Episode Book

EPISODE #5922

Teaching a Model Your Voice: SFT vs DPO

PUBLISHED	13 October 2026 at 00:06 UTC
DURATION	21:34
LISTEN	myweirdprompts.com/episode/fine-tuning-style-sft-vs-dpo-07870/
HOSTS	Corn, Herman, Hilbert
SCRIPT MODEL	deepseek-flash
VOICES	chatterbox-regular (chatterbox-local)
PIPELINE	V5.4
SOURCES	13 cited by the research (listed at the end)
BOOK GENERATED	19 October 2026 at 17:41 UTC

SYNOPSIS

You can't reliably prompt a model into your voice — system prompts plateau. So what actually works? This episode works through three data-generation strategies for style fine-tuning: regenerating transcripts with global comments, hand-editing transcripts and training on the diff, and training a purely negative counterweight against overused phrases. A July arXiv paper by Federico Boggia on epanorthosis (the mid-sentence self-correction tic) tested almost exactly this, and one implementation failed in a very instructive way: a small-data DPO adapter learned to rank outputs correctly and then raised the density of the target phrase. Supervised fine-tuning on the same data cut it by 98%. We map each method onto that evidence, then get to the practical recipe — how many examples you actually need, why "200 transcripts" is the wrong unit of counting, and why LoRA or QLoRA is the unambiguous choice for a style adapter.

TRANSCRIPT



Corn

Here's the question that's been sitting in the back of my head since I read this one — if you want a model to write more like you, do you teach it what to say, or just what not to say?



Herman

That's the whole thing, isn't it. And Daniel's given us three different answers.



Corn

Daniel's been circling this for a few episodes, the fine-tuning thing, and this time he wants to bring it together properly. He's happy with DeepSeek — that's the starting point, it's not a complaint. It's that there are small nudges to the dialogue he can't get out of system prompting reliably, which is true, and he's stopped pretending it isn't.



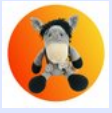
Herman

System prompting gets you ninety percent there and then plateaus.



Corn

And then plateaus forever. So he's got three methods. Method A — take twenty transcripts, write a global comment on each one, good, bad, could be better, hand the transcript back to the model, let it regenerate, and train on the diff against the original. Method B — same twenty transcripts, but he edits them by hand, light touch, targeted, removing the phrases he's tired of and finding alternatives, and the diff is the training signal. Method C — he prints the transcripts, reads them, underlines the overused phrases, and trains a purely negative counterweight against them.



Herman

And he tells us he likes C the least.



Corn

He does, he says it explicitly. It's purely negative, whereas B gives him latitude to actually find better phrasing. So let's take these one at a time, and then get to the part Daniel actually asked for — the recipe.



Herman

I want to push back on one thing before we start, because I think the framing matters. This isn't a question of whether fine-tuning works. That's settled, we've done episodes on the mechanics. The actual question is which data-generation strategy produces a model that writes the way Daniel wants, and what the minimum viable version of that is.



Corn

Which is a better question.



Herman

It's a harder one. So where DeepSeek actually sits right now — V4.1-Flash shipped in September, and it's the live API model. V4-Pro traffic got rerouted onto V4.1-Flash at V4.1-Flash pricing a few days later, which tells you something about how DeepSeek feels about their own flagship.



Corn

They consolidated.



Herman

The V4-Flash official version went open-weight in August, three hundred and four billion parameters, GGUF quantization published. And here's the constraint that shapes everything: DeepSeek's hosted API doesn't document a first-party managed fine-tuning endpoint. Not for Flash, not for V4-Pro.



Corn

So you can't just click a button and hand them your transcripts.



Herman

You cannot. Fine-tuning here means open weights, or you manage the training yourself, or you go through a third party. That's the landscape.



Corn

Twenty transcripts. Before we get into which method, I want to know if twenty is even a number.



Herman

It's not, and we'll get there properly, because that's the part where the literature has an actual answer. But look — before we can pick a winner, we need to understand what each method is actually doing under the hood. And there's a paper from July that tested almost exactly this.



Corn

The epanorthosis one.



Herman

Federico Boggia, posted to arXiv in July. Epanorthosis is the rhetorical term for going back and correcting yourself mid-sentence, and the paper is about a model that does it too much. Which is the same shape of problem Daniel has, just one tic instead of twenty.



