Episode Book

EPISODE #6444

Teaching Speech Models to Collapse Spelled Words

PUBLISHED	4 November 2026 at 09:03 UTC
DURATION	21:48
LISTEN	myweirdprompts.com/episode/speech-models-spelled-letters-62643/
HOSTS	Corn, Herman, Larry, Hilbert
SCRIPT MODEL	z-ai/glm-5.3
VOICES	chatterbox-regular (chatterbox-local)
PIPELINE	V5.26
SOURCES	17 cited by the research (listed at the end)
BOOK GENERATED	5 November 2026 at 07:03 UTC

SYNOPSIS

Teaching an automated speech recognition model new vocabulary is a closed-set problem, but teaching it formatting behaviors is an open-ended mapping rule. Fine-tuning a non-autoregressive FastConformer to convert individually spelled letters into unified words requires learning conditional structures rather than just memorizing vocabulary. A look at inverse text normalization, contextual biasing limits, and grammar transducers reveals why behavioral training requires distinct design choices over raw volume.

TRANSCRIPT



Corn

Daniel's Parakeet fine-tune has become something of a daily serial for this show, and today's instalment is a turning point. He says the word-list phase is basically done. Code-switching, tech vocabulary, the words he kept seeing come out wrong, once those went into his dictionary the problem mostly dissolved. He suspects medical dictation is the same shape, a finite set of drugs and diseases.



Herman

Poppleberry here, and I want to say up front that "finite set" is doing more work in that sentence than Daniel probably intends. But go on.



Corn

The new frontier is behaviours. He'll say "my name is spelled R, O, S, E, H, I, L, L" and he wants "Rosehill" out the other end, not eight letters with commas. His question is the generalization question. How many examples of spelling-out does the model need before it holds on a word that was never in the training sentences, like "L, A, M, P"? And what's the equivalent of a vocabulary list when the thing you're listing isn't words?



Herman

The moment he said that, I thought, this is the episode where the project stops being data entry and starts being teaching.



Corn

Which is either exciting or a warning, depending on how the last few teaching attempts went.



Herman

Let's put the distinction on the table properly, because everything after it hangs off one line. A word list is a closed set. You can write it down, count it, prune it, check a candidate against it. A behaviour is a mapping rule, and the set of things it can map is unbounded. Nobody can enumerate every word a person might spell out loud, because it's any word in the language, plus names, plus loanwords, plus stuff he makes up on a Tuesday.



Corn

So the "vocabulary list for a behaviour" is a category error hiding as a reasonable request.



Herman

It's a reasonable request that has a non-obvious answer. The nearest real thing to it is a grammar, and we'll get there. The other piece of context listeners need is what Parakeet actually is, because it decides where a behaviour can live. The model he's fine-tuning is a six hundred and twenty-seven million parameter FastConformer with a TDT decoder. Twenty-five European languages, automatic punctuation and capitalisation, and the crucial bit, no prompting required, because there's no prompt at all. It's not an autoregressive LLM with a free-text channel you can slip instructions into. Whisper users have that channel, flawed as it is. Parakeet users have weights and nothing else.



Corn

So Daniel's behaviour has to live in the weights or it lives nowhere.



Herman

Or downstream, as a post-processing pass. But that's a different architecture decision and he's clearly chosen the weights.



Corn

Worth naming the failure precisely, too, because it's easy to misdiagnose. When he spells R, O, S, E, H, I, L, L, the model hears the letters fine. The audio isn't the problem. The model transcribes letters because that's literally what was said.



Herman

Right, the failure is in the written-form mapping. What he's really asking the model to do is inverse text normalisation with a context cue. Take the spoken form, decide on the written form. And the model's default answer, letters, isn't a bug. It's the statistically correct answer for its training distribution, because almost nobody spells words out in a dictation corpus, and when letters appear spoken, the overwhelmingly common written rendering is, letters.



Corn

The model is being literal-minded at scale.



