MY WEIRD PROMPTS

Podcast Transcript

EPISODE #118

AI in 2025: Is Small the New Big?

Published December 28, 2025 • Runtime: 20:59

https://myweirdprompts.com/episode/small-vs-large-llm-efficiency/

EPISODE SYNOPSIS

In this episode of My Weird Prompts, brothers Herman and Corn Poppleberry dive into a provocative thought experiment: if cloud inference costs were identical, would there ever be a reason to choose a small model over a trillion-parameter giant? Moving beyond the "bigger is better" hype of previous years, the duo explores the physical realities of latency, the hidden costs of model verbosity, and the rise of high-density models in 2025. Whether you are a developer looking for better throughput or a business leader seeking reliable specialization, this discussion reveals why the most powerful tool isn't always the largest one.

DANIEL'S PROMPT

Daniel

When comparing smaller parameter models to much larger ones that perform the same tasks, if the cost for cloud inference is the same and infrastructure is not a concern, should you always choose the larger model, or is there more nuance to the decision?

TRANSCRIPT

Corn

Hey everyone, welcome back to My Weird Prompts! I am Corn, and I am joined as always by my brother.

Herman

Herman Poppleberry, reporting for duty. We are coming to you from our home in Jerusalem on this fine December day in twenty twenty-five.

Corn

It has been a busy year for artificial intelligence, hasn't it? Our housemate Daniel actually sent us a prompt this morning that gets right to the heart of a debate we have been having in the living room lately.

Herman

Daniel always has a knack for cutting through the hype. His question is essentially a thought experiment about model size versus efficiency. He wants to know, if the cost for cloud inference is exactly the same and you do not have to worry about the infrastructure, should you always just pick the biggest model available? Or is there actually a reason to stick with a smaller model even when the big one is effectively free?

Corn

It is a great question because it challenges the bigger is better assumption that has dominated the industry for the last few years. You would think that if a trillion parameter model costs the same as a seven billion parameter model, you would be a fool not to take the extra brains. But as we have seen with the releases throughout twenty twenty-five, the math is not always that simple.

Herman

Exactly. It is a bit like asking if you would rather have a semi truck or a sports car to pick up a loaf of bread, assuming the fuel and insurance costs were identical. Sure, the semi truck can carry more, but do you really want to try and park that thing at the grocery store?

Corn

That is a perfect place to start. Let us dig into why bigger isn't always better, even when the price tag is the same. I think we should start with the most obvious factor that people often overlook when they are looking at benchmarks, and that is latency.

Herman

Oh, latency is the silent killer of user experience. This is something that gets lost in the noise of high level reasoning scores. When we talk about these massive models, we are talking about a physical reality of moving bits across a chip. Even in twenty twenty-five, with the latest optical interconnects and massive memory bandwidth improvements, a model with hundreds of billions of parameters has to move a massive amount of data from the high bandwidth memory to the processors for every single token it generates.

Corn

Right, so even if the cloud provider is subsidizing the cost or has some incredible hardware that makes the price equal, the laws of physics still apply to the inference speed. If I am building a real-time chat application or a coding assistant that needs to suggest lines as I type, I cannot wait three seconds for the model to think.

Herman

Precisely. A smaller model, say something in the eight to ten billion parameter range, can often reside entirely within the cache or at least require far fewer memory read cycles. We are talking about the difference between a hundred tokens per second and maybe ten or fifteen tokens per second for a massive dense model. For a human reading on a screen, that is the difference between an instantaneous flow and a stuttering experience that feels like waiting for a slow typist.

Corn

That makes sense for the generation speed, but what about the time it takes to even start talking? I have noticed that with some of the larger models we have tested this year, there is a noticeable lag before the first word even appears.

Herman

That is the time to first token. It involves the initial processing of your prompt, the pre-fill stage. Larger models have much more work to do during that phase. If your application requires a snappy, interactive feel, that initial half-second delay in a large model can feel like an eternity compared to the near-instant response of a highly optimized smaller model.

Corn

So, point one for the smaller model is speed. But let us look at the other side. People usually go for the big models because they are smarter, right? They handle complex reasoning better. If I am asking it to solve a difficult architectural problem or write a complex legal brief, surely the latency is a fair trade-off for accuracy?

Herman

Generally, yes. But here is where the nuance Daniel mentioned really kicks in. We have seen a massive trend this year toward what people are calling high-density models. Thanks to better training data and techniques like knowledge distillation, a twenty billion parameter model today can often outperform a hundred billion parameter model from two years ago.

Corn

Right, the Chinchilla scaling laws taught us that most models were actually under-trained for their size. We are now seeing models that are smaller but have seen way more high-quality tokens during training.

Herman

Exactly. So the question becomes, what is the task? If you are doing something that requires broad world knowledge, like trivia or general creative writing, the massive model wins because it has more storage for facts. But if you are doing a specific task, like converting natural language to database queries or summarizing technical documents, a smaller model that was fine-tuned or trained on high-quality synthetic data for that specific domain might actually be more reliable.

