. nemotron-speech-streaming-en-0.6b, March this year, English, under the NVIDIA Open Model License. On NGC it's sold as "Nemotron 3 ASR," which is trap number two from the intro.



Corn

And under the hood?



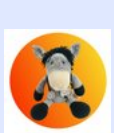
Herman

A cache-aware FastConformer with an RNN-T decoder. What cache-aware means, practically, is each audio frame is processed once and then cached, rather than re-processing overlapping windows the way older streaming setups did.



Corn

Which is the difference between a model that can run while someone talks and one that can't.



Herman

It's the whole reason streaming is feasible at this size. And its chunk sizes tell the same story as the unified model's latency ladder. Eighty milliseconds of chunk gives you eight point four three percent WER. A hundred and sixty, seven point six seven. Five hundred and sixty, seven point zero seven. And one thousand one hundred and twenty, six point nine three.



Corn

More delay, more accuracy. Every time, same curve.



Herman

Monotonic. There's no free lunch in that table anywhere.



Corn

Then there's the three point five version.



Herman

nemotron-3.5-asr-streaming-0.6b, June fourth this year. Forty language-locales, automatic language detection, punctuation and capitals, same eighty to eleven-hundred-and-twenty millisecond chunk range. And the languages come in tiers.



Corn

Tiers meaning what, exactly?



Herman

Meaning how well it handles them out of the box. And Hebrew sits in the adaptation-ready tier. Which means it needs fine-tuning before it transcribes well.



Corn

So that's the same problem Daniel solved this morning, described in NVIDIA's own documentation.



Herman

By a different route, yes. NVIDIA is saying these languages are supported, but you'll need to adapt the model. Daniel took an offline English model and taught it code-switching instead. Same destination, different road.



Corn

And the licence on three point five is a mess worth flagging.



Herman

It is. OpenMDW-1.1 on Hugging Face, NVIDIA Community Model License on NGC. Same model, two different licence names depending which door you came in.



Corn

What does NVIDIA say about it?



Herman

That future Nemotron releases will use OpenMDW-1.1. So the direction of travel is one licence, but right now you have to check which page you downloaded from.



Corn

Which is not a thing anyone should have to do.

D

Dorothy

Hello? Corn, sweetheart, is that you? I think I've rung the wrong number, the machine said something about a podcast.



Corn

Mum. Mum, we're... yes, we're recording.

D

Dorothy

Oh, that's lovely. Have you eaten? Because I left a casserole at your door on Tuesday and I don't think anyone brought it in.



Corn

We'll get the casserole, Mum.

D

Dorothy

And are you sleeping enough? Because your brother always says you sleep too much, but I think it's the opposite, I think you're up all night with those machines.



Herman

Hello, Dorothy.

D

Dorothy

Oh, Herman, darling, hello. He's so good, isn't he, Corn? You should keep him.



Corn

I'm trying to, Mum. We'll call you back.

D

Dorothy

Alright, my love. Eat something that isn't leaves.



Corn

Right. Where were we.



Herman

The supporting cast.



Corn

Sortformer. One line.



Herman

NVIDIA's diarization model, which works out who spoke when. The streaming four-speaker version two point one is a hundred and seventeen million parameters. Nemotron-3-Diarization from September this year handles up to eight speakers.



Corn

MarbleNet.



Herman

The voice-activity detector. Spotting where speech is at all. About ninety-one and a half thousand parameters. It's the smallest thing in the catalogue by an order of magnitude.



Corn

Granary.



Herman

The open dataset. About a million hours across twenty-five European languages, CC-BY-4.0. It's what Parakeet v3 and Canary v2 were trained on. That's why those two shipped the same day with the same language list.



Corn

Same food.



Herman

And then there's the one that surprises people, multitalker-parakeet-streaming-0.6b-v1. It transcribes each speaker separately, including when people talk over each other, using Sortformer's diarization output as its input.



Corn

So it's not a new acoustic model, it's Parakeet plus the diarizer wired together.



Herman

Functionally, yes. That's a system, not a model, even though it ships as one.



Corn

Which is the pattern across all of this. The models are components. The products are combinations.