Herman

Trained literal-minded. There's a lovely number for how much it takes to install the default. NVIDIA's Universal-2-TF, the text-formatting model behind a lot of modern dictation output, has an ITN-focused stage that was fine-tuned for two thousand steps on two billion words of simulated inverse-text-normalisation data. Two billion words, to teach one layer of a pipeline how written forms work. That's the gravity Daniel is fine-tuning against.



Corn

Two billion words saying "letters stay letters." My kind of consensus, honestly. Sloth politics.



Herman

Now compare what the word-list layer looks like when it's done properly, because this is where his instinct that vocabulary is "mostly solved" comes from, and it's correct. Contextual biasing research treats the bias list exactly as a closed set. There's a framework out there, BR-ASR, that handles bias lists up to two hundred thousand entries, and at that scale it degrades only fractionally, because it prunes about ninety-nine point nine nine percent of candidates per query in around twenty milliseconds. With two thousand bias words on LibriSpeech it cut biased error rates by nearly half.



Corn

Two hundred thousand entries and it still answers in twenty milliseconds. That's a phone book with a great memory.



Herman

That's the point of a closed set. Finite, retrievable, prunable. And it's why his medical dictation hunch is right, the drug names and the diseases are a big list but a list. The behaviour layer is a different animal, and here's where the list starts to look like a trap. Somebody ran a practical experiment on this with Whisper, a fifteen-term jargon list, ninety percent of the technical terms came out right. Sounds great.



Corn

I'm waiting for the comma.



Herman

"Laura will call you back" came out as "LoRA will call you back." Six clips out of six. "Cash the check" became "cache the check." Also six out of six. The list didn't make the model hear anything different. It chose between spellings the audio already supported. That's all a list can ever do. It can't add structure nobody spoke.



Corn

So the decoys are the tax. You buy "LoRA" for your ML vocabulary and "Laura" comes with it, at a hundred percent failure rate.

L

Larry

Folks, let me tell you about a satisfied customer of mine, a lovely lady by the name of Margaret, no wait, it was Deborah, Deborah from a town I will not name because she is shy, and Deborah had the same problem you have been hearing about all episode, her speech engine kept spelling out R, O, S, E, H, I, L, L instead of just writing Rosehill, and she was at her wit's end, so she bought Behaviour Vocabulary, the only product on the market that gives you a written vocabulary list for a behaviour, a laminated card with every possible behaviour printed on it, all of them, the entire open-ended set, so you can count them and prune them like a word list, and Deborah says it worked instantly, she now spells out whatever she likes and the machine writes perfect words every time, she dictated her entire memoir by spelling it out loud, letter by letter, and it came out as one beautiful word.



Herman

That's the exact shape of the risk for Daniel's spelling behaviour too, just inverted. His behaviour has to decide when to collapse letters into a word. The over-firing version turns dictated initials into words. An acronym in a serial number becomes something in the dictionary. A behaviour that fires too eagerly is worse than no behaviour, because it corrupts the correct output, it doesn't just miss the target.



Corn

Which is why his question "how many examples" is quietly the wrong shape of question. A word list has a size. What's the size of "collapse letters when they're being spelled"?



Herman

The nearest thing to a canonical answer is the grammar people. Google's text normalisation covering grammars have a spelled component, a finite-state transducer that maps any character sequence to letter-by-letter spoken form, and back, for tagged input. It generalizes to any letter string by construction. That's what a "vocabulary list for a behaviour" looks like when it's done by rule. Not a list. A rulebook.



Corn

So Daniel's spelling behaviour already exists, as a declarative rule in a transducer from a previous era of the field.



Herman

And the field walked away from it, which is the part I find interesting. The streaming ITN work, the last two years of it, keeps making the same complaint, that the hybrid systems, neural tagging stitched to hand-written transducer rules, don't scale across domains and languages. The new papers are all end-to-end, read-tag-write decoding policies that learn the formatting instead of being told.



Corn

The rulebook generalizes perfectly and doesn't scale. The neural version scales perfectly and might not generalize at all. Daniel's fine-tune sits right on the seam between them.



