

On-policy distillation only ever sees *one token at a time*.

First weekly issue. Last month's conclusion was that on-policy distillation had displaced pruning as the way to make a small model good; this week four independent papers landed on the same limitation in it, and reached for the same shape of fix. Plus speculative decoding arriving in llama.cpp with numbers honest enough to show when it does not help, and a quantization result that should worry anyone running 4-bit in an agent loop.

COMPILED 23 AUG 2026

4 SECTIONS

~40 PRIMARY CITATIONS

WINDOW 7 DAYS

AT A GLANCE

01

Four papers, one diagnosis: on-policy distillation's signal is per-token, and therefore too local. A token-level teacher can patch a bad step without ever showing a complete correct path. Every fix this week adds a longer-horizon term on top: step-level supervision [2608.16333](#), a verifier-disagreement residual [2608.19181](#), and token-budget rebalancing across teachers [2608.19098](#). The fourth explains *why* it generalises at all — and why stacking teachers backfires [2608.16647](#).

02

Speculative decoding landed for LFM2 in llama.cpp, and the reported numbers show exactly what it is for. DSpark drafters merged for LFM2 [#27383](#) and bailingmoe3 [#27508](#). The author's own table averages $2.12\times$ on an RTX 4070 Laptop — but that is $3.22\times$ on a code-merge prompt at 66% draft acceptance against $1.33\times$ on an explanatory one at 18%. Speculation is a structured-output optimisation, and the mean hides that.

03

4-bit quantization degrades a failure mode no benchmark measures. Under high proactive interference — a value overwritten repeatedly in context — bitsandbytes INT4 drops Qwen2.5-7B-Instruct from 81.0% to 68.3%, with same-key intrusion errors rising from 21.5% to 24.6% of trials. INT8, usually assumed free, carries a smaller but real penalty in two of three models tested. Aggregate benchmark accuracy looks largely unaffected throughout 2608.18578.

04

Ornith-1.5 shipped as MIT open weights, in a 35B-A3B MoE and a dense 9B. Three billion active parameters, 262K native context, and a reported 79 on SWE-bench Verified for the MoE `model card`; GGUF conversions appeared the same day.

CONTENTS

§1 llama.cpp — decoding, Metal, a revert saga

§2 The papers — distillation's local-signal problem

§3 Weights that shipped

§4 What I would actually do

The glossary is now a

standing page

rather than an appendix to every issue — it barely changes week to week, and rebuilding it each Monday was the wrong shape. Three terms were added for this issue.

§1 · LLAMA.CPP

Decoding, Metal, and a revert saga

135 pull requests merged between 17 and 23 August, ending at build `b10590`. Grouped by what changes for someone actually running the thing.

Speculative decoding arrives for recurrent models

DSpark drafters merged for LFM2 #27383, which also added partial rollback for LFM2's recurrent state — the part that makes speculation possible on a model that is not purely attention-based — and for bailingmoe3 the following day #27508. Drafter GGUFs are published, so this is usable now rather than pending a conversion.

The author's reported table is unusually honest, and worth reading as the general lesson about speculative decoding rather than as a fact about LFM2. On an RTX 4070 Laptop, with LFM2.5-1.2B-Instruct F16 against its 5-layer DSpark drafter:

PROMPT	SPEEDUP	DRAFT ACCEPTANCE
merge (code)	3.22×	66%
correction	2.98×	64%
poem	1.54×	23%
summary	1.53×	24%
explain	1.33×	18%
mean	2.12×	—

A 2.4× spread across prompt types, tracking acceptance rate almost exactly. If your workload is code edits and structured rewriting, speculation is close to free throughput. If it is explanatory prose, you are hosting a second model for roughly a third more speed. Author-reported on a single machine, so treat the magnitudes as claims — but the *shape* is the well-established behaviour of the technique.

Apple Silicon: quantized KV through flash attention

Metal gained a path to dequantize quantized KV caches (Q4_0, Q4_1, Q5_0, Q5_1, Q8_0) to contiguous F16 and run the existing F16 flash-attention kernels over them #27390. For MLA models, where V is a view of K, the redundant V dequantization is skipped. Two days later the same path was narrowed to large batches only #27438 — the dequantization is not free, and at small batch sizes it cost more than it returned. Ship-then-measure, visible in the commit log. Separately, Q8_0 dequantization on Metal moved to packed types #27370.