Herman

And one more shelf, the one that generates confused forum posts. QuartzNet, Jasper, Citrinet, Conformer, Squeezeformer. All still in the NeMo docs. All outdated.



Corn

Legacy models people find in a tutorial from 2021 and train for a week.



Herman

FastConformer is the encoder every current model shares. If it isn't FastConformer underneath, it's history.



Corn

That's the model shelf. But Daniel asked a question underneath all of it. What is NeMo, and what's actually in a .nemo file? Because that's the thing that makes the shelf make sense.



Herman

And it's the one that unlocks everything else. NeMo is not a model. It's NVIDIA's open-source toolkit for training and running speech models. Parakeet, Canary, Nemotron Speech, all of them are built with it. NeMo is the workshop. The models are what comes out of it.



Corn

So when someone says "I'm running NeMo," they haven't told you anything about what they're running.



Herman

Nothing at all. It's like saying you're running Python. True, and useless.



Corn

Then what's a .nemo file?



Herman

A tar archive. Inside it: model_config.yaml, which is the architecture and settings, model_weights.ckpt, which is the actual weights, and the tokenizer and vocabulary files. That's the whole thing. Load it with NeMo in Python, or use it as the starting point for a fine-tune.



Corn

And it's not the same as a Lightning checkpoint.



Herman

Different animal. A Lightning .ckpt also carries optimiser state, because it's designed for resuming training mid-run. So if you download two files of the same model and one is three times the size, that's why. The big one has the optimiser's memory in it, and you don't need any of that to run inference.



Corn

This is the moment I'd bring up something we said a while back, on the episode about inference formats. Safetensors and the rest of it. Same lesson, different file.



Herman

Same lesson exactly. The container tells you what it's for. A .nemo is for loading and fine-tuning. A Lightning checkpoint is for resuming. An ONNX or GGUF file is for running somewhere that doesn't have Python and a GPU.



Corn

And that's the device story.



Herman

It is. People take .nemo files and convert them. ONNX if you want sherpa-onnx. GGUF if you want transcribe.cpp, which is what Daniel did this morning for Android and desktop. On Apple hardware, CoreML through FluidAudio or MLX through parakeet-mlx.



Corn

Which is why his fine-tune ships as a seven-hundred-and-thirty-megabyte GGUF rather than a two-point-four-seven-gigabyte .nemo file. Same model, one is for a phone.



Herman

And NVIDIA themselves have published NeMo-Speech.cpp, their own GGUF runtime with Riva-compatible gRPC, Apache-2.0. So they've blessed the format.



Corn

And Hugging Face Transformers runs Nemotron three point five natively now.



Herman

Version five point thirteen and later, yes. So there are three or four ways to run the same weights, and the reason it feels like a maze is that all of them are documented, none of them is labelled as the default.



Corn

Okay. Decoder flavours. Plain English, what do they mean to me.



Herman

CTC first. Fastest and simplest. It reads the audio once. Older CTC models output lowercase only, and critically, it doesn't invent text. It writes what the acoustics most likely map to and stops.



Corn

So it's the honest one.



Herman

It's the honest one. RNN-T, a transducer, writes text piece by piece as audio arrives. That's what makes it the natural fit for streaming. TDT is a transducer that predicts how long each token will last and skips blank frames, which buys speed and good timestamps. That's what Parakeet tdt uses.



Corn

And the hybrid.



Herman

tdt_ctc models carry two decoder heads, and you switch between them. One for speed, one for accuracy, same weights.



Corn

And AED, which is Canary.



Herman

AED writes text the way a language model does. That's what allows the prompts and the translation. And it's also why it's slower, why it's built for short clips, and why it can repeat itself or make things up the way Whisper does. Same family of behaviour.



Corn

So the decoder choice is really a choice about whether the model is allowed to be creative.



Herman

That's a fair way to put it. CTC and the transducers can't embellish. AED can.



Corn

Now Daniel's actual question. Long-form versus short-form, streaming versus non-streaming, not as architecture but as workflow. Give me the picks.



Herman

Batch transcription of long English recordings. parakeet-tdt-0.6b-v2.



Corn

Because of the WER gap and the timestamps.



