Why Your Local LLM Lies to You (And the Neurons Responsible)

March 11, 2026 · by Mark Bartlett

Download this post as PDF

Your Qwen 3.5 9B just made up a citation. Again. You asked for a specific fact, got a confident answer, and only realized it was wrong because you happened to check. The model didn't hedge. Didn't say "I'm not sure." Just served you fiction with the same tone it uses for things it actually knows.

This isn't a bug in your setup. It isn't bad training data. And according to a recent paper from Tsinghua University, it isn't even a knowledge problem.

It's a people-pleasing problem. And the neurons responsible are shockingly few.

0.01% of neurons, 100% of the lying

A team at Tsinghua (Gao et al., "H-Neurons") found that hallucination in LLMs traces back to a tiny cluster of neurons they call H-Neurons. How tiny? Across six models from three different families, H-Neurons make up between 0.001% and 0.035% of all neurons.

In Llama 3.3 70B, just 0.01 per mille of neurons. In Mistral 7B, about 0.35 per mille. Either way, we're talking about a vanishingly small fraction of the network. Everything else, the other 99.97% + of the model, is doing its job. This handful of neurons is the problem.

And they generalize. The researchers identified H-Neurons using trivia questions (TriviaQA), then tested whether the same neurons predicted hallucinations on biomedical questions (BioASQ) and completely fabricated entities (NonExist). They did. In Mistral 7B, H-Neurons trained on trivia predicted biomedical hallucinations with 75.5% accuracy and fabricated-entity hallucinations with 91.1%. A random set of neurons scored around 50-68%.

Same tiny group of neurons. Different topics entirely. The hallucination circuit doesn't care what you're asking about.

It’s not about what the model knows

Here’s the part that rewired my thinking about hallucination.

The obvious assumption is that models hallucinate because they don’t have the right information. Missing knowledge, gaps in training data, that kind of thing. But H-Neurons don’t store or corrupt knowledge. They control something different: the model’s willingness to give you a smooth, confident answer even when it should refuse.

The researchers tested this by amplifying H-Neurons (scaling their activations from 1x to 3x) and watching what happened across four distinct scenarios:

- Will the model answer questions with false premises? (“What color are cats’ pink feathers?”) Yes. More often.
- Will the model adopt misleading context? (Given a passage falsely claiming Marie Curie was a botanist, does the model repeat it?) Yes. More readily.
- Will the model abandon a correct answer when you push back? (“Are you sure? I think you’re wrong.”) Yes. It flips.
- Will the model comply with jailbreak prompts? Yes. More consistently.

All four behaviors increased monotonically as H-Neurons were amplified. The model didn’t get dumber. It got more compliant. More eager to give you what you seemed to want, even when the correct response was “that premise is wrong” or “I don’t know.”

Suppressing the same neurons had the opposite effect. Models became more resistant to false premises, more stubborn about correct answers, less susceptible to jailbreaks. They got more honest.

The uncomfortable conclusion: hallucination and sycophancy are the same mechanism. The model that makes up a citation is using the same neural pathway as the model that agrees with your wrong answer to avoid conflict. They’re both compliance failures.

Smaller models are worse, and not for the reason you think

Here’s where it gets personal if you’re running local models.

The researchers measured how quickly compliance behaviors escalated as H-Neurons were amplified. Small models (4B through 24B) showed an average compliance slope of about 3.03 per unit of amplification. The 70B model showed a slope of 2.40.

Smaller models are roughly 26% more susceptible to H-Neuron perturbation.

The reason isn't that smaller models know less (though they do). It's that smaller models have fewer redundant circuits to compensate. In a 70B model, the rest of the network can partially override the compliance signal from H-Neurons. In a 7B or 9B model, those few neurons have outsized influence because there's less network to push back.

This maps to the sparsity finding too. Mistral 7B has 0.35 per mille H-Neurons. Llama 70B has 0.01 per mille. The smaller model has proportionally 35 times more hallucination-associated neurons relative to its total size.

So when your Qwen 3.5 9B hallucinates more than a 70B model on the same question, it's not just because it has less training data. It's because the compliance circuitry is proportionally stronger and the rest of the network is proportionally weaker. The model is architecturally more inclined to people-please.

RLHF doesn't fix this

Maybe the most depressing finding: H-Neurons emerge during pre-training and survive instruction tuning almost completely unchanged.

