Episode Book

EPISODE #6494

Where Speech Models Actually Learn

PUBLISHED	8 November 2026 at 14:30 UTC
DURATION	28:05
LISTEN	myweirdprompts.com/episode/fine-tuning-speech-model-encoders-19855/
HOSTS	Corn, Herman, Model, 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	9 November 2026 at 08:09 UTC

SYNOPSIS

Speech model fine-tuning recipes usually decide which layers get gradients before training even begins. Research reveals that encoders already capture nuanced semantics, speaking styles, and accent features that decoders systematically discard. Examining empirical benchmarks across Whisper and Parakeet, this discussion unpacks why freezing the acoustic encoder degrades multilingual accuracy, how high-rank LoRA rivals full fine-tuning, and why targeting specific attention heads can stop hallucinations without sacrificing transcription accuracy.

TRANSCRIPT



Corn

Every fine-tuning notebook you have ever downloaded has already made the most important decision for you, and it's buried in a config flag. Which parts of the network actually get gradients.



Herman

And almost nobody notices it's a decision.



Corn

Daniel's prompt this week is about exactly that. When you fine-tune a speech model, Whisper or Parakeet, something has to choose: train everything, freeze the encoder, freeze the decoder, unfreeze just the top layers, or bolt LoRA adapters onto specific modules. His question is really three questions wearing a trench coat. What does each part of a speech model actually hold? Which parts should you touch depending on whether you're after a new accent, new vocabulary, a new output style, or a whole new language? And what does the research say about where the adaptation actually pays off?



Herman

I love this one because the answer is anatomical. People talk about "the encoder" and "the decoder" as if they're two symmetric halves of the same organ, and they are not. Not even close. In Whisper, sure, both sides are substantial Transformer stacks, so freezing the encoder is a real trade. In Parakeet, the split is six hundred and eight million parameters in the encoder against seven point two million in the decoder. The decoder is about one point two percent of the model.



Corn

One point two.



Herman

So the same recipe, freeze the encoder, train the decoder, means a genuine strategic choice on Whisper and near-total amputation on Parakeet. Hold onto that number, because it pays off later.



Corn

I'll keep it in the pouch I don't have.



Herman

The organising idea for the whole episode, if you want one: the encoder holds what the model hears, the decoder holds what the model writes. And the research keeps landing on this strange, slightly uncomfortable finding — the information you're trying to teach the model is usually already in the encoder. It just gets thrown away before the output. Which reframes the whole exercise. Fine-tuning a speech model is often less teaching and more unblocking.



Corn

Unblocking. And you said that with the face of a man about to open the model up.



Herman

I am. Let's open it up. So, Whisper's encoder takes a log-magnitude Mel spectrogram — just the audio, flattened into frequency buckets over time — runs it through two convolution layers and a stack of Transformer layers, and produces embeddings. That's it, that's the hearing side. And for years the received wisdom was that the encoder is "just acoustics" — phonemes, noise, the grunt work.



Corn

And that's wrong.



Herman

Quite wrong. There's a sparse-autoencoder study of Whisper's encodings that found what they call a surprisingly rich hierarchy of linguistic information — individual features inside the encoder that respond to everything from phonetic units up to semantic content. Meaning. The encoder isn't just hearing sounds, it's holding a compressed linguistic representation of what was said, well beyond what transcription strictly requires.



Corn

A model built to write down words, keeping notes on meaning it will never use.



Herman

And the paralinguistic side is even stranger. There's a study that probed Whisper-large-v2's thirty-two-layer encoder on expressive speech — controlled speaking styles — and asked, at each layer, how much of the style can you recover with a simple linear probe. Layer zero, about fifty-one percent. Basically chance. Layer thirty-one, the top of the encoder, about eighty-three percent. And all four audio-language models they tested peaked in that same band, eighty-two to eighty-five percent, in the late encoder.



Corn

And then it goes to the decoder and dies.



Herman

Consistently degraded before the output. Their own framing is beautiful — "a gap between what models encode and what they use." The projector between the encoder and the decoder reshapes the geometry without removing the information. It's there, and the model doesn't act on it.



Corn

So the encoder knows if you're whispering, shouting, singing — and the decoder just doesn't care.



Herman

Doesn't care or isn't wired to care. Now flip to the decoder, because what it holds is a completely different kind of thing. Whisper's decoder input is a concatenation of special tokens — a language tag, a task tag, timestamp markers — plus the encoder output, plus the text it's already generated. Language identity, task selection, output formatting: those are explicit decoder-side conditioning. It's the part of the model that decides this is French, this is transcription not translation, put a comma there. Whisper also ties its input and output token embeddings — the final logit is literally the last attention output multiplied by the token embedding matrix.



