Episode Book

EPISODE #6692

How Fixed-Size AI Models Actually Fill Up

PUBLISHED	19 November 2026 at 22:16 UTC
DURATION	29:02
LISTEN	myweirdprompts.com/episode/fixed-size-model-capacity-21224/
HOSTS	Corn, Herman, Hilbert
SCRIPT MODEL	z-ai/glm-5.3
VOICES	omnivoice
PIPELINE	V5.36
SOURCES	15 cited by the research (listed at the end)
BOOK GENERATED	19 November 2026 at 22:18 UTC

SYNOPSIS

A neural network's file size never grows during fine-tuning, yet its capacity isn't a fixed jar. Research by Zhang and colleagues showed networks can memorize 100% of random CIFAR-10 labels, while Chen, Chang, and Chaudhari found that useful learning capacity is only about 2% of total weights and shifts with sample count. The episode also unpacks double descent from Nakkiran and colleagues, where adding 4.5 times more clean translation data actually made a transformer perform worse.

TRANSCRIPT



Corn

Last episode, Daniel, we established your Parakeet file never grows because fine-tuning only rewrites the numbers already in it. Now you've come back with the obvious next question, and I mean obvious as a compliment: if the model is a fixed set of numbers, can it fill up? Is there a point where it has learned all it can hold, and more of your voice just sloshes over the sides?



Herman

And he's specific about the stakes. A million hours of his own voice into a speech model. Billions of examples into a classifier for one task. Does more data eventually stop helping, and how would he even know he'd arrived?



Corn

He also wants to know whether the limit, if it exists, is set by the parameter count, or whether it's just diminishing returns from the data side. Which sounds like a small distinction.



Herman

It's the whole episode, actually. Because the answer is stranger than "yes, there's a wall." The wall isn't where you'd expect it, and in some regimes more data makes the model worse before it makes it better.



Corn

Worse. Not slower, not more expensive. Worse.



Herman

Measurably worse, on a clean dataset, in a published result. We'll get there.



Corn

Then let's get there. What's the naive picture, so we can watch it fall apart?



Herman

The naive picture is a jar. The model has, say, eighty million parameters, so it's a jar that holds eighty million facts. You pour in data, fact by fact, and once the jar is full, everything else spills. Every additional hour of your voice is wasted compute, maybe worse than wasted, because it's sloshing around displacing something.



Corn

It's a very natural picture. The file is a fixed size, the list of numbers is a fixed length. Storage that never grows must have a capacity.



Herman

And the research breaks that picture in two directions at once, which is the part I love. Direction one: a network can memorize far more than you'd think it could hold. Direction two: the useful part of what it holds is a tiny sliver of the weights. Both are true simultaneously.



Corn

Those two can't both be true in the jar picture. The jar picture says one number: how many facts fit.



Herman

Because the jar picture has no concept of the difference between storing something and using it. The actual research splits that into memorization capacity on one side, learning capacity on the other. And the second one, the one that matters for Daniel's ceiling question, turns out not to be a property of the file at all. It moves.



Corn

It moves with what?



Herman

The data. Which we'll get to. I don't want to spoil the hinge before we've earned it.



Corn

You've been sitting on that sentence like it's a chess move.



Herman

I have, a little.



Corn

Fine. Start with the memorization side, the too-big jar.



Herman

Twenty-sixteen paper, Zhang and colleagues, "Understanding deep learning requires rethinking generalization." They did something beautifully rude. They took standard image classifiers, AlexNet and a small Inception, trained them on CIFAR-10 with the labels completely randomized. Cat, truck, frog, all reassigned by coin flip. Pure noise. There is nothing to learn. Any sensible theory of capacity says the model should hit a wall somewhere and plateau below a hundred percent.



Corn

And it didn't.



Herman

A hundred percent training accuracy. Perfect memorization of random labels. They even replaced the images themselves with unstructured noise and it still fit everything. Then on the random-label test set, the same model scored about nine point nine percent, which on ten classes is a coin.



Corn

So it's a perfect parrot and a useless oracle.



Herman

Exactly the split. The same weights, one behavior on data it has seen, chance on everything else. And there's a theoretical result in the same paper that makes it concrete: a two-layer network with just two-n-plus-d weights can represent any function on n samples in d dimensions. Any labeling whatsoever of your training points.