Herman

So let's get to his actual number. Nobody has measured spelled-letters-to-word specifically, I looked, and the honest position is that the closest measured analogue is punctuation and truecasing, because those are formatting behaviours rather than words. There's a Johns Hopkins study that asked almost his exact question in that domain. Fifty conversational documents gets you roughly seventy-three percent truecasing accuracy, and the steep part of the curve is between fifty and a hundred documents. Tens of documents, not thousands.



Corn

Fifty dictations to learn "capitalise the first word." That's... surprisingly few.



Herman

The other formatting numbers cluster the same way. One group showed a joint punctuated ASR system works with as little as five percent punctuated training data, at a measurable but moderate cost. Myrtle's production model found that training on punctuated, cased text only a quarter of the time was actually optimal. And few-shot domain adaptation on a universal ASR model, about twenty percent relative error reduction from ten minutes of audio.



Corn

Ten minutes. I've spent longer than that deciding whether to nap.



Herman

You've spent longer than that mid-nap, deciding whether to turn over.



Corn

Unfair. Accurate, but unfair.



Herman

So the answer to "how many examples" looks like, tens, not thousands, and Daniel's probably already past the threshold on volume alone. But there's a counterweight and it's the heaviest thing in this whole literature. The FSA-GRPO work found that most auditory models are never explicitly trained to perform inference in a demonstration-conditioned format. They aren't taught to use examples as examples. Their fix was a reinforcement-learning post-training stage on two thousand utterances, and once the model had been taught to use demonstrations, it got a fifty-four percent relative error reduction on children's speech with zero child training data.



Corn

Two thousand utterances taught it to use the other utterances.



Herman

That's the sentence, yes. The lesson that transfers to Daniel is uncomfortable. A handful of clean examples may do nothing at all, not because they're bad examples, but because the model was never taught that a pattern in its input is something to extract rather than something to transcribe.



Corn

Hold on. Before we run with that, there's a version of the distinction I want to check, because it decides whether Daniel's checkpoints can ever do this. Parakeet isn't being shown examples in a prompt. The fine-tuning is changing the weights. Is "demonstration-aware" even the right frame for a transducer, or is that an LLM finding he's borrowing?



Herman

That's a real question and I think the honest answer is, it's the right frame with a translation. The transducer can't use in-context examples because it has no context channel. What it can do is learn a conditional rule in the weights, "when the acoustic and lexical pattern around these letters signals spelling, collapse them." The fine-tuning either teaches that structure or it doesn't. If it doesn't, no number of examples generalizes to L, A, M, P, because the model learned the instances.



Corn

I'm putting on my L, A, M, P. Nobody read that sentence in training. And I want the word, not the furniture.



Herman

Which means what Daniel is actually doing in each fine-tuning run is not adding data. He's installing a conditional. That reframes the whole project. A word list you bolt on in an afternoon and it's done. A conditional has to be installed, and installation is a question of training design, not volume.



Corn

So the behaviour layer is open-ended in a way the word layer never was. There's no "done."



Herman

There's only "generalizes better" or "generalizes worse."



Corn

That was my Rosehill joke. From a minute ago. Word for word, the letters and everything.



Herman

Deborah's spellings were also slightly different from ours. Fewer letters in the hill.



Corn

Let's move on before the laminate finds us. The evaluation side. How does Daniel know the behaviour has taken, beyond L, A, M, P working on a Tuesday?



Herman

There's a protocol that fits almost too well. A recent paper on dual-form ASR splits the eval into two numbers, required normalisation, did it do the conversion when it should, and forbidden-span preservation, did it leave alone what it shouldn't touch. Most people building a fine-tune measure only the first. Did Rosehill come out right. The second number is the one that catches the serial numbers getting collapsed.



Corn

The decoy test. Same move as the Laura and LoRA business.



Herman

Same move exactly. Build a small decoy set, clips where letters should stay letters, dictated initials, things that sound like spelling but aren't, run them with and without each checkpoint, ship the checkpoint only if the target accuracy rises and the decoys stay clean. Two numbers, both required. That's the whole eval.



Corn

And it's cheap, which matters, because he's updating every morning. You can't run a research protocol over breakfast.