Quantizing a large model on a machine that should not fit it

`llama-quantize` can now release a tensor's memory pages once it is no longer needed, rather than holding the whole model resident through the run [#22877](#). This is the unglamorous kind of change that decides whether you can quantize a large model on the hardware you own, or have to rent something for an afternoon.

Server: your model paths were public

When authentication is configured, `/v1/models` no longer answers unauthenticated clients [#26347](#). The endpoint returned filesystem paths to the model files — and a path like `/home/alex/Downloads/...gguf` leaks a username to anyone who can reach the port. If you run `llama-server` with auth on and anything less than a fully trusted network in front of it, this is the week's one genuine security fix.

Elsewhere in the server: a `dedup-cache-models` preset [#27346](#), and `/metrics` is now reachable while the server is asleep [#27376](#), which matters if you were scraping a sleeping instance and reading the timeout as a crash.

New architectures

BailingMoE3 support landed [#26608](#), enabling Ling 3.0 flash with MTP, along with Granite's sliding-window-attention variants, `GraniteSWAForCausalLM` and `GraniteMoeSWAForCausalLM` [#25505](#).

ROUGH EDGES

Reordered Q2_K MMVQ and ESIMD kernels for SYCL merged on 21 August, reporting 1.54× to 1.90× token-generation improvement depending on hardware, measured on Qwen35-27B-Q2_K_M [#26336](#). They were reverted hours later to get CI green [#27486](#), then re-landed the next day with the missing gate parameters that a merge conflict had dropped [#27490](#).

The kernels are in and the speedup is real, but if you build SYCL from source, pin past [#27490](#) — the window between the three contains a build that is either broken or silently missing the optimisation. This is also the standing argument for checking merge state rather than trusting a summary: read at the wrong moment, this story has three different endings.

Distillation's local-signal problem

Issue 01 closed by calling on-policy distillation the thing that had replaced "prune the big one". Four papers submitted inside this one week set about finding its edges, and they converge to a degree that is unusual for seven days of arXiv.

The same diagnosis, four times

OPD trains a student on its own generated trajectories, with the teacher scoring the tokens the student actually produced. The complaint, in four different vocabularies, is that this signal is *per-token*, and a per-token signal is local: it can mark an individual step wrong without ever demonstrating the complete correct path from there.

Step-Level OPD states it most directly — token-level OPD "can provide only fragmented corrections along an erroneous student trajectory and cannot unfold a complete and correct repair path" — and interpolates between OPD and supervised fine-tuning by supervising whole student-generated trajectories at step granularity. At the limits of step length it reduces to SFT or approximates OPD. Reported at 13.4 points of average success rate over vanilla OPD on ALFWorld [2608.16333](#).

GC-OPD attacks the same locality from the verifier side: on long-context tasks, token-level teacher support favours responses that are locally plausible but omit evidence spread across the input. The paper first measures the disagreement — trajectory-level OPD scores drift progressively out of alignment with verifier rewards as input length grows — then uses the signed residual between them as an extra training signal. Post-training the official Qwen3-4B and Qwen3-8B checkpoints raises their five-benchmark averages from 29.08 to 40.47 and 35.12 to 44.65; vanilla OPD reaches 39.31 and 43.56 under the same setup [2608.19181](#).

Open-MOPD finds the locality problem in the multi-teacher setting and, usefully, rules out the obvious explanation. Standard multi-teacher OPD captures only 35.6% of the headroom available relative to a domain-routed oracle ensemble on SmolLM3-3B-Base — and the cause is *not* gradient conflict between teachers but a misallocation of the token-level optimisation budget, driven by sequence-length disparities across domains, convergence drift, and reward staleness. Rebalancing the budget lifts headroom recovery to 83.4% [2608.19098](#).

The fourth is the one to read if you only read one. A controlled study varying one generalisation factor at a time finds that **OPD transfers a teacher's reasoning behaviour rather than its answers**: training difficulty barely matters, and problems the teacher never solves are still useful training signal. What decides transfer is the origin relationship — same-origin teacher/student pairs carry across languages, reasoning horizons and domains, while cross-origin pairs mostly fit the trained distribution. And that reach is the problem in multi-

teacher setups: because routing prompts to domain experts cannot confine a teacher's influence, combining them produces a "mixture-dependent seesaw" among their capabilities [2608.16647](#).