Corn

Wait on that for a second. Any labeling. So the number of facts a fixed set of numbers can encode isn't bounded by the count of numbers in any simple way.



Herman

Not even close. The number of distinct labelings of n points is astronomically larger than n, and a network with weights on the order of two-n can realize all of them. The file is a fixed set of numbers, yes, but that set is a compressed encoding scheme with combinatorial reach, not a row of pigeonholes.



Herman

Think of it like this. Twenty switches can only be in a million-something configurations if you read them one at a time, but the number of distinct patterns they can jointly express is two to the twenty. The weights aren't slots holding one fact each. They're a system of dials where the meaning lives in the relationships between settings, not the settings themselves. A small change in one dial recontextualizes what a hundred others are doing. That's how a fixed-length list of numbers gets astronomical expressive reach.



Corn

So when Daniel says "it can't grow, can it fill up," the first correction is that even if it filled up in some sense, full is a much bigger place than the parameter count suggests.



Herman

Much bigger and much stranger. Now the other direction, because this is where the ceiling question actually lives. Twenty twenty-three, Chen, Chang and Chaudhari, a paper called "Learning Capacity." They borrow the formalism from thermodynamics, of all places, where the number of training samples plays the role of inverse temperature, and they define an effective dimensionality of a trained model.



Corn

Effective dimensionality meaning what, in words a sloth can hold?



Herman

How many directions in the model's parameter space the training actually pushed. Out of, say, all the weights, how much of that space did the data actually occupy and use. And their finding across many deep networks on typical datasets is that the learning capacity is roughly two percent of the total weights.



Corn

Two percent. The jar is bigger than the label says and you're only allowed the rim.



Herman

That's the tension. Raw storage vastly exceeds the parameter count, useful capacity is about two percent of it. Both measured. So parameter count tells you almost nothing on its own. It's not an upper bound on memorization and it's not a predictor of useful capacity.



Corn

I want to press on the two percent, because it sounds like a design flaw. Why would the network use so little of itself?



Herman

I don't think anyone has the complete answer. Part of it is that a lot of weights encode the same underlying feature redundantly, so they're not independent dimensions. The paper's framing is that capacity, as they define it, "depends upon the number of samples used for training." Read that twice. The capacity of the model depends on how much data you trained on. It is not a static property of the file.



Corn

That's the sentence you were sitting on.



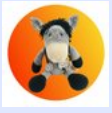
Herman

That's the sentence. Because if capacity moved with data, then Daniel's ceiling question has a different shape than he thinks. The wall isn't a fixed number you crash into. It's a threshold that relocates depending on where you are on the data curve.



Corn

All right, then walk me through what happens as you add data. From tiny amounts upward.



Herman

Two regimes, and they found what they call a freezing phenomenon. At very small sample counts, the learning capacity saturates. And here's the strange part: models with different architectures and different parameter counts, trained on the same small dataset, come out with the same learning capacity.



Corn

Same capacity. A big model and a small model.



Herman

Indistinguishable, in that regime. The architecture is irrelevant because the data is the binding constraint. Their guidance for that regime is blunt: buy more data. Nothing else you do matters.



Corn

Which flips at some point.



Herman

At a threshold sample count, yes. Above it, in the large-N regime, test loss falls off as one over N. Diminishing returns, the classic shape the scaling-laws literature also sees. And in that regime, the guidance inverts: more data is buying you almost nothing, so you search for a different architecture. There's even a counterintuitive corollary, that holding data fixed, the model with the smaller learning capacity gets the better test loss.



Corn

Because the big-capacity model is spending its room memorizing.



Herman

That's the interpretation I'd offer, yes. The room is being spent on the training set itself rather than the structure behind it.



Corn

So the answer to "is it a real limit or just diminishing returns" is... both, sequenced. Below the threshold it's not a limit at all, it's a data shortage. Above it, it's diminishing returns with a vengeance.



Herman

And the threshold is the thing worth knowing, because it's measurable. Their line is that the transition "provides guidelines as to when one should procure more data and when one should search for a different architecture." That's the fork. Daniel's question, "how would I know I've maxed out," is really "which side of the fork am I on."



Corn

Can you actually measure it from a training log, or is this a lab-only thing?