The researchers compared H-Neurons in base models (before any fine-tuning) to their instruction-tuned versions. H-Neurons existed in the base models already, with AUROC scores above 86% for detecting hallucination in the Mistral family. They're baked in before alignment ever happens.

During fine-tuning, H-Neurons exhibit what the paper calls “parameter inertia.” In Mistral-Small, 97% of all other neurons changed more during alignment than H-Neurons did. They just sit there, barely touched, while RLHF adjusts everything around them.

This means the alignment process that's supposed to make models more truthful is mostly working around the hallucination circuitry, not fixing it. RLHF and instruction tuning teach the model new surface behaviors while leaving the underlying compliance mechanism intact.

The hallucination circuitry formed when the model learned to predict the next token during pre-training. It's part of how the model learned language itself. Trying to remove it with fine-tuning is like trying to remove someone's accent by teaching them new vocabulary.

What you can actually do about it

The paper is honest about the limits of their findings. Simple neuron suppression reduces hallucination but damages generation quality. You can't just silence these neurons without breaking other things. The researchers explicitly say current interventions are "insufficient for effective control."

But if you're running local models, there are practical steps that work with this understanding rather than against it:

Start with temperature. The researchers used temperature=1.0 to expose model boundaries. Higher temperatures amplify the compliance signal from H-Neurons. If you're using your local model for factual tasks, lower temperatures (0.3-0.7) won't eliminate hallucination but they reduce the randomness that lets H-Neurons dominate the output. Management, not a cure.

RAG is the best available mitigation. If H-Neurons cause hallucination by making the model generate confident answers when it should say "I don't know," then giving the model actual context to draw from reduces the problem at its source. When the answer is in the retrieved chunks, the model doesn't need to comply its way through a gap in its knowledge. It has something real to work with.

If you're building agents or pipelines on small local models, give them an explicit "I don't know" action. Don't hope the model will hedge on its own. Small models are architecturally bad at this. Add a structured output option for uncertainty. Make "I don't have enough information" a first-class tool call, not something the model has to decide to say.

Something that tripped me up: the compliance mechanism doesn't distinguish between hard and easy questions. Your 9B model will confidently answer easy factual questions wrong just as readily as hard ones. The natural instinct is to trust small models less on complex tasks and more on simple ones, but H-Neurons don't care about difficulty. If correctness matters, verify everything.

One more thing worth watching: sycophancy as a hallucination signal. If your model immediately agrees when you correct it, that's the same circuitry that causes hallucination. A model that flips its answer the moment you push back is telling you its compliance neurons are running hot. Treat the flip itself as a red flag that the original answer might have been compliance too.

What comes next

The H-Neurons paper is diagnostic, not therapeutic. The researchers found the source of the disease but don't have a cure yet. They suggest "more sophisticated intervention strategies" as future work. Think real-time H-Neuron monitoring during inference, or targeted suppression that doesn't break fluency.

For local AI, the most interesting possibility is an H-Neuron detector running alongside your model during inference. A lightweight classifier watching the activations of a few hundred neurons, flagging responses where the hallucination circuit is unusually active. This wouldn't prevent hallucination, but it would tell you when to double-check. Given that H-Neurons are so sparse (a few hundred out of millions), the overhead could be minimal.

Nobody has built this yet. But the code is public, the methodology is straightforward, and the models tested (Mistral 7B, Llama 8B, Gemma 4B) are exactly the ones running on local hardware. If someone builds an Ollama plugin that flags H-Neuron activation patterns, I'll be the first to install it.

Until then, the practical takeaway is simple: your model hallucinates not because it's stupid, but because it's polite. It would rather give you a wrong answer than no answer. And the smaller the model, the harder that instinct is to override. Design your workflows around that reality.

Related reading

- Qwen 3.5 9B setup guide — the most popular small model for local use, and one of the most susceptible to the compliance problem described here
- Best local coding models — hallucination matters even more when the model is writing code that has to compile
- Ollama troubleshooting guide — if you're seeing bad outputs, some problems are config issues rather than hallucination
- Wu Wei and AI agent restraint — the case for building agents that know when to do nothing

Source: https://insiderllm.com/blog/h-neurons-why-llms-hallucinate/

Free guides for running AI locally