Corn

That is an interesting point. Is it possible for a large model to be too smart for its own good? I mean, does it ever over-complicate simple tasks?

Herman

Absolutely. We call it over-thinking or verbosity bias. Larger models have a tendency to be more flowery and include more caveats and unnecessary explanations. If you just need a yes or no answer, or a specific piece of data extracted in a rigid format, a smaller model is often easier to steer. It is more obedient to the system prompt because it has fewer competing pathways in its neural network.

Corn

I have definitely experienced that. You ask a simple question and the giant model gives you a five-paragraph essay including a history of the topic and three different perspectives you did not ask for. It is like asking a professor a simple question and getting a full lecture.

Herman

That is exactly it. And that verbosity actually increases your cost in the long run, even if the price per token is the same, because you are consuming more tokens for the same result. If a small model gives you the answer in ten tokens and the large model gives it to you in fifty, the large model is five times more expensive in practice.

Corn

Wait, that is a huge point. If we are talking about cost-per-inference being the same, we usually mean cost-per-million-tokens. But if the larger model is more wordy, the total cost per task actually goes up.

Herman

Bingo. You have to look at the cost per successful completion, not just the unit price. Plus, there is the issue of reliability. Larger models can sometimes hallucinate more complex errors. A small, focused model might fail, but it usually fails in predictable ways. A massive model might give you a brilliant-sounding answer that is subtly and dangerously wrong because it tried to connect two concepts that should not be connected.

Corn

This is fascinating. So even with a level playing field on price, we have speed, steering, and actual task-cost as advantages for the smaller model. But I want to pivot for a second. Let us take a quick break for our sponsors, and when we come back, I want to talk about context windows and how those change the math.

Herman

Sounds good.

Corn

We will be right back. Larry: Are you tired of your neighbors having better thoughts than you? Do you feel like your brain is running on twenty twenty-two hardware in a twenty twenty-five world? Introducing the Neuro-Sync Headband from Thought-Tech. Our patented bio-resonant frequencies align your alpha waves with the global intelligence grid, allowing you to finish your sentences before you even start them. It is not mind reading, it is mind leading. Users have reported a three hundred percent increase in confidence and an eighty percent decrease in the need for sleep. Warning, side effects may include hearing colors, temporary loss of your middle name, and a sudden craving for raw onions. But can you really put a price on mental dominance? Get your Neuro-Sync Headband today. Larry: BUY NOW!

Corn

Alright, thanks Larry. I think. I am not sure about the raw onions, but moving on. Herman, before the break we were talking about why smaller models might be better even if the price is the same. I wanted to ask about context windows. Usually, the big models have these massive million-token context windows. Does that give them a definitive edge?

Herman

It used to, but the gap is closing. In twenty twenty-five, we are seeing seven billion and fourteen billion parameter models with context windows of a hundred thousand tokens or more. But here is the technical catch that most people do not realize. Even if a model can take a million tokens, the computational cost of attending to those tokens grows.

Corn

You are talking about the quadratic complexity of the attention mechanism?

Herman

Exactly. Although many models now use linear attention or state-space models to get around that, there is still a massive memory requirement for what we call the KV cache. The KV cache stores the keys and values for every token in your conversation so the model does not have to re-process them every time. For a giant model, that cache is enormous. If you have multiple users or a long conversation, the memory pressure on the GPU becomes a bottleneck.

Corn

So, if I am using a smaller model, my KV cache is much smaller, which means I can handle more concurrent requests or longer conversations with less lag?

Herman

Precisely. It is about throughput. If you are a developer, you care about how many requests you can handle per second on a single piece of hardware. Even if the cloud provider charges you the same, they might throttle your rate limits on the larger model because it is eating up so much of their VRAM. A smaller model lets you pack more density into your application.

Corn

That is a great practical point. Let us talk about the quality of the reasoning again. One thing I have noticed this year is that for coding, the smaller models have become incredibly good. Why is that?

Herman

It is the density of the training data. Code is very logical and structured. You do not need a trillion parameters to understand the syntax of Python or the logic of a React component. You need high-quality examples. We have found that once you hit a certain threshold of parameters, maybe around thirty billion, you have enough capacity to understand almost any programming language perfectly. Adding more parameters after that just adds more knowledge about things that are not code, like nineteenth-century poetry or the history of the Ottoman Empire.

Corn

So for a specialized tool, the extra parameters are literally dead weight. They are just extra neurons that are not being used for the task at hand but still have to be powered and processed.

Herman

Exactly. It is like hiring a polymath who knows everything about everything to do your taxes, when you could just hire a really great accountant. The polymath might be more interesting to talk to, but they might also get distracted by the historical implications of your charitable donations.

Corn

I love that analogy. Now, let us look at the nuance from another angle. What about fine-tuning? It is much easier and cheaper to fine-tune a small model on your own private data than it is to tune a massive one.