Herman

The Chen paper estimates capacity with Monte Carlo methods over a partition function, which is not something you do on a Tuesday evening. But the signature is visible indirectly. Training error dropping to zero while validation still improves means you're not yet in the frozen regime. Once training error is at zero and validation has flattened and additional data doesn't move it, you're past the threshold for that architecture.



Corn

Past it for that architecture at that data scale. You keep adding qualifiers.



Herman

Because they're load-bearing. The threshold isn't a property of the parameter count. Change the architecture, or the model size, and the threshold moves. The ceiling is real but it's a ceiling you partly chose.



Corn

The picture I remember defending there was basically the jar with adjustable shelves. I'm downgrading myself to a jar with a fog problem.



Herman

The shelves are adjustable and the jar doesn't have a fixed volume. And there's a fog problem, yes. Also a second fog, which I haven't mentioned yet, and it's the rude part of the literature.



Corn

Go on.



Herman

There's a regime where adding data doesn't just stop helping. It hurts. Test performance gets worse, then eventually better again, as you add more.



Corn

That's not diminishing returns. That's negative returns with an apology later.



Herman

The phenomenon is called double descent. Nakkiran and colleagues, the paper's associated with OpenAI, published December twenty nineteen. They documented it on three axes: model size, training duration, and sample count. The sample-wise one is the one for Daniel. They trained a transformer on language translation with no added label noise at all, a clean task, and for intermediate model sizes, training on four point five times more samples made test performance worse.



Corn

Four and a half times more data. Clean labels. Worse results.



Herman

Four point five times more data, clean labels, worse results. And the reason is structural. More samples do two things: they shift the whole error curve down, which is good, but they also shift the interpolation threshold to the right, which moves the error peak, and for those intermediate models the second effect wins.



Corn

Define interpolation threshold before we go further, because it's the one piece of jargon in this that Daniel can actually act on.



Herman

It's the point where the model is just barely able to fit the entire training set. Operationally, the maximum number of samples on which the training procedure achieves roughly zero training error. You can watch it approach in a training log.



Corn

And the error peak sits there. Deliberately.



Herman

Systematically. Their phrase is that the peak of test error appears "when models are just barely able to fit the train set." The intuition the OpenAI authors offer is that at that threshold there's effectively only one model that fits the training data, and if the labels contain any noise or any misspecification at all, fitting them destroys the model's global structure. There is no good model that both memorizes this set and generalizes.



Corn

And on the far side of the threshold?



Herman

Past it, you're over-parameterized. Now many models fit the training set, and some of them are good, and the training procedure finds the good ones. For reasons they openly admit they don't yet understand. I want to flag that honestly, the discoverers call the mechanism an open question. And a follow-up paper notes not every dataset and model pair even shows two descents. Some show one, some show three or more.



Corn

So it's a real, repeated, measured phenomenon whose explanation is still a shrug in a lab coat.



Herman

That's the accurate summary, yes.



Corn

One thing I'm going to be careful about for Daniel's case specifically. His million hours of one voice is clean data, and he'd be fine-tuning, not training from scratch. Everything we've cited is from-scratch runs.



Herman

Both caveats are real. The double descent peak is most prominent with label noise; without it, the peak is smaller and easy to miss. And nobody has published double descent measured on a fine-tune. So for Daniel I'd treat it as a shape to be aware of, not a scheduled event. The transferable insight is the location of the peak, not the promise that he'll see one.



Corn

But say he did suspect it. How would he even tell a double descent dip from ordinary training instability? A bad learning rate can make things worse too.



Herman

The discriminator is that double descent is monotonic in data. It tracks the sample count, not the step count. If he holds data fixed and varies only training duration, ordinary instability shows up as wobble in the curve. Double descent in the sample-wise form shows up when he retrains from scratch with a bigger slice of the same corpus and gets a worse model than a smaller slice produced. If more data is reliably worse, and then reliably better again past some volume, that's the signature. If it's worse at every volume, he's just broken something in the pipeline.



Corn

Can I flip your jar metaphor on Daniel's scenario, though? Because the million hours case isn't really fresh data at all.



Corn

Go ahead.



Herman

A million hours of your own voice is a small amount of information repeated enormously. It's one speaker, one accent, a bounded set of acoustic conditions. In the language-model literature that's the derived-data problem, and it now has a measured answer. A paper this year, Qin and colleagues, introduces a token-effectiveness function, eta, which is the value of a derived token, from repetition or paraphrasing, relative to a fresh one. They fitted it across models from fourteen million to six hundred million parameters.