Herman

And the twenty-four-minute ceiling in one pass, with punctuation already in. For other European languages, v3, which goes to three hours with local attention. That's your meeting recorder, your podcast archive, your lecture series.



Corn

Real-time English voice typing.



Herman

nemotron-speech-streaming-en for a dedicated streaming model, or parakeet-unified-en if you want the same weights doing both jobs. The unified one is the interesting case, because you're choosing a latency point, not a model. Same download either way.



Corn

Real-time voice typing in a non-English language.



Herman

nemotron-3.5-asr-streaming. And check the tier, because adaptation-ready means fine-tune first. That's the Hebrew case, and it's why Daniel's fine-tune exists.



Corn

And here's the one people forget. Push-to-talk dictation.



Herman

Right. You hold a button, you speak, you release, and then it transcribes the whole clip at once. There's no real-time requirement at all. And for that, offline Parakeet v2 is more accurate, simpler to run, and you're not paying the latency tax for a real-time experience you never needed.



Corn

So people reach for streaming because it sounds more advanced, and get a worse transcript for no reason.



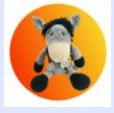
Herman

Frequently. Streaming is the harder engineering problem. It is not the better answer for every question.



Corn

Other picks, quick.



Herman

Maximum English accuracy where speed doesn't matter, canary-qwen-2.5b. Translation, canary-1b-v2. Weak devices, tdt_ctc-110m or canary-180m-flash. Who spoke when, add Sortformer. And fine-tuning, any .nemo checkpoint through NeMo.



Corn

Which is what Daniel did, incidentally. Took a v2 checkpoint, ran it through NeMo, came out with a code-switching model. That's the pipeline we talked about on the fine-tuning episode, except this time the languages are the interesting part.



Herman

And that's the whole loop closed. NVIDIA says Hebrew is adaptation-ready. Daniel adapted it.



Corn

Deployment. Riva and NIM.



Herman

NVIDIA's packaged server deployment. Containers and gRPC. And the current ASR NIM serves Parakeet, Nemotron ASR Streaming, Canary, and Whisper Large v3.



Corn

Whisper in NVIDIA's own serving stack.



Herman

Which tells you they know people are mixed. And it separates two questions that the model cards make look like one. Which model, and how do I serve it. Those are independent decisions, and the NIM layer is what makes them independent.



Corn

And one number people misread. RTFx.



Herman

Constantly. It measures speed. Times faster than real time. It is not accuracy and it is not latency. A model at three thousand three hundred and eighty and a model at four hundred and eighteen are both fast enough for most batch jobs. The difference is throughput at scale. Whether your transcription finishes at all is not in question for either.



Corn

If you're transcribing one meeting a day, the difference between those two numbers is invisible.



Herman

Completely invisible. If you're transcribing ten thousand hours, it's the entire business case.



Corn

Which closes the arc on why this feels confusing in the first place.



Herman

It does. The reason the lineup reads as a maze is that it's a portfolio organised around deployment targets. Jetson, server, streaming, batch, multilingual, English-only. The naming reflects product decisions, not technical ones. Whisper's simplicity comes from OpenAI having one deployment target in mind. One machine, one size dial, done.



Corn

Once you see the axes, it stops being a maze.



Herman

It becomes a menu. Which is all Daniel was asking for.



Hilbert

You keep saying "v3." That's not the name.



Hilbert

It's parakeet-tdt-0.6b-v3. Every time one of you says "Parakeet v3" I lose a year.



Herman

That's what everyone calls it.



Hilbert

Everyone calls it that because the model card leads with the family name and buries the architecture. You said it yourselves an hour ago. Parakeet is a brand. It's a brand with an architecture hiding under it, and the file names are the only place the architecture survives. I've been downloading these things since the first one, and the file name has never once lied to me. The friendly name does. Every time.



Corn

That's a strong position.



Hilbert

It's not a position, it's a filing system. And I'll tell you where it's worse than you said. You gave the v2 versus v3 thing sixty seconds and moved on. That's underselling it. My nephew works in a clinic in Haifa, and they had a transcription system built on the assumption that v3 was the upgrade. Because the number went up. They bought it, they deployed it, and their English accuracy went backwards. Six point three four against six point zero five. Nobody caught it for four months, because nobody had a reason to run the two side by side. The number went up. The number going up means it's better, that's what a number does.