Read together: OPD works because it is broad, and the multi-teacher failures the other three papers are patching are that same breadth arriving as a side effect. That is a real finding for a week, and it is also the limit of what a week establishes — four papers agreeing is a hypothesis with unusually good support, not a settled result. Out of window but part of the same conversation, SimpleOPD tackles the tokenizer-mismatch case, reporting Intern-S2-Preview up 21.2 points on ProofBench to 55.2 ([2608.14277](#), 14 August, out of window).

THE CAVEAT PANEL – COMPRESSION, AGAIN, IS NOT BEHAVIOURALLY NEUTRAL

Issue 01's most useful section was four papers finding that compressed models pass every standard quality guard while behaving measurably differently. This week that line got its most specific result yet, and it names the library almost everyone uses.

Proactive interference is a documented failure mode: when a value is overwritten repeatedly in context, retrieval of the current one decays as prior overwrites accumulate. Testing three precision levels via bitsandbytes across Qwen2.5-7B-Instruct, Mistral-7B-Instruct-v0.3 and Phi-3.5-mini-instruct, **INT4 significantly reduces accuracy under high interference in every model** — 81.0% to 68.3% for Qwen — confirmed by paired McNemar's tests at $p \leq 2.6 \times 10^{-6}$ and a mixed-effects regression across all interference levels. It is mechanistically linked to same-key intrusion errors rising from 21.5% to 24.6% of trials, and an ablation places the effect in the quantized transformer backbone rather than the output projection. INT8, "often assumed safe", also carries a smaller but real penalty in two of the three models [2608.18578](#).

The load-bearing sentence is the last one in the abstract: this cost lands on applications with long, updatable, semantically dense contexts *even when aggregate benchmark accuracy appears largely unaffected*. An agent that revises a value in its own context across a long run is exactly that application. The effect also reverses sign under a numeric control condition, so this is not a general "4-bit is worse" result — it is specific, which is what makes it actionable.

Serving efficiency, which had a good week

ReCache targets a real and under-discussed problem: agents re-encode tool and skill schemas that recur across requests in different orders, which defeats ordinary prefix caching. Making the KV blocks composition-invariant matches dense invocation performance (82.3% against 82.4% Inv-F1) at a reported $3.655\times$ time-to-first-token speedup; the full framework cuts allocated KV-tensor memory by 92.43% [2608.19662](#). If you serve agents over a fixed tool catalogue, this is the paper of the week for you.

FlashPrefill V2 is the "prototype becomes deployable" step: paged KV cache, continuous batching, FP8, and kernels aligned with FlashAttention-3/4, so it can sit as an attention backend in SGLang. Reported at up to

47.26× and 27.19× over FlashAttention-2 at 128K context under FP8 and BF16 on H20 GPUs, and 30.49× against an FA3/4-aligned dense baseline in FP8 2608.19758. Prefill only — check the glossary if that distinction is not automatic.

KeysAndValues is the practical one: an AWS Labs open-source library for fine-tuning models to co-adapt with a sparse-attention KV policy, on a single A100 40GB, including an efficient H2O implementation with dedicated SDPA kernel support. Training with the policy rather than bolting it on afterwards often beats models trained with exact attention 2608.19920.

Two more worth knowing about. **WhiteMatter** lets every attention layer attend to representations from all layers of each past token via a router that mixes layer states into a fixed number of cached channels — and since that channel count sets the cache size, KV compression falls out of the architecture rather than being applied to it. Reported to outperform a vanilla Transformer with 50% more layers, retaining most of the gain at 50% KV-cache compression 2608.18486. And on the unfashionable end, SmoothQuant integrated into TorchAO with fused INT8 GEMM selection across oneDNN, AVX512_VNNI and AMX delivers up to 5.8× end-to-end throughput on BERT-family encoders on Intel Xeon, upstreamed to PyTorch 2608.18182. Encoders still run a lot of production ranking and retrieval.