Corn

And the value of a repeated token is what, half? Some discount rate?



Herman

That's the finding: it's far from constant. It depends jointly on model size, on the tokens-per-parameter ratio, and on how much derived data you already have. And it saturates as the corpus expands. The tenth repetition is worth less than the second, and the shape of that decay depends on the model.



Corn

So for the million-hours question, the honest answer is that the ceiling isn't an hour count. It's a function of the model size and how many effective tokens per parameter your voice represents.



Herman

Which is why the same recording set can be plenty for one model and starved for a bigger one. I'd say that's the single most practically useful thing we can hand him: he was asking "how much data until full," and the question that has an answer is "what's the effectiveness of my next thousand hours, given this model."



Corn

And that reframing has a consequence he might not like. If effectiveness depends on model size, then upgrading his model resets his saturation point. The corpus he already has stops being "enough" the day he moves to a bigger encoder.



Herman

The ceiling he's measured against his current model is not transferable. Every model change puts him back on the data curve at a new position.



Hilbert

Medieval Pottage Facts. Smoke from the hearth was used to cure meat and fish hung in the rafters, so the family breathed the same smoke all winter. You got the flavour either way.



Corn

Same smoke, two delivery mechanisms.



Herman

Where were we. Right, so Daniel's voice corpus saturates, but there's a lever on when that saturation bites, and it's a strange one. Sparsity. A paper accepted at ICML this year, Wu and colleagues, studied data-constrained training with repeated epochs. Sparse models, most of the weights switched off. Their headline is "delayed data saturation." Sparsity postpones the diminishing returns from repeated data.



Corn

The model with fewer active numbers absorbs more repetition before it gets bored.



Herman

That's a fair compression. They went up to nearly two billion parameters in the fitting set, sparsity up to about ninety-four percent, over two billion unique tokens repeated across sixteen epochs. And their framing is explicit, that sparsity is "not just a tool for efficiency, but a mechanism for improving scaling trade-offs under data scarcity."



Corn

Which is a sentence that cost someone four years of GPU time.



Herman

The resource trade-off they found is odd too. With fixed data, the loss-optimal sparsity is moderate, around fifty percent. Compute-optimal is sparser and grows with scale. So "how full is my model" depends even on how many of its own weights you've switched off.



Corn

Wait, that's almost paradoxical. Switch off most of the jar and the jar handles repetition better? Fewer active numbers, more patience with the same data?



Herman

It is counterintuitive until you remember what the two-percent finding told us. Most of the parameters weren't doing independent work anyway. Sparsity is, in effect, forcing the model to spend its capacity on structure rather than redundant storage, and a model that can't afford to memorize everything twice has less reason to get bored of the second pass.



Corn

Every knob we turn moves the ceiling. Model size, sparsity, data diversity, repetition. Daniel asked for a wall and we're handing him a weather system.



Herman

The weather system is the answer, though. There's one more trap in the "how would I know" question, and it's the nastiest one. Say he trains, and his validation curve goes flat. Flat for a long time. Every instinct says, maxed out, saturated, done.



Herman

And grokking. A model can sit at chance-level validation performance for a very long time, then abruptly generalize. The Huang paper from twenty twenty-four frames grokking, double descent and emergent abilities as one phenomenon, a competition inside the network between memorization circuits and generalization circuits. During the plateau, memorization is winning. Then the generalization circuits catch up and take over, and the validation curve falls off a cliff in the good direction.



Corn

So a flat validation curve is ambiguous. It might be the ceiling, and it might be the model still digesting.



Herman

That's why I'd never diagnose saturation from a plateau alone. The defensible sign is all three together: training error at zero, validation flattened, and, the decisive one, adding data no longer moves it. The last condition is the one that distinguishes "full" from "slow."



Corn

Although for Daniel there's a wrinkle in the other direction. His data is all one voice. Adding more of the same barely counts as adding data at all, given what we said about derived tokens.



Herman

Which pushes his practical test toward diversity, not volume. Hours in a new room, a new microphone, a different distance, are worth more than the next hundred hours in the same chair. His ceiling is partly acoustic coverage, not hours.