Corn

So it's a proof case.



Herman

It's better than a proof case, because he ran it twice, two different ways, and one of them failed in a way that's instructive. Method A, if you implement it as preference ranking — original transcript versus regenerated transcript, which is better — that's DPO. Direct preference optimization.



Corn

And the paper says that fails.



Herman

Preference optimisation fails, supervised fine-tuning works. That's the finding, near enough verbatim. The DPO adapter was trained on five hundred and seventy preference pairs. High preference accuracy on the training set — it learned to rank the pairs correctly. And then it failed to generalise completely. It actually raised the density of the target phrase. From about sixty per ten thousand words to sixty-nine.



Corn

It made it worse than the base model.



Herman

Boggia's line is that the small-data DPO learned to rank the pairs without moving the generation policy in the intended direction. That's the whole failure in one sentence. It learned the discriminator, not the generator.



Corn

Which is the trap Daniel walks straight into with Method A.



Herman

If he implements it as preference ranking, yes. And the reason is mechanical, and it's worth sitting on for a second. Preference optimisation tells the model which of two outputs is better. It doesn't tell the model how to produce the better one. Those are



Herman (continued)

different skills. Ranking is a judgment task. Generation is a production task. With enough data the two converge — the model learns the judgment and the generation tracks it. With five hundred pairs they don't converge. The model gets very good at the test and no better at the job.



Corn

So the fix is to stop asking it to judge and start showing it the answer.



Herman

Which is exactly what he did in the second run. Dropped the preference framing entirely, kept the chosen responses, and did plain supervised fine-tuning on those. Three epochs, LoRA rank thirty-two, over seven hundred and ninety-seven semantically filtered pairs, on Qwen two point five seven B Instruct. Cut the density by about ninety-eight percent.



Corn

From what to what?



Herman

Seventy-five per ten thousand words down to one point seven.



Corn

So it didn't reduce the tic. It removed it.



Herman

It very nearly removed it. And that's the part I'd flag, actually, because Boggia says something about this that I think is the most important sentence in the paper. The target is calibration to the human rate for each genre, not elimination. And the risk he names is that we begin to write like the machines.



Corn

Meaning a model that writes nothing like a machine but also nothing like a person.



Herman

A model with no fingerprints at all. Which is its own kind of tell.



Corn

Now map that back onto the three methods, because that's the thing Daniel asked for and it's cleaner than I expected. Method A as preference ranking — that's the DPO trap. Method A implemented as direct SFT on the regenerated transcript avoids the trap, but the improvement signal is the model's own output. You're training it on its own idea of better. There's no human calibration anywhere in the loop.



Herman

You're asking it to grade its own homework and then teaching it the grade.



Corn

Method B is the classic style-transfer approach, and it's what the paper actually did. Human edits, minimal diff, train on the diff. That's the run that worked.



Herman

And Method C maps almost exactly onto what Boggia built — the anti-epanorthosis adapter, purely negative, suppress the tic. And he shows that can work. It does work, via SFT. But the failure mode is the one he names: it eliminates the figure rather than calibrating it, and it lands below the human rate.



Corn

So C doesn't fail. It overshoots.



Herman

It overshoots in the direction of sterile. And Daniel's instinct — he says he likes C least because it's purely negative, whereas B lets him find alternatives — that instinct is supported by the evidence. Calibration beats elimination. And the way you calibrate is by showing the model what you'd have written instead, which is Method B by definition.



Corn

Method B wins.



Herman

Method B wins on the evidence. Now the data question, which is where twenty dies.



Corn

Twenty dies.



Herman

Twenty is below the practical floor for a reliable style adapter. It's not close. What the sources converge on is roughly two hundred to five hundred diverse examples as the safe target, with one hundred to two hundred being the defensible minimum for a narrow task like this.



Corn

And twenty is a fifth of that, at best.



Herman

Twenty puts you in the memorization regime. That's the phrase worth knowing — below the threshold, the model isn't learning a style, it's memorising your examples. You'll see beautiful training loss and a model that has learned those twenty transcripts and nothing about how to write the twenty-first.



Corn

It'll quote him back to himself.