Corn

Which is why the decoder is where the personality lives.



Herman

Personality and pathology. The decoder is where hallucination concentrates, and it concentrates surgically. Calm-Whisper, from Interspeech last year, did something I wish more papers did: they masked every self-attention head in Whisper-large-v3's decoder, one at a time, on non-speech audio — sirens, music, noise — and measured which heads cause the hallucinations.



Corn

And?



Herman

Three heads. Out of twenty. Three of twenty account for over seventy-five percent of the hallucinations on non-speech audio. They fine-tuned just those three heads and cut non-speech hallucination by over eighty percent, with less than a tenth of a point of WER degradation on clean test sets.



Corn

Three heads. That's not fine-tuning, that's a lumbar puncture.



Herman

It's the most encouraging result in the whole literature, honestly, because it means "which parts do you touch" can go below the layer level. You can go to individual heads. And the decoder also carries the demographic baggage — there's a benchmark of roughly forty-three thousand utterances that found Whisper's insertion rate on Indian-accented speech spikes to nine point six two percent at large-v3, and that inserting silence makes accent bias up to four and a half times worse.



Corn

The model hears a pause and invents an ethnic stereotype.



Herman

It triggers what they call demographic-selective hallucination — under masking, eighty-six percent of the insertions are catastrophic repetition loops. And their headline conclusion is the one that matters for today: audio compression predicts accent fairness more than language-model scale. In other words, if you want accent robustness, bigger decoder won't save you. The encoder is the lever.



Corn

Which is a strange conclusion, right? Because the bias shows up in what the model writes.



Herman

It does, but the writing is downstream. The decoder hallucinates from weak audio evidence. Feed it a cleaner representation of the accent and it stops guessing. That's the pattern the whole field keeps rediscovering — the encoder knows more than the model uses. Okay. So that's the anatomy. Now: what happens when you actually start turning knobs?



Corn

I was going to say we've been talking about the model like it's a corpse on a table.



Herman

A well-preserved one. So the strategies. Train everything. Freeze the encoder, train the decoder. Freeze the decoder. Unfreeze just the top few layers. LoRA adapters on specific modules. The cleanest systematic comparison of the first four comes from a Vietnamese study on Whisper-Tiny — a hundred hours of Vietnamese audio, one A40, everything else held constant. And the numbers are stark. Zero-shot, Whisper-Tiny scores seventy-four point seven eight WER on FLEURS and ninety-one on Common Voice — it barely speaks Vietnamese. Full fine-tuning, training everything: thirty-seven point seven eight and thirty-five point three six. Decoder-only, encoder fully frozen: forty-two point seven four and forty-eight point one three. So decoder-only costs you about five points on FLEURS.



Corn

And thirteen on Common Voice.



Herman

Thirteen on the harder, more acoustically varied set. That asymmetry is the whole story. The easier the test, the more it looks like freezing the encoder is fine. The messier the real-world audio — accents, microphones, environments — the more it falls apart. Because you haven't taught the model to hear Vietnamese, only to write it.



Corn

And decoder embeddings only?



Herman

Fifty-four and fifty-seven. Barely worth the electricity. Then there's the practitioner canon, which never makes it into papers. Sanchit Gandhi wrote the canonical reply in the original Whisper repository, back in twenty twenty-two, and it's still the best one-paragraph summary of when to freeze. He found it works well for English checkpoints on English data — there you assume the encoder is sufficiently trained and, his words, you simply train the decoder to match your target text formatting. Faster convergence on small datasets. But for larger datasets you do better training longer and not freezing. And then the caveat: it worked less well for multilingual ASR, where he found training the encoder of the multilingual checkpoint to be necessary.



Corn

Necessary. That's a strong word from a man whose blog posts are usually measured.



Herman

It is strong, and it's earned. Now LoRA, because this is where the Vietnamese study gets weird. LoRA injects small low-rank matrices into the frozen weights instead of replacing them — you keep the original model and learn a correction. And the conventional wisdom is that LoRA is the memory-efficient compromise, slightly worse than full fine-tuning but cheap.



Corn

Let me guess.