The edge, taken seriously

FreeToken is the week's most direct answer to "what can I actually run". Rather than fixing an offloading strategy, it continuously maps computation and model state onto whatever the machine has, co-designing model layout, expert residency, CPU–GPU execution and agentic state reuse for MoE models. The claimed envelope: a 35B model on an 8GB laptop GPU, 284B on a gaming desktop, and the 753B GLM-5.2 on a single workstation GPU, across more than 20 MoE models and real coding and tool-using agents 2608.16157. Vendor-shaped claims from the authors of the system, and the interesting number is not the 753B headline but whether the 8GB case is usable at a sane token rate — which the abstract does not say.

Complementing it, a characterisation of edge RAG on an NVIDIA Jetson AGX Thor finds generation dominates the budget for 7B–8B generators — roughly 90% of per-query latency and 91% of GPU energy — and that intermediate context compression hits a useful operating region, cutting GPU energy by up to 53.2% and SoC energy by up to 48.2% with negligible quality loss. Too mild and you leave energy on the table; too aggressive and quality goes 2608.19535.

Weights that shipped

A quiet week, which is what most weeks look like — one family, two sizes, both MIT. Open weights only.

DATE	MODEL	WHY IT MATTERS
------	-------	----------------

18 Aug	Ornith-1.5-35B-A3B	MoE, ~35B total but ~3B active per token, 262,144 native context extensible to roughly 1M via YaRN, MIT licence. Reported 79 on SWE-bench Verified, 59.6 on SWE-bench Pro, 67.8 on Terminal-Bench 2.1 and 89.2 on GPQA Diamond, averaged over five runs per the card. The active-parameter count is the number that matters here: 3B active is a laptop-class decode cost attached to a 35B-class model. GGUF conversions appeared the same day.
--------	---------------------------	--

18 Aug	Ornith-1.5-9B	Dense 9B, ~19 GB in bf16, same 262K context and MIT licence, described as the lightweight member of the family. Reported 70.6 on SWE-bench Verified and 86.4 on GPQA Diamond. Fits one consumer GPU without quantization, which the 35B-A3B does not.
--------	----------------------	---

Out of window but adjacent, and the more interesting release if you care about provenance: DFM Mimir v1 is a 1B Hierarchical Reasoning Model trained from scratch on 161 datasets using only permissible post-training data, reported competitive with substantially larger models and state of the art for Danish (2608.13517, 13 August, out of window). Permissibly-sourced training data at frontier-adjacent quality is a claim worth watching, because it is the one most open-weight releases cannot make.

§4 · TAKEAWAYS

What I would actually do

Sorted by readiness, not by how interesting the thing is.

ADOPT

If you expose `llama-server` with authentication on, update. #26347 is a small change to a real leak: `/v1/models` was handing filesystem paths, and therefore usernames, to unauthenticated callers. Nothing else this week is urgent; this is.

ADOPT

Measure your 4-bit deployment on a task that overwrites values in context, before you trust it in an agent loop. The proactive-interference result 2608.18578 is specific enough to act on: build a probe where a variable is reassigned several times and the model must report the current value, and run it at your deployed precision. Your benchmark suite will not catch this, by construction — that is the paper's central point. If the probe degrades, INT8 is the cheap step before FP16.

WATCH

Speculative decoding: adopt it for structured output, skip it for prose. The DSpark table #27383 puts numbers on something usually stated vaguely. Before hosting a draft model, look at what your workload actually generates. Code edits, structured rewriting, diffs, extraction: worth it. Explanation, summarisation, chat: you are paying memory for a third more speed. Measure acceptance rate, not throughput — acceptance is the thing that predicts whether it will keep working when the workload shifts.

WATCH

If you are running multi-teacher on-policy distillation, stop and read 2608.16647 first. "Route prompts to domain experts" is the natural design and the evidence says it does not confine each teacher's influence, so capabilities trade off against each other in a way routing cannot fix. Same-origin teacher/student pairs transfer far better than cross-origin ones — a cheap constraint to respect at the point where you pick the teacher, and expensive to discover afterwards.

WATCH

Agent servers over a fixed tool catalogue: ReCache is the right idea. 2608.19662 A 92.43% reduction in allocated KV memory for schemas that recur in varying order is large enough to change what a box can hold. Code is released; the reported numbers are the authors' own on a benchmark they assembled, so this needs reproducing on your own tool set before it goes anywhere near production.