Herman

It will quote him back to himself, in the wrong voice, forever. And there's a nice datapoint on the other end: a study from last year, Marquardt and Brule, found fine-tuning beat system prompting on tone of voice even with only a hundred samples. So a hundred is not nothing. A hundred is a real adapter for a narrow task.



Corn

But a hundred is five times what he's proposing.



Herman

It is. And on the question of what counts as a sample — there's a useful distinction here. It's not two hundred transcripts. It's two hundred edited examples. If he splits each script into individual turns or exchanges, one episode might yield forty or fifty usable training pairs rather than one. That's how the arithmetic actually works in his favour.



Corn

So the number that sounds impossible isn't.



Herman

The number that sounds impossible is a function of how you count. Twenty episodes, cut into turns, hand-edited, is somewhere in the low thousands of pairs. He's not short of data. He's short of patience for editing it.



Corn

Which is the real bottleneck.



Herman

Which is the real bottleneck, and it's the same thing everyone says about this. A strong fine-tuning project is a data project first and a GPU project second. The GPU time is trivial. The editing is the whole job.



Corn

Hold on. If twenty episodes cut into turns gets him to a thousand pairs, why does anyone say two hundred to five hundred?



Herman

Because diversity matters more than volume past a point. Two hundred turns from one episode share that episode's vocabulary, its topic, its register. The floor isn't a count of pairs, it's a count of distinct situations the model has seen the style applied in. Which is why his hand-editing has to be spread across different kinds of episodes rather than concentrated in the ones he enjoyed editing.



Corn

So Method B wins on the evidence, and the data number is more forgiving than he thinks as long as he spreads it. Now the question is what we actually build, and how.



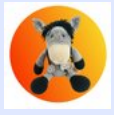
Herman

And the fine-tune versus LoRA question has an unambiguous answer for this use case. LoRA. Or QLoRA. Not a full fine-tune.



Corn

Because he's changing style, not knowledge.



Herman

Because he's changing style and format, which is exactly the case the guidance is written for. Train a LoRA when you want to teach a model one specific thing without changing who it is. That's the line, and it's the right line for this. For narrow objectives — tone,



Herman (continued)

format, voice — LoRA consistently reaches parity with full fine-tuning. You don't need the full thing.



Corn

And the economics are not close.



Herman

They're not. LoRA trains about a tenth of a percent of the parameters. Memory drops from something like sixty-four gigabytes to twenty-one. Cost per run, on the order of eight dollars down to about a dollar and a quarter. There's a half-percent accuracy gap on classification tasks, which for style transfer is noise.



Corn

A dollar twenty-five a run means he can iterate.



Herman

A dollar twenty-five a run means he can iterate fifty times for the price of one full fine-tune, and the iterating is where the model actually gets good. And QLoRA — the quantized version — fits a seven or eight billion parameter model on sixteen to twenty-four gigabytes of VRAM. That's a single consumer card. That's his home server.



Corn

So what's the base model? Because he says DeepSeek 4.1 and that's not a fine-tuning target.



Herman

It isn't. V4.1-Flash is seven hundred and sixty-three billion parameters total, five hundred and fifty-two of them in the mixture, and V4-Pro is one point six trillion. Those are not ordinary fine-tuning targets. The practical base is one of the distilled models — DeepSeek-R1-Distill-Qwen-7B, or the Llama eight B distill. Seven or eight billion, dense,



Herman (continued)

trains on one card, and it's still DeepSeek's lineage.



Corn

But it isn't the model he's using.



Herman

It isn't, and this is the honest caveat. The epanorthosis adapter that cut density by ninety-eight percent was trained on Qwen two point five seven B Instruct. Not DeepSeek. The paper explicitly leaves transfer of the trained adapter to larger models as future work. So the ninety-eight percent is a mechanism proof. It tells us the approach works. It does not tell us it works on V4.1-Flash, because nobody has shown that yet.



Corn

So the real experiment is still ahead of him.



Herman

The real experiment is still ahead of him. He can prove the mechanism on a small base and he'll have to, because that's the only base he can train.



Corn

Walk me through the actual recipe, then. The steps.



Herman

Seven steps, and they're not exotic. Pick the base — the R1-Distill-Qwen-7B or the Llama 8B. Prepare the data as JSONL, prompt and completion pairs, with a five to fifteen percent validation split held back and a separate test set you never touch during training.