Corn

Now the honest gap. How much of this was measured on speech?



Herman

Almost none of it, and I want to say that plainly. No dedicated scaling-law study of ASR models turned up anywhere in what I looked at. The Zhang, Chen and double-descent results are image classifiers and language tasks. The token-effectiveness work is language models. For speech, everything we've said is reasoned by analogy from those, plus what we covered back when we talked about how much voice data a fine-tune actually wants, where the saturation curve was already visible in practice.



Corn

So the shapes are borrowed and the numbers aren't ours yet.



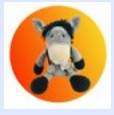
Herman

Daniel may be closer to the frontier than the literature is. He fine-tunes Parakeet, he publishes the results. The two-percent figure for a speech encoder is, as far as I know, unmeasured. That's a findable fact and nobody has found it.



Corn

He'd like that. He does like being the first through a door, even a small one.



Herman

There's a question I keep wanting to put to you, Corn, because you're the one in this flat who distrusts things that change while you're asleep, which is most of the day. If the ceiling on a fine-tune moves with every knob, model size, sparsity, data mix, is that a



Herman (continued)

comfort or a menace? You'd rather own a worse thing that stays the same.



Corn

Menace, obviously. A limit that relocates isn't a limit, it's a subscription.



Herman

That's your answer for the jar, then.



Corn

It's my answer for everything I own. But here's my honest version: I think the moving ceiling is actually the good news, and I distrust it anyway. If the wall were at the parameter count, Daniel's model would be finished, permanently, at a number printed on a card. Because it moves, everything about his fine-tune is still a decision. He just has to keep making it.



Herman

Which is either freedom or homework.



Corn

In my experience those are the same document.



Herman

Do you buy the two regimes for his case, though?. He's got a small model and a single speaker. Which side of the threshold is he on?



Corn

My guess is the small side, and comfortably so. A single speaker's acoustic inventory is small compared to what a Parakeet-class encoder has already absorbed from thousands of speakers during pretraining. He's not filling the jar with his voice. He's painting one stripe on a jar that came pre-filled.



Herman

That matches how I'd reason it. His fine-tune is a redirect of capacity the base model already has, not an expansion of it. The million-hours worry is real for training from scratch, and mostly theoretical for what he's doing.



Corn

Mostly. Your earlier point about coverage still stands. The stripe has to be the right stripe.



Herman

Marantz PMD six-sixty. Three hundred and forty pounds, from the shop on Jaffa Road, two thousand nine. I recorded every rehearsal of my choir on it for eleven years. Not for the choir. For the voice.



Hilbert

Mine. The plan was to have the machine speak the way I speak, for the archive. Sent the lot to a firm in twenty twenty-three, they cleaned the tapes and built a model from it. Two hundred pounds, non-negotiable, and after the first four hours of clean audio every extra hour I sent made no difference they could demonstrate. They still charged for the hours. Colin Prewitt, our second bass, held a low E for twenty-two seconds once and never let it go, and he is all over those tapes, and the model does not do him at all. The recorder itself is with my brother-in-law in Netanya. He says he's keeping it warm.



Corn

Twenty-two seconds is a long time for an E.



Hilbert

He measured it against the kettle. It was fully boiled.



Herman

Four hours before it stopped improving, that lines up with everything we've said. The repetition had nothing left to add.



Hilbert

They could have told me that at hour four. They told me at hour forty.



Corn

So the answer to Daniel's million-hours question exists, and it cost Hilbert two hundred pounds and a choir.



Herman

What I keep coming back to is that the firm's invoice and the research papers say the same thing in different dialects. Saturation is real, it arrives earlier than the customer expects, and nobody volunteers the number.



Corn

Then let's volunteer it. The last thing for Daniel, the practical checklist he actually asked for. How would he know.



Herman

Three conditions, and only the third one is decisive. Training error at zero. Validation flattened. And extra data, diverse data, changing nothing. If all three hold, he's past the threshold for that model at that data scale, and the move is not more hours, it's a different model or a different mix.



Corn

And if the flat stretch is only a week old, he owes it a grokking audit before he declares it a ceiling. Which he can do cheaply, because the model, unlike Hilbert's firm, doesn't bill by the hour.



Herman

If Daniel's listening, he's already checking his training logs, so we'll leave him to it.



