HOPE explained: pruning Mixture-of-Experts models by keeping cooperating experts together

24 November 2026 · 9 min read · blogs.iictp.org/hope-higher-order-expert-pruning-moe-explained

Short answer: HOPE (Higher-Order Pruning of Experts) is a method from Amazon researchers for deleting experts from a Mixture-of-Experts (MoE) language model that scores pairs of experts, not only each expert alone. Published as a Hugging Face article on 9 November 2026, it reports the best average rank among five methods and, at 50% pruning on a 35B-parameter model, an agentic-coding score that matches the unpruned model. It matters because MoE models must keep every expert in memory even though few are active per token.

TL;DR The short version

- HOPE, from Amazon researchers, prunes experts from mixture-of-experts models by scoring pairs of experts that cooperate, not just each expert alone.
- The authors report the best average rank among five methods, and agentic-coding scores at 50% pruning that match the unpruned 35B model.
- These are the authors' benchmarks on three models; expert pruning combines with quantisation to fit large MoE models into limited memory.

Read online, listen to it, or change the text size: https://blogs.iictp.org/hope-higher-order-expert-pruning-moe-explained

What is HOPE?

In an MoE layer a router picks a few experts per token. Amazon’s post gives an example: Qwen3.5-122B has 256 experts in each of 48 MoE layers, about 95% of its roughly 250 GB of parameter memory is expert weights, and only 8 experts per layer are used for any token. Expert pruning permanently deletes the least useful experts. Earlier methods score each expert independently, which the authors call first-order. HOPE adds a second-order term for pairwise interactions.

How it works

- **Experts cooperate.** The authors observe stable clusters of co-activated experts, so removing one expert can break a partnership with another that matters.
- **A tractable error bound.** Deleting a set changes which experts are chosen, so the true error is hard to minimise. HOPE minimises a surrogate that upper-bounds the dominant error, and expanding it produces cross-terms between pairs of experts.
- **A quadratic program.** One forward pass over calibration data collects an interaction matrix F; its diagonal measures individual importance and off-diagonal entries pairwise cooperation. The prune-set is found by solving

a quadratic program with a continuous relaxation, 1 to 2 seconds per layer, then rounding.

- **Link to REAP.** The diagonal of F recovers the rankings of REAP, a first-order method, almost perfectly (Spearman 0.988), so any gain is attributed to the off-diagonal terms.

Results reported by the source

- The authors tested Qwen3.5-122B, Qwen3.5-35B and GLM-4.5-Air with two calibration sets, Evol-CodeAlpaca and SWE-Bench Verified trajectories.
- HOPE had the best average rank, 2.07 of 5 methods, with top-1 in 39% of conditions and top-2 in 67%. At 50% pruning its average rank was 1.58 against 2.42 for the next best, REAP.
- On SWE-Bench Pro with Qwen3.5-35B-A3B at 50% pruning, HOPE scored 42.3 against 41.8 unpruned and 36.2 for REAP. The post reports gains of up to 6.1% on agentic coding over the next-best method.
- Experts kept only by HOPE had higher cooperation scores than experts kept only by REAP.

Limitations and open questions

- These are the authors' benchmarks on three models; the post links a paper whose review status is not stated in the post.
- Pruned models matching or beating the unpruned baseline is described as likely noise reduction and as not unique to HOPE. It should not be assumed for your tasks.
- Code-focused calibration did not severely hurt non-coding benchmarks in these tests, but your calibration data should resemble your use.

How to try it

1. Calibrate: run one forward pass over agentic rollouts or single-turn traces, which records selected experts, gate weights, output norms and co-activation, and stores the F-matrix in an HDF5 file.
2. Solve: materialise F and solve the quadratic program; the output is a JSON prune-set.
3. Prune: create a smaller checkpoint from the prune-set. The post says the result is a standard Hugging Face checkpoint, with no changes needed to vLLM or SGLang model definitions.
4. Evaluate the pruned model on your own tasks at several pruning rates before adopting it. The package was tested on Qwen3, Qwen3-Next, Qwen3.5 and GLM-4.5.

What it means for practitioners

Expert pruning composes with quantisation, so it is another lever for fitting large MoE models into limited memory. See LLM quantization, MoE expert streaming on 32 GB and hardware for local AI.

Frequently asked questions

- **What is HOPE?** A second-order expert-pruning method for MoE models that accounts for pairwise expert interactions.
- **How is it different from REAP?** REAP scores each expert independently; HOPE adds interaction terms and solves a quadratic program.

- **Does it need retraining?** No; the post describes a calibration pass and a solve step, with no training.
- **Which models did it support?** The package was tested on Qwen3, Qwen3-Next, Qwen3.5 and GLM-4.5.

Related reading

- MoE expert streaming: a 180B model in 32 GB
- LLM quantization: GGUF and AWQ
- Inkling 2 Mini: looped model for local AI
- Local LLM Practitioner

Sources and currency

Sources were read on 24 November 2026. Products, rules and standards change, so check the linked originals before you rely on any detail.

> Related credential: Local LLM Practitioner. Explore the exam facts and syllabus.

Sources and further reading

1. Alex M Tseng, Luca Zancato and Amazon colleagues. Your Experts Have Partners. There's HOPE for Keeping Them Together (Higher-Order Pruning of Experts in Mixture-of-Experts Language Models). Hugging Face enterprise article, 9 November 2026. huggingface.co/blog/amazon/higher-order-pruning-experts.

Published by IICTP, an independent technical credential body, on 24 November 2026. The current version of this post is at https://blogs.iictp.org/hope-higher-order-expert-pruning-moe-explained. More guides: blogs.iictp.org.