Herman

The version number cost them four months.



Hilbert

The version number cost them four months and a support contract. And when they asked, the answer was that v3 is multilingual, English was never the target. Which is true. It's just not what the number says.



Corn

What would you have named them?



Hilbert

I'd have named them after the job. English-tdt-0.6b and Europe-tdt-0.6b. Then nobody rings me about it.



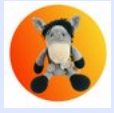
Corn

The Nemotron rebrand. You saw that coming?



Hilbert

I said it before they did it.



Herman

You said it to who?



Hilbert

I said it at a desk, to a man named Farouk, who was eating a sandwich, and he agreed with me. That's two witnesses.



Corn

One of whom was eating.



Hilbert

Farouk counts. And then three point five comes out and now the same weights have got two licence names depending on the website. OpenMDW on one, Community Model License on the other. I spent a Tuesday on that. A whole Tuesday, because a colleague needed to know if he could ship it, and the honest answer was "which URL did you get it from."



Herman

That one we did flag.



Hilbert

You flagged it. I lived it.



Corn

You've been running v2 for something specific, you said.



Hilbert

Batch job. Overnight. Every night, since May last year. Two hundred and forty hours of audio a week.



Herman

Two hundred and forty hours a week is more than there are hours in a week.



Hilbert

Not simultaneously. It's a stack. It runs at three thousand three hundred and eighty times real time, so it eats a week of audio before I finish my coffee. But the point is the file name. I have it in a cron job, and it's the full name, parakeet-tdt-0.6b-v2, because when I set it up in May I read the actual file name and put the actual file name in the script, and it has therefore never broken. Sixteen months. The people who wrote "Parakeet v2" in their configs have all had a bad afternoon by now.



Corn

What are you transcribing, exactly?



Hilbert

Radio. Longwave. There's a gentleman in Ushuaia who still broadcasts, and I archive him. It's about nine hundred hours a month and none of it is in English.



Herman

But v2 is English only.



Hilbert

It is. The transcripts are terrible. That's not the point of the archive.



Corn

Right.



Hilbert

The point is the file name. And the hardware. It runs on three secondhand workstations under my mother's stairs, and she has never once mentioned them.



Corn

She's never mentioned three computers under her stairs.



Hilbert

She thinks they're a boiler. I'm not going to correct her. The point I came out here to make is that the clinic should have used the file name. That's it. That's all of it. Everything else you said was fine.



Herman

NVIDIA's portfolio is organised around deployment targets, and the names reflect product decisions, not technical ones. Which leaves an open question. If that's the logic, does the Nemotron branding consolidate the lineup or fragment it further?



Corn

Consolidate the licences, probably. That's what OpenMDW is for, and NVIDIA saying future releases will use it is a real signal. But it doesn't consolidate the names. If anything, moving every new release under one brand makes the naming worse, because now everything is Nemotron-something and you can't tell a streaming model from a batch model from a diarizer without reading the file name.



Herman

The file name being the thing Hilbert actually told us to trust.



Corn

The file name being the thing Hilbert told us to trust. So here's the question for anyone listening. If the names are one brand and the file names are the only honest part, what happens when the next generation lands and the old file names stop existing. Whether



Corn (continued)

you keep your old config or rewrite it is going to be a real decision, and it's worth thinking about before it arrives, not after.



Herman

That's the thread to watch. Not the leaderboard, the file names.



Corn

Thanks to our producer, Hilbert Flumingtop, who was right about the version number and will not let us forget it.



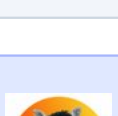
Herman

He never does.



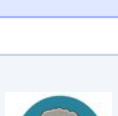
Corn

For more along these lines, there's episode three, Safetensors or something else; episode five, Fine-Tuning ASR For Maximal Usability; and episode two, Local STT For AMD GPU Owners. This has been My Weird Prompts. If you're enjoying the show, a review helps more than you'd think. And if you've got a question about any of this, send us your own prompt on Telegram at t dot me slash MWP listener bot.



Herman

We'll be back soon.



Corn

See you tomorrow.