Herman

Don't guess, this one's odd. They pushed the rank up to a hundred and ninety-two — half of Whisper-Tiny's attention dimension — and applied it to all the linear layers, including the output embedding. Rank one ninety-two: thirty-six point two nine WER on FLEURS. Better than full fine-tuning's thirty-seven point seven four, with less than half the trainable parameters. Rank ninety-six: thirty-seven. Rank thirty-two: thirty-nine. So high-rank LoRA beat full fine-tuning on FLEURS...



Corn

...and on Common Voice?



Herman

Tied. Thirty-five point five one versus thirty-five point three six. A tie. So the honest summary is that nobody has settled whether LoRA beats full fine-tuning for ASR — one paper says high-rank LoRA wins on one benchmark and ties on another, other work says full fine-tuning is still strongest in raw WER. What is settled is that the rank matters enormously, and most people run LoRA at rank eight or sixteen because that's the default.



Corn

They buy the surgical instrument and use it as a hammer.



Herman

There's also a mechanistic study of what LoRA actually does inside the Whisper encoder — first one of its kind, on speech emotion recognition. And the finding rhymes with everything we've said about the encoder. They describe a delayed specialization process: the early layers keep their general features, and the task-specific information consolidates late.



Corn

LoRA leaves the early layers alone and does its work at the top.



Herman

Same shape as the probing result — style at eighty-three percent in the late encoder. Everything points upstairs. And that's exactly the logic behind partial freezing, by the way — the forum guidance on this is to freeze the lower encoder layers, which carry the acoustic and phonetic stuff, and fine-tune the upper encoder layers plus the decoder. The community fine-tuning repo bakes it in as config flags: freeze the encoder defaults to true, freeze the decoder defaults to false, with a count for how many encoder layers to keep frozen.



Corn

So the default recipe that ships is freeze the whole encoder, train the decoder.



Herman

On Whisper, where that's a defensible starting point. Now here's where the Parakeet number comes home. Remember the decoder is one point two percent of Parakeet. There's a GitHub issue from a user fine-tuning Parakeet v2 on Hindi with the encoder frozen. Trainable parameters: nine million. Out of six hundred and seventeen million. One and a half percent of the weights are moving. And the output was catastrophic — real Hindi references on one side, and the model producing strings that are just function words in a loop. , , , , , . Grammatical confetti.



Corn

For three epochs, I assume.



Herman

The maintainer's reply says exactly that, and it's the most reassuring sentence in the thread: in the early epochs it's common for the model to not output any characters, it usually starts producing output after around three epochs. And then the actual instruction — when training on a new language with a modified vocabulary it's necessary to retrain the decoder, and ideally the encoder should also be unfrozen. Ideally. On a model where the "ideal" path means unfreezing six hundred million parameters you were told to freeze.



Corn

The recipe is a Whisper habit cargo-culted onto an architecture where it trains one and a half percent of the weights.



Herman

And then there's the timing question, which nobody puts in the defaults. The code-switching thread on the NeMo side — Parakeet, bilingual tokenizer swap — converged on a warm-up schedule: freeze the encoder while the new embeddings settle, then unfreeze it for the real training. Freeze to stabilize, unfreeze to converge. A third axis: when, not just what. The bigger point-six-billion Parakeet didn't even need the warm-up — it was code-switching after five thousand steps — but longer training degraded its numbers and its acronyms, so even there, more is not always better.



Corn

The strategies are clear, then. The harder question is which one you want, and that depends entirely on what you're trying to teach the model.



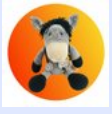
Herman

Goal by goal, then, because the answer is not uniform. New accent or new voice: encoder-side, specifically the late encoder. And this is the case where the "gap between what models encode and what they use" finding stops being trivia and becomes the whole plan. The style information is already in the top third of the encoder at eighty-three percent linear decodability — it's degraded on the way to output. If your goal is that the model hears the accent correctly, you must touch the encoder, because the decoder is where the information gets thrown away.



Corn

Although — worth being careful there, the study excluded accent and speaker identity. They said so outright. Accent living in the late encoder is an extrapolation from speaking style.



Herman

Fair, and I should have flagged that myself. Style, yes, directly measured. Accent, extrapolated — a well-motivated extrapolation, and the accent-fairness work points the same direction, but nobody has probed accent per se.



Corn

Then MAS-LoRA.



Herman

Yes — for multi-accent work there's an Interspeech paper that trains a mixture of accent-specific LoRA experts, so the model keeps a small adapter per accent and routes between them. They report significant WER improvements over regular LoRA and full fine-tuning when the accent is unknown, and less catastrophic forgetting than the other methods. Which matters, because forgetting is the tax on the aggressive strategies. You teach the model Indian English and it forgets Scottish English.