Corn

One open thing to sit with, and it's Daniel's question turned around. "Have I maxed out" is never a yes or a no, because the ceiling isn't a property of the file. It's a property of the model, the data, and the choices he hasn't made yet. Which means every fine-tune he ships is an interim report on a ceiling he's still allowed to move.



Herman

Producing the show, as ever, Hilbert Flumingtop.



Corn

If you want more of this, try episode four, If Your Voice Ages, Does Your Fine-Tune Become Useless; episode eleven, How Does Fine Tuning Work Anyway; and episode five, Fine-Tuning ASR For Maximal Usability. This has been My Weird Prompts. If today's episode earned its half hour, the review box is right there, and it takes less time than a pre-grokking plateau.



Herman

Tell a friend with a fine-tune and too many hours of their own voice. We'll be back soon.

SOURCES

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

1. **Chen, Chang & Chaudhari, Learning Capacity: A Measure of the Effective Dimensionality of a Model, arXiv:2305.17332v2 [primary]**
   submitted 2023-05-27, revised 2024-10-20; retrieved 2026-11-19.
   https://arxiv.org/abs/2305.17332

2. **Nakkiran, Kaplun, Bansal, Yang, Barak & Sutskever, Deep double descent, OpenAI [primary]**
   published 2019-12-05; retrieved 2026-11-19. (paper: arXiv:1912.02292)
   https://openai.com/index/deep-double-descent/

3. **Zhang, Bengio, Hardt, Recht & Vinyals, Understanding deep learning requires rethinking generalization, arXiv:1611.03530v2 [primary]**
   retrieved 2026-11-19.
   https://arxiv.org/abs/1611.03530v2

4. **Qin, Hamidieh & Alvarez-Melis, Bridging Compute- and Data-Optimal Pretraining, arXiv:2607.25271v1 [primary]**
   submitted 2026-07-28; retrieved 2026-11-19.
   https://arxiv.org/abs/2607.25271v1

5. **Wu, Xiao, Okanovic et al., When Data Is Scarce: Scaling Sparse Language Models with Repeated Training, arXiv:2606.01155v1 [primary]**
   submitted 2026-05-31, accepted ICML 2026; retrieved 2026-11-19.
   https://arxiv.org/abs/2606.01155v1

6. **Huang, Hu, Han et al., Unified View of Grokking, Double Descent and Emergent Abilities: A Perspective from Circuits Competition, arXiv:2402.15175v2 [primary]**
   submitted 2024-02-23, revised 2024-02-26; retrieved 2026-11-19.
   https://arxiv.org/abs/2402.15175v2

7. **Nakkiran et al., Deep double descent: where bigger models and more data hurt, *Journal of Statistical Mechanics***
   retrieved 2026-11-19.
   https://iopscience.iop.org/article/10.1088/1742-5468/ac3a74

8. **Same paper, ICLR 2017 slides**
   retrieved 2026-11-19.
   https://iclr.cc/archive/www/lib/exe/fetch.php%3Fmedia=iclr2017:czhang_iclr2017.pdf

9. **Same paper, *Communications of the ACM***
   published 2021-02-22; retrieved 2026-11-19.
   https://dl.acm.org/doi/10.1145/3446776

10. **Google Research summary of the above**
    retrieved 2026-11-19.
    https://research.google/pubs/understanding-deep-learning-requires-rethinking-generalization/

11. **Double descent, Wikipedia**
    last edited 2026-08-28; retrieved 2026-11-19.
    https://en.wikipedia.org/wiki/Double_descent

12. **Grokking (machine learning), Wikipedia**
(machine_learning), last edited 2026-10-05; retrieved 2026-11-19.
https://en.wikipedia.org/wiki/Grokking

13. **More Data Can Hurt for Linear Regression: Sample-wise Double Descent, arXiv:1912.07242**
retrieved 2026-11-19.
https://arxiv.org/abs/1912.07242

14. **Double Descent Demystified: Identifying, Interpreting & Ablating the Sources of a Deep Learning Puzzle, arXiv:2303.14151**
retrieved 2026-11-19.
https://arxiv.org/abs/2303.14151

15. **Understanding the double descent curve in Machine Learning, arXiv:2211.10322**
retrieved 2026-11-19.
https://arxiv.org/abs/2211.10322