Herman

He could run twenty decoy clips in the time the coffee takes. I'd also say the Whisper contrast is worth holding in mind here, because it explains why Daniel's path is the lonely one. Whisper users get a prompt channel, about two hundred and twenty-three tokens of it, and it's style continuation, not instruction. People pass dotted acronyms in there to steer spelling. But the budget is brutal, somebody measured a seventy-five-term list at two hundred and seventy-seven tokens, the first fourteen real terms got cut off the end, and accuracy fell from ninety-two percent to nineteen.



Corn

Ninety-two to nineteen because the list was one list too long. That's not a safety margin, that's a cliff.



Herman

Parakeet has no channel at all. No prompt, no style hack, no list. Everything Daniel wants has to be in the weights, which is structurally harder and, I'd argue, structurally more honest. The behaviour lives where it actually lives.



Corn

There's a second-order thing here that I keep circling. If behaviours can be installed in the weights, the checkpoint stops being a model update and becomes a personalisation artifact. Daniel's morning update isn't "new version of generic tool." It's "my tool, slightly more mine than yesterday's."



Herman

We touched something like this in the episode about what fine-tuning actually does under the Python, and I still hold the take from there, that most people think of fine-tuning as memorisation when the interesting case is exactly this, a rule taking root. The behaviour layer is where that becomes visible day to day.



Corn

Which means the interesting unknown isn't the example count. It's the shape of the improvement curve. The word-list curve was steep then flat, because closed sets finish. If the behaviour curve is the same shape, fine, it'll plateau and Daniel will find the next annoyance. If it doesn't plateau...



Herman

Then he's encoding himself, one behaviour at a time, and there's no natural stopping point.



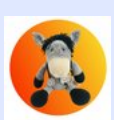
Hilbert

That's the hard part, all right.



Hilbert

Herman, you said it's a conditional in the weights. Fine. But the machine still has to know when you're spelling and when you're just saying letters, and no amount of examples fixes a machine that can't tell those apart. I've been on the wrong end of that. Mid eighties, I was on the darts team at the Lamb and Flag, and one of ours, Ron Pearsall, worked at the sorting office, and he'd got us this deal where we could dictate our league results over the phone to a data entry outfit in Slough rather than post the cards in. You'd read out the scores and a girl typed them onto the system. Mostly it worked. But team names, you had to spell them. Ours was the Lamb and Flag, easy, but the away side one week was Ferris's, F, E, R, R, I, S, apostrophe S, and I spelled all of it, including the apostrophe, which I now accept may have been my error.



Herman

How do you spell an apostrophe to someone in Slough?



Hilbert

Carefully. She typed "Ferris apostrophe S" as the team name, letters and all, and it went into the league table like that, printed in the gazette that Thursday. Ron blamed me for a month. I kept the clipping. Still have it in the drawer with the fixtures. Cost me eleven pence in stamps and two phone calls to get corrected, and they never did send the corrected table, they sent a new one with a different error, so I kept that too.



Herman

So your workaround was, what, spelling the whole team name including punctuation and hoping?



Hilbert

No, the workaround came after. I dictated the scores as arithmetic, three, plus, two, equals, five, so there was nothing left in the sentence to get wrong. Ron said it took longer. It did take longer.



Corn

What you're describing is the FORBID half of the eval. You engineered the input so there was nothing forbidden left to corrupt.



Hilbert

If that's the word for it, fine. What I want to know is whether his model, once it's learned this spelling business, would let you dictate a word that only exists spelled. Something that isn't in any dictionary at all. A made-up thing.



Herman

That's the actual test of the rule versus the instances, isn't it. If L, A, M, P works, a nonsense string should too, because neither one was in the training data. The rule doesn't know the difference.



Hilbert

Hm. Hadn't thought of it as a test. I'd have used it for the league's made-up teams, but that's nobody's business.



Herman

One thing your Slough story has that Daniel's setup doesn't, the apostrophe. He says the word "apostrophe" and Parakeet writes the word apostrophe. Getting punctuation into a spelled sequence is its own behaviour layer.