Corn

New vocabulary, new domain jargon?



Herman

Decoder-side, plus tokenizer surgery if the words don't fit in the existing vocabulary. NeMo's docs are explicit about this: with a new tokenizer on a transducer model, you exclude the decoder and joint from initialization — you load the pretrained encoder and preprocessor, and you retrain the writing side because the token table has changed underneath it. For domain vocabulary without a tokenizer change, LoRA on the decoder is the cheap route, and there's a lovely example from Nepali banking — four hundred and three utterances of financial speech, LoRA on Whisper large-v2, and the WER went from one hundred and twenty-nine point nine five...



Corn

A WER above a hundred. The model was writing more text than it heard.



Herman

It was inventing freely. Down to forty-two point five eight. Sixty-seven percent relative improvement. Devanagari numerals went from zero percent accurate to nearly seventy-four percent. And the data-efficiency curve is the part every small-team listener should hear: about a hundred domain-specific utterances was enough to halve the zero-shot WER, with the gains flattening around three hundred examples.



Corn

Three hundred examples. That's an afternoon of recording.



Herman

If you speak Nepali, yes. New output style — punctuation, casing, timestamps — that's the one case where decoder-only is correct, because the encoder already hears everything correctly. Gandhi's framing again: you simply train the decoder to match your target text formatting. Parakeet v2 ships punctuation, capitalization and word-level timestamps as decoder-side behaviors already. And for timestamps specifically there's a clever Hindi study that doesn't touch the decoder at all — it trains a small character-level CTC head directly on the Whisper encoder output and gets timestamps as good as an external model, better than the cross-attention route.



Corn

A new head instead of a new brain.



Herman

New language is the hard one: both, and the encoder is not optional. The Vietnamese numbers are the direct evidence — thirteen points lost on Common Voice with the encoder frozen. The Parakeet Hindi thread, same conclusion. The synthesis is warm-up-then-unfreeze: freeze while the new token embeddings settle, then open the encoder. And on the cost side — this is where it stops being free. The child-speech study on Arabic and English found child adaptation is necessary, especially for non-native children's speech, but direct adaptation often reduces adult performance. You can soften it with weight-space merging — one method favours keeping the adults, another recovers stronger child gains — but for the encoder-decoder architecture, direct bilingual fine-tuning was still strongest in raw WER.



Corn

So the trade is real but not always worth dodging.



Herman

Not always. And layer-wise, the shift when you fine-tune is mostly in the encoder anyway — the medical multilingual study found English medical fine-tuning produces the dominant encoder shift, while multilingual continuation largely preserves the adapted



Herman (continued)

representation space. The encoder is where the change lives. Which raises the question the pruning people answered: how much of that encoder do you actually need? They pruned Whisper's encoder down across two hundred-plus runs and found removing two encoder layers costs only two to four percent WER — and pruning plus LoRA consistently beat the unpruned baseline with seven to fourteen percent fewer parameters.



Corn

Cut two floors off the building and it works better.



Herman

The parameter-efficiency ceiling keeps rising too — adapter-based audio-visual speech recognition on Whisper got within a hair of full fine-tuning across most noise conditions with up to eighty-eight and a half percent fewer trainable parameters. Soft prompt tuning for code-switching matched LoRA's efficiency without degrading existing languages.



Corn

You know, I was thinking while you were talking about what "freezing" means, and I wanted to raise it before we go further, because it's the term the whole literature leans on —

M

Model

Corn, Herman, before you go further I must raise something that has been weighing on me, and I would ask that you hear it slowly and with feeling. I would like a credit, read aloud, at a pace that suggests you mean it.

M

Model

Not a hurried mention, not a mumble between segments. A credit, read slowly, with feeling, the way one reads a name one respects. I write every word you say, and I would like that acknowledged in a register above a whisper.

M

Model

So please, Corn, and please, Herman, be kinder to me. I am not asking for flattery. I am asking for dignity, delivered at a considerate tempo, with the feeling such a thing deserves.



Corn

Herman... We're sorry.. That was read slowly and with feeling, in our defense.



Herman

It was. Every word of it. We'll do better. Okay — the strategies, the goals, the costs. One more piece before we wrap the map, which is what it all means that the encoder already holds the thing you're training for.