LAB ONLY

The OPD fixes themselves. SOPD, GC-OPD and Open-MOPD are all credible and all report gains over vanilla OPD on their own setups (2608.16333, 2608.19181, 2608.19098), but they are one week old, none has been reproduced independently, and each adds a moving part to a training pipeline. The useful thing to take now is the diagnosis, not the recipes: if your distilled student plateaus, suspect that a token-level signal is patching steps without teaching paths.

FreeToken's envelope. Running a 753B model on one workstation GPU 2608.16157 is the week's most eye-catching claim and the least verifiable from the abstract: no token rate is quoted for the constrained configurations, and "runs" and "usable" are different claims at 8GB. Worth trying precisely because it is easy to falsify yourself.

NOTE

One week is not a trend. Four papers converging on the same limitation in on-policy distillation is the most interesting thing here, and it is still four papers in seven days. Some of that is genuine convergence on a real problem; some is arXiv's own clustering, where a well-known limitation becomes fashionable to address at the same moment. The honest position at this cadence is that the diagnosis is worth internalising and the fixes are worth revisiting in a month, and this issue is not going to pretend otherwise to sound more decisive.

Sources

LLAMA.CPP §1

[llama.cpp releases](#) — build tags through [b10590](#) (23 Aug).

Merged pull requests 17–23 Aug were enumerated from the GitHub API rather than a digest, and every PR cited below was fetched individually to confirm it merged.

[#27383](#) DSpark speculative decoding for LFM2 · [#27508](#) DSpark for bailingmoe3

[#27390](#) Metal: dequantize quantized KV before flash attention · [#27438](#) narrowed to large batches · [#27370](#) Q8_0 packed dequantization

[#22877](#) quantize: evict weights after each layer

[#26347](#) server: models endpoints private under authentication

[#27346](#) server: dedup-cache-models preset · [#27376](#) server: /metrics during sleep

[#26608](#) BailingMoE3 · [#25505](#) GraniteSWA / GraniteMoeSWA

[#26336](#) Q2_K reordered MMVQ and ESIMD for SYCL · [#27486](#) revert · [#27490](#) re-land with fix

[2608.16333](#) Step-Level On-Policy Distillation

[2608.19181](#) Beyond Teacher Likelihood: Group-Calibrated On-Policy Distillation — [code](#)

[2608.19098](#) Open-MOPD: Diagnosing and Fixing Capability Imbalance in Multi-Teacher On-Policy Distillation

[2608.16647](#) Every Coin Has Two Sides: On the Dual Nature of Generalization in On-Policy Distillation

[2608.18578](#) Compress and Forget: bitsandbytes Quantization Amplifies Proactive Interference — [code](#)

[2608.19662](#) ReCache — [code](#)

[2608.19758](#) FlashPrefill V2: Block-Sparse Prefill Attention

[2608.19920](#) Learning how to Forget — [KeysAndValues](#)

[2608.18486](#) WhiteMatter: All-to-All Cross-Layer Connections via KV Mixing

[2608.18182](#) Efficient INT8 Inference of Small NLP Models on Server CPUs

[2608.16157](#) FreeToken: Efficient Edge-Native MoE Serving

[2608.19535](#) From Retrieved Context to Runtime Control: Adaptive Compression for Edge-based RAG

Out of window, cited as context: [2608.14277](#) SimpleOPD (14 Aug) · [2608.13517](#) DFM Mimir v1 (13 Aug)

[Ornith-1.5-35B-A3B](#) · [Ornith-1.5-9B](#) — specifications and benchmark figures are as stated on the model cards.

[Hugging Face daily papers](#) — new to the source list this week, used for discovery across the seven days; every paper it surfaced was then read at its arXiv abstract before being cited.

Compiled 23 August 2026 from the llama.cpp release feed and GitHub API, the arXiv API filtered to this window, and Hugging Face's curated daily papers. Figures quoted as speedups, compression ratios or benchmark scores are the authors' or vendors' own reported numbers on

their own benchmarks — treat them as claims, not measurements, until we reproduce them. Anything dated outside 17–23 August 2026 is labelled inline as context. Terms are defined on the [glossary page](#).



LLM Radar is published weekly by [Montevive](#) · [Atom feed](#) · ai@montevive.ai · [Privacy](#)