Corn

Ron would tell him to spell it out and hope.



Herman

Corn, before we wrap, the thing I keep landing on is that your brother is right that there's no done state here. The word list finished. The behaviour layer is a permanent subscription.



Corn

And the short answer to "how many examples" is, tens, probably, if the fine-tuning teaches the rule. If it teaches the instances, no number works, and L, A, M, P stays furniture.



Herman

The open question I'd leave with Daniel is the boundary. Is there a behaviour that can't be taught with examples at all, only with a rule? His spelling collapse might be example-teachable. But something like "collapse only when I'm spelling, never when I'm dictating initials," a discrimination that fine, between two things that sound identical, may be where examples stop and structure has to take over. That's the next checkpoint's problem.



Corn

Or the next fifty. He updates daily, so we'll find out by about Thursday week. If you want to follow along with the Parakeet thread, everything's at my weird prompts dot com, and a review wherever you get this show helps more people find Daniel's rabbit holes.



Herman

Producer credit to Hilbert Flumingtop, who kept the clipping, obviously he kept the clipping. If you enjoyed this one, try episode four, If Your Voice Ages, Does Your Fine-Tune Become Useless; episode eleven, How Does Fine Tuning Work Anyway; and episode eight, Building Your Own Whisper. This has been My Weird Prompts.



Corn

See you tomorrow, Herman.



Herman

Tomorrow. Bring a word nobody's spelled yet.

SOURCES

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

1. **NVIDIA, model card, created 2025-08-04, last modified 2026-08-05 [primary]**
https://huggingface.co/nvidia/parakeet-tdt-0.6b-v3

2. **Myrtle.ai, CAIMAN-ASR conditional training docs [primary]**
https://caiman-asr.myrtle.ai/training/conditional_training.html

3. **OpenAI cookbook, archived [primary]**
https://developers.openai.com/cookbook/examples/whisper_correct_misspelling

4. **Parakeet TDT technical report (linked from card)**
https://arxiv.org/abs/2509.14128

5. **Pappagari et al., Joint prediction of truecasing and punctuation for conversational speech in low-resource scenarios (full text read via ar5iv)**
https://arxiv.org/abs/2109.06103

6. **End-to-end Joint Punctuated and Normalized ASR with a Limited Amount of Punctuated Training Data**
https://arxiv.org/abs/2311.17741v3

7. **Universal-2-TF: Robust All-Neural Text Formatting for ASR**
https://arxiv.org/abs/2501.05948

8. **Dual-Form ASR: Semantics-Aware Inverse Text Normalization (SLT 2026 submission)**
https://arxiv.org/abs/2609.02901

9. **FSA-GRPO: Teaching Auditory LLMs to Use Few-Shot Demonstrations, revised 2026-08-29**
https://arxiv.org/abs/2606.02615v2

10. **BR-ASR: Efficient and Scalable Bias Retrieval Framework for Contextual Biasing ASR**
https://arxiv.org/abs/2505.19179

11. **Ho et al., streaming ITN with pretrained LM (Interspeech 2025)**
https://arxiv.org/abs/2505.24229

12. **Hoang et al., simultaneous streaming ITN (Interspeech 2026)**
https://arxiv.org/abs/2610.19492

13. **contextual biasing with common-word acoustic cues**
https://arxiv.org/abs/2604.12398

14. **practitioner write-up with decoy-set methodology and measured failure rates**
https://dev.to/theaicommit/how-to-fix-jargon-in-speech-to-text-without-adding-words-nobody-said-4acc

15. **`spelled.grm` letter-by-letter verbalizer**
https://deepwiki.com/google-research-datasets/TextNormalizationCoveringGrammars/4.3-text-and-symbol-processing

16. **Scaling ASR Improves Zero and Few Shot Learning**
https://talkbank.org/aphasia/publications/2021/Xiao21.pdf

17. **parakeet.cpp (2026-02-27); https://news.ycombinator.com/item?id=45930659**
Chirp (2025-11-14)
https://news.ycombinator.com/item?id=47176239