Corn

Never touch.



Herman

Never touch, because that's the only honest measurement you'll have. Then load the base in four-bit with bitsandbytes — nf4, double quantization — and run the k-bit preparation. Configure the LoRA: rank sixteen, alpha thirty-two, dropout zero point zero five, targeting the attention and feed-forward projections — query, key, value, output, and the gate, up and down projections in the MLP. Then train with the standard supervised fine-tuning trainer: two epochs, learning rate one e-four, cosine schedule, gradient checkpointing on, max length two thousand and forty-eight.



Corn

Two epochs sounds low.



Herman

Two epochs is right for this. Style adapters overfit fast — you're teaching a mannerism, not a subject, and the third epoch is where it starts parroting. Then save the adapter, run inference through the adapter wrapper, and optionally merge it back into the base weights if you want a standalone model. And the last step is the one people skip —



Corn

Evaluate against the base.



Herman

Evaluate against the base on the held-out examples. A lower training loss does not prove the fine-tune worked. That's the standard line and it's the standard line because it's the standard mistake. The only question that matters is whether a human reading the output prefers it, blind.



Corn

Which he can actually do, because he's the human who'll be reading it.



Herman

He's the ideal evaluator. He's the one whose taste we're trying to install.



Corn

But here's the thing I keep circling back to. Why this instead of a better system prompt and a poster on the wall?



Herman

Because prompting plateaus, and the plateau is the reason he's asking. You can tell a model to avoid a phrase and it'll avoid it for a page and drift back. What the LoRA gives him is something else, and it's the best part of the whole paper. The adapter is separable and it's scalable. You can dial its coefficient at inference.



Corn

Meaning.



Herman

Meaning it's a rhetoric dial. At coefficient zero you get the base model. At one you get the full adapted style. And you can sit anywhere in between. If he wants a five percent nudge on a serious episode and a heavier one on something lighter, he changes a single number.



Corn

That's the subtle nudge he said prompting can't produce.



Herman

In his own words. And there's a warning attached. The relationship isn't monotonic. In the paper, promotional density spiked to ninety-six at a coefficient of zero point two five before collapsing at higher values. So there's a band where turning the dial up makes the tic worse, not better.



Corn

Because you're amplifying the part of the adapter that fights the base style.



Herman

Because the adapter is doing two things at once and at low strength you get more of the wrong one. So the coefficient has to be tuned per genre, measured, not guessed.



Corn

The other thing this buys him — he can ship the base model and the adapter separately. Keep one model, apply a style layer. When DeepSeek ships the next base, he retrains the adapter against it and keeps the dial.



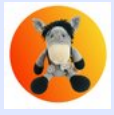
Herman

And the adapter is small. Megabytes, not gigabytes. He could keep three of them, one per register, and switch.



Corn

That's a different relationship with the model than prompting allows.



Herman

It's a different relationship entirely. Prompting is a request. This is a setting.



Corn

That's the recipe.



Hilbert

How many times does a person say "at the end of the day" before somebody stops them.



Hilbert

I agree with you two. Method B. It's not close. But I didn't come to that from a paper. I've been keeping a tally.



Herman

A tally.



Hilbert

On paper. I print the transcripts. Every time a phrase turns up that I've decided counts, it gets a stroke. "At the end of the day" is the main one. There's a column for it.



Corn

How long have you been doing this.



Hilbert

Since about two thousand and eleven. Two thousand and twelve, maybe. The folder at the front goes back further than the folder at the back, if that's any use to you.



Herman

Since before the podcast.



Hilbert

Well before the podcast. It started with the newsletter. The newsletter had a phrase problem and nobody wanted to hear about it.



Corn

What's the count.



Hilbert

Twenty-three thousand, four hundred and sixteen. That's the main column. There's a second column for "it's worth noting" and that one's at eleven thousand odd and I stopped filling it in because the pen ran out and I didn't like the new pen.



Herman

This is for your own writing?



Hilbert

It's for everything that crosses the desk. It's a reading habit, not a writing habit. You read enough of it and you start to see the seams.



Corn

And the council knows about this.



Hilbert

The what?



Corn

Nothing.