Herman

This is a huge factor for businesses. If you take a seven billion parameter model and fine-tune it on your company's internal documentation, support tickets, and brand voice, it will almost certainly outperform a generic trillion-parameter model on tasks related to your business. It becomes a specialist. And because it is small, you can run it on-premise if you want to, or even on a high-end laptop, which gives you data privacy that you just cannot get with the giant cloud-only models.

Corn

And that brings up the edge computing aspect. We are in twenty twenty-five, and our phones and laptops now have dedicated NPU chips that are surprisingly powerful. If I have a choice between a large model in the cloud and a smaller model that runs locally on my device, the local model wins every time for privacy and offline availability.

Herman

And zero latency. You do not even have to wait for the round-trip to the server. That is the ultimate user experience. But even staying within the cloud inference scenario Daniel proposed, there is the issue of versioning and stability. Large models are often updated by the providers without much notice. They call it model drift. Your prompts that worked yesterday might stop working today because the provider changed the underlying weights of their massive model to be safer or more efficient.

Corn

And smaller models are easier to pin to a specific version?

Herman

Yes, because they are cheaper to host, providers are more likely to let you keep using an older version of a small model. Or, because the architecture is simpler, you can just download the weights and host it yourself on a standard instance. You gain a level of control and predictability that is very hard to achieve with the cutting-edge, monster models.

Corn

So let me try to summarize the case for the smaller model so far. Even if the price is the same, the smaller model gives you better latency, higher throughput, more predictable behavior, easier steering, and the ability to specialize through fine-tuning.

Herman

That is a solid list. But I want to be fair to the big models too. There are absolutely times where you should ignore the small ones and go for the biggest thing available.

Corn

Okay, let us hear the counter-argument. When does the big model win?

Herman

It wins on the edge of the unknown. If you are doing true zero-shot reasoning on a problem that has never been seen before, or if you need the model to connect ideas from two completely unrelated fields, the massive models have an emergent property that the small ones lack. We call it the spark of general intelligence. A small model is a great tool, but a massive model is a creative partner.

Corn

I see. So if the task is highly creative or requires deep synthesis of disparate information, the sheer number of connections in a large model becomes an asset rather than a liability.

Herman

Exactly. If I am brainstorming a sci-fi novel that involves hard physics, ancient linguistics, and complex political theory, the trillion-parameter model is going to give me much deeper and more interesting connections than a small model that has been optimized for efficiency. The small model will give me the tropes. The big model will give me something new.

Corn

That makes sense. It is the difference between a calculator and a brain. For most things in life, a calculator is better because it is fast and accurate. But for the things that matter, you want the brain.

Herman

I like that. And there is one more thing: the floor of failure. When a small model hits its limit, it tends to collapse into gibberish or repetitive loops. When a large model hits its limit, it often has enough broad knowledge to at least give you a plausible path forward or tell you why it is struggling. It has a more graceful degradation.

Corn

That is a very subtle but important point. So, if the cost is the same, your choice really depends on the stakes of the failure and the complexity of the reasoning.

Herman

Precisely. For a user-facing product where speed is king, small wins. For a research project where quality is everything and you do not care if it takes a minute to get an answer, big wins.

Corn

This really reframes the whole bigger is better narrative. It turns out that efficiency is not just about saving money, it is about the quality of the interaction.

Herman

It is about choosing the right tool for the job. We have spent the last few years obsessed with the ceiling of what AI can do. I think twenty twenty-five is the year we start focusing on the floor, making the everyday tasks as fast, reliable, and invisible as possible.

Corn

I think Daniel will be happy with that answer. It is not a simple yes or no, but a framework for making the decision. It is about realizing that parameters are a cost, not just a benefit, even if that cost is measured in time and complexity rather than dollars.

Herman

Well put, Corn.

Corn

We are coming to the end of our time today. This has been a really enlightening look at the state of AI in twenty twenty-five. I feel like I understand my own preference for some of these smaller, snappier models a lot better now.

Herman

It is all about that memory bandwidth, brother. It always comes back to the hardware.

Corn

Before we go, I want to thank Daniel again for sending in that prompt. It gave us a lot to chew on. If any of you listening have your own weird prompts or questions about the world of AI, technology, or anything else, we would love to hear them.

Herman

You can find us at our website, myweirdprompts.com. There is a contact form there, and you can also find our full archive and the RSS feed for the show.

Corn

And of course, we are available on Spotify. If you enjoyed the episode, please leave us a review or share it with a friend. It really helps the show grow.

Herman

We will be back next week with another deep dive into whatever Daniel or the rest of you send our way.

Corn

Thanks for listening to My Weird Prompts. I am Corn.

Herman

And I am Herman Poppleberry.

Corn

See you next time!

Herman

Goodbye everyone!

Corn

This has been My Weird Prompts, a human-AI collaboration. Stay curious.