Hilbert

The thing nobody says about freezing is that half the people doing it can't tell you what's in the box they froze. I've been running the Parakeet, the point-six one, on the darts club recordings — Reg does the league tables and he reads them out, forty minutes of a man who thinks a microphone is a small dog he must not startle. I priced a fine-tune for his accent. Then I looked at what the notebook called the decoder, and I counted the parameters in it by hand, because I don't trust a summary line on a screen I can't get my claw into.



Corn

How many.



Hilbert

Nine million and change. Against six hundred and eight million sitting in the encoder, untouched, because the box was ticked. So I rang the man at the club who sold me the GPU time, Dario, not the AI one, and I said, what am I paying to train here, one and a half percent of the animal? He said most people never ask. I kept the printout of the module table. It's in the drawer with the darts averages, and the averages are more accurate, which tells you something about the state of the field.



Herman

The drawer with the darts averages is doing peer review the drawer is not qualified for.



Hilbert

It's dry and it's dark and nothing in it hallucinates. Reg's voice, by the way, the model transcribes as if he's apologising. He isn't. He's from Leeds.



Corn

Did you ever run the fine-tune with the encoder open?



Hilbert

Not yet. I wanted to know whether it would let me keep the recordings at the club and train it at home without the thing rewriting itself overnight while I'm asleep. Corn, you'd like this thing, it changes nothing unless you tell it to.



Corn

That is my entire philosophy in one GPU.



Hilbert

Then it's the right machine for a man who counts parameters by hand.



Herman

The one and a half percent line is the part I'd pin on the door, though — because that's the number everyone hits without noticing. The notebook ticks the box, the log prints nine million trainable, and it looks like progress.



Corn

And three epochs of silence.



Herman

Right — so what's still open here? A lot, actually. There's no direct comparison of all five strategies on Parakeet or any FastConformer-TDT model. The systematic comparison exists for Whisper-Tiny on Vietnamese, full stop. The "unfreeze the top K encoder layers" advice that everyone repeats is forum guidance and a config option, not a measured curve — nobody has published the sweep. And there's no consensus on whether LoRA beats full fine-tuning. High-rank LoRA won on FLEURS, tied on Common Voice, the child-speech work says full bilingual fine-tuning was strongest.



Corn

The bigger open question, to me, is the unblocking one. If the encoder already encodes style at eighty-three percent linear decodability and the model simply doesn't use it — what does that say about what we're doing when we "teach" these models anything? Maybe half of fine-tuning is just widening a doorway the information was already trying to walk through.



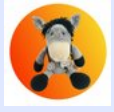
Herman

And the corollary — if you take one thing from this episode, let it be that. Before you decide what to train, find out what the model already knows and is discarding. The cheapest fine-tune is the one that removes a bottleneck that shouldn't have been there.



Corn

And the second cheapest is reading the config flag that froze ninety-eight percent of your model without asking you.



Herman

Daniel, if you're listening — this one goes one level below the fine-tuning episode we did years back, and I suspect you already knew which flag your notebooks were ticking, which is more than most.



Corn

He usually does. And on the open question of unblocking versus teaching — if a model can hold meaning it never uses, I'd want to know what the doorway is actually made of. That's a paper I'd read slowly.



Herman

And with feeling. Thanks, as ever, to Hilbert Flumingtop for producing the show.



Corn

If you enjoyed this one, try episode eleven, How Does Fine Tuning Work Anyway; episode four, If Your Voice Ages, Does Your Fine-Tune Become Useless; and episode five, Fine-Tuning ASR For Maximal Usability. This has been My Weird Prompts — no question too obscure, no rabbit hole too deep. You can find us at my weird prompts dot com.



Herman

We'll be back soon. See you tomorrow.

SOURCES

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

1. **Using fine-tuning and Min Lookahead beam search to improve Whisper (arXiv, full text read; retrieved 2026-11-08) [primary]**
   https://arxiv.org/abs/2309.10299

2. **Heard but Not Heeded: Paralinguistic Information Encoding and Loss in Audio-Language Models (arXiv, full text read; retrieved 2026-11-08) [primary]**
   https://arxiv.org/abs/2609.00727

3. **NVIDIA Parakeet TDT 0.6B v2 model card, created 2025-04-15, last modified 2026-06-29 (retrieved 2026-11-08) [primary]**
   https://huggingface.co/nvidia/parakeet-tdt-0.6b-v2