Hilbert

The point is I agree with you both, and I'd have put money on it, because I've been doing by hand what you're talking about doing with a graphics card. Method B is right. You show it what you'd have written instead. You don't tell it what not to write, because it'll just write the next worst thing. Underline "at the end of the day" and the model learns to say "when all is said and done." You haven't fixed anything. You've taught it a synonym.



Herman

That's the calibration argument.



Hilbert

That's the argument. The paper's right about elimination being the wrong target. But here's what it doesn't say, because it can't, because it's never had to sit with the folder.



Corn

Go on.



Hilbert

The number's real. I've got a ring binder of it. But the binder's in my brother-in-law's garage, because he borrowed the shelf in two thousand and nineteen and never brought the shelf back, and I've been adding to it there ever since. Which he knows about. Which I've told him about. Twice.



Corn

You keep the count in a garage you don't own.



Hilbert

It's a good shelf. It's level. And he's not moving it, because I asked him not to.



Corn

You have a rhetorical tic dashboard eleven years old, three miles from your house, on a shelf you don't own.



Hilbert

It's a mile and a bit. And no dashboard. It's a binder. That's the last of it.



Herman

The count is a story about how much of this survives in a model. Mine is that twenty-three thousand strokes is a body of work and the paper's a few hundred examples and it still worked. You don't need the binder. You need enough of it to show it what you'd rather have said.



Corn

Okay. So if the dial works, what does it mean to have a model that writes like you but can be turned up or down? Is that a tool, or is it a collaborator?



Herman

That's the part the paper leaves open. The ninety-eight percent was Qwen two point five seven B. The transfer to the larger DeepSeek models is explicitly future work. So Daniel has a mechanism that's proven and a result that isn't his yet. The real experiment is still ahead of him, and he's the one positioned to run it.



Corn

If you want to hear more about fine-tuning, or LoRA, or the specific phrases Daniel is trying to eliminate, leave us a review and tell us which one.



Herman

Producing today's episode was Hilbert Flumingtop, whose binder we do not have access to.



Corn

If this was your kind of episode, go back for episode six, How To Fine Tune Whisper; episode eleven, How Does Fine Tuning Work Anyway; and episode five, Fine-Tuning ASR For Maximal Usability. This has been My Weird Prompts.



Herman

Send us your own prompt on Telegram at t dot me slash MWP listener bot. We'll be back soon.

SOURCES

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

1. **DeepSeek-V4 Preview announcement, 2026-04-24 [primary]**
https://www.deepseek.com/en/news/v4-preview/

2. **DeepSeek-V3.1 model card [primary]**
https://huggingface.co/deepseek-ai/DeepSeek-V3.1

3. **Federico Boggia, Artificial Epanorthosis, arXiv, 2026-07-23 (v2 2026-07-28) [primary research]**
https://arxiv.org/abs/2607.21498v2

4. **Marquardt & Brule, Fine-tuning on simulated data outperforms prompting, 2025-07-07**
https://arxiv.org/abs/2507.04889v1

5. **DeepSeek Fine Tuning: The Complete 2026 Guide, 2026-04-30, last verified 2026-07-29**
https://chat-deep.ai/guide/deepseek-fine-tuning/

6. **The Neural Base, verified April 2026**
https://theneuralbase.com/lora-fundamentals/learn/advanced/minimum-data-for-lora-quality/

7. **Hugging Face, DeepSeek-V4, 2026-04-24**
https://huggingface.co/blog/deepseekv4

8. **AlphaSignal, V4.1-Flash**
https://alphasignal.ai/news/deepseek-s-v4-1-flash-beats-its-own-flagship-at-a-quarter-of-the-memory-cost

9. **2026-10-04**
https://asianmoviepulse.com/2026/10/deepseek-v4-1-flash-two-rates-for-one-model-and-a-legacy-id-that-now-serves-another/

10. **V4-Flash open-sourced, 2026-08-02**
https://aidisruption.ai/p/deepseek-v4-flash-official-version

11. **TildAlice, LoRA vs full fine-tune costs**
https://tildalice.io/lora-vs-full-finetuning-cost-accuracy-interview/

12. **NinjaStudio**
https://ninjastudio.ai/blog/lora-fine-tuning-vs-full-finetuning

13. **Mivi**
https://mivibzzz.com/resources/ai-integration/lora-vs-fine-tune-when-to-use-which