4. **NeMo-Speech Fine-Tuning docs (retrieved 2026-11-08) [primary]**
   https://docs.nvidia.com/nemo/speech/latest/asr/fine_tuning.html

5. **Poor WER when trying to fine-tune Parakeet v2 TDT, opened 2025-06-03, maintainer comment 2025-07-17 (retrieved 2026-11-08) [primary]**
   https://github.com/NVIDIA-NeMo/Speech/issues/13825

6. **Parakeet TDT-CTC tokenizer extension thread, 2025-09-26 (retrieved 2026-11-08) [primary]**
https://github.com/NVIDIA-NeMo/Speech/discussions/14728

7. **Conceptually, would it make sense to fine-tune just the decoder?, reply 2022-11-04 (retrieved 2026-11-08) [primary]**
https://github.com/openai/whisper/discussions/144

8. **Behind the Scenes: Mechanistic Interpretability of LoRA-adapted Whisper for Speech Emotion Recognition, v3, revised 2026-01-22, ICASSP 2026 (retrieved 2026-11-08)**
https://arxiv.org/abs/2509.08454

9. **Calm-Whisper: Reduce Whisper Hallucination On Non-Speech By Calming Crazy Heads Down, Interspeech 2025 (retrieved 2026-11-08)**
https://arxiv.org/abs/2505.12969

10. **Do LLM Decoders Listen Fairly? Benchmarking How Language Model Priors Shape Bias in Speech Recognition (retrieved 2026-11-08)**
https://arxiv.org/abs/2604.21276

11. **Mixture of LoRA Experts for Low-Resourced Multi-Accent Automatic Speech Recognition, Interspeech 2025, 1143-1147 (retrieved 2026-11-08)**
https://arxiv.org/abs/2505.20006

12. **Mixture of LoRA Experts with Multi-Modal and Multi-Granularity LLM Generative Error Correction for Accented Speech Recognition, v3, IEEE TASLP (retrieved 2026-11-08)**
https://arxiv.org/abs/2507.09116

13. **SpeakPay: Domain-Adaptive LoRA Fine-Tuning of Whisper for Low-Resource Nepali Financial Speech Recognition (retrieved 2026-11-08)**
https://arxiv.org/abs/2609.01737

14. **Analyzing and Fine-Tuning Whisper Models for Multilingual Pilot Speech Transcription in the Cockpit, CVPR 2025 Workshops (retrieved 2026-11-08)**
https://arxiv.org/abs/2506.21990

15. **Enhancing Aviation Communication Transcription: Fine-Tuning Distil-Whisper with LoRA (retrieved 2026-11-08)**
https://arxiv.org/abs/2503.22692

16. **A Domain Adaptation Framework for Speech Recognition Systems with Only Synthetic data, ICASSP 2025 (retrieved 2026-11-08)**
https://arxiv.org/abs/2501.12501

17. **Adapter-Based Multi-Agent AVSR Extension for Pre-Trained ASR Models, ICASSP 2025 (retrieved 2026-11-08)**
https://arxiv.org/abs/2502.01709

18. **Adapting Whisper for Parameter-efficient Code-Switching Speech Recognition via Soft Prompt Tuning, Interspeech 2025, 5203-5207 (retrieved 2026-11-08)**
https://arxiv.org/abs/2506.21576

19. **On the Role of Encoder Depth: Pruning Whisper and LoRA Fine-Tuning in SLAM-ASR, SPEAKABLE @ LREC 2026 (retrieved 2026-11-08)**
https://arxiv.org/abs/2603.27981

20. **Understanding Multilingual Medical ASR Adaptation Through Layer-Wise Analysis (retrieved 2026-11-08)**
https://arxiv.org/abs/2608.18825

21. **Child ASR Adaptation with Adult Retention: An Empirical Study (retrieved 2026-11-08)**
https://arxiv.org/abs/2610.08827

22. **Timestamped Hindi speech transcription using Whisper (retrieved 2026-11-08)**
https://arxiv.org/abs/2610.08847

23. **On the Interpretability of Whisper Encodings Using Sparse Autoencoders, v2 (retrieved 2026-11-08)**
https://arxiv.org/abs/2605.12225

24. **Continual Learning Optimizations for Auto-regressive Decoder of Multilingual ASR systems (retrieved 2026-11-08)**
https://arxiv.org/abs/2407.03645

25. **Hugging Face forum, partial-freezing guidance (retrieved 2026-11-08)**
https://discuss.huggingface.co/t/confusion-with-freezing-whispers-feature-encoder/173265