Nils Matteson

San Jose, CA | (208) 921-2576 | nilsmatteson@icloud.com | linkedin.com/in/nilsmatteson | github.com/matteso1 | nilsmatteson.com

EDUCATION

Northeastern University – Incoming M.S. Computer Science — Expected May 2028

University of Wisconsin–Madison – B.S. Data Science, Minor in Computer Science — May 2026

TECHNICAL SKILLS

Languages: Python, Rust, Go, C++, Java, TypeScript/JavaScript, SQL, CUDA
Systems & Cloud: AWS (Bedrock/S3), Linux, Docker, PostgreSQL, Redis, gRPC, Kafka, Git, CI/CD
ML & Inference: PyTorch, vLLM, SGLang, Triton, Hugging Face, XGBoost, scikit-learn

EXPERIENCE

vLLM (open-source fellowship, sponsored by Inferact) — Jul 2026 – Present
Open-Source Fellow & Contributor — *Python · C++ · CUDA · PyTorch*

- Landed **5 upstream vLLM PRs in 8 days** across GPU sleep/wake, fingerprinted startup-plan caching, compile-cache correctness, and startup observability; added CPU-only regression tests while preserving default behavior.
- Diagnosed a cache-key bug where `kv_cache_memory_bytes` forced full recompilation; H100 tests showed it cost **21–27s** to save ~2s of profiling, then merged the fix and regression test as **PR #47356**.
- Submitted **PR #48190**, a portable DeepGEMM JIT-cache key that achieved **41/41 cross-layout cache hits** and cut Qwen3-30B startup from **170.6s to 76.9s (55%)** in a controlled H100 A/B.
- Built a 2×A100 C++/CUDA/NCCL harness validating captured all-reduce graph replay across communicator suspend/resume without recapture: **12/12 oracle-clean trials** after a ~63 GiB memory-pressure arm.

thaw (open-source LLM-inference infrastructure) — Apr 2026 – Present
Founder & Lead Engineer — *Rust · CUDA · Python*

- Built and shipped **thaw**, open-source checkpoint/branch/restore infrastructure for live vLLM/SGLang state; forked an H100 session in **0.88s median** versus a ~340s cold boot and shipped **16 PyPI releases**.
- Reached **14.3 GB/s** disk-to-GPU weight restore (3.4× faster 70B load) with double-buffered `O_DIRECT` DMA and parallel CRC32C verification; hot-swapped an 8B model in **0.29s at 55 GB/s**.
- Cut KV-cache snapshot overhead **60**× by coalescing ~16K per-block DMAs into one gather; rebuilt vLLM's prefix-cache index on restore so forked sessions skip prefill.

Matteson Systems LLC — May 2026 – Present
Founder & Software Engineer — *Next.js · PostgreSQL · Playwright · Claude*

- Built and shipped an autonomous SMB outreach platform that scored **10,500+ businesses**, surfaced 158 priority leads, and tracked per-call spend at **~$0.03 per lead**.
- Built Claude-vision audits from Playwright/Lighthouse data and an append-only, event-sourced PostgreSQL CRM with threaded Gmail outreach and payments.

Research Cyberinfrastructure, UW–Madison DoIT — Jan 2026 – Apr 2026
AI Workflows Research Assistant — *AWS Bedrock · RAG · LLM eval*

- Built an AWS Bedrock evaluation and cost-tracking framework for **9 LLMs** and 3 ensembles across 282 questions; identified Llama-4 Maverick at **98% of top accuracy** with lower cost/latency and presented at UW's ML+X Forum.

SELECTED PROJECTS

Sentinel: Distributed Message Queue | github.com/matteso1/sentinel — *Go*

- Implemented core **data structures and algorithms** for a Kafka-inspired distributed log: skip-list memtables at 3.9M reads/s, SSTables with CRC32, WAL, leveled compaction, gRPC, and consumer groups.
- Implemented **Raft** leader election and log replication; validated split-brain prevention and partition recovery with a deterministic network simulator and **45 tests**.

Madison Metro ML: Real-Time Bus Arrival Prediction | madisonbuseta.com — *Python, XGBoost, React*

- Shipped a live **47-feature XGBoost** ETA service with calibrated 90% conformal intervals; automated nightly retraining behind a $\geq$2s MAE/no-regression deployment gate and rendered 200+ vehicles at 60fps.

Re-feeding Is Not Replaying: LLM Reproducibility Study | arXiv:2606.15621 — *Python, vLLM*

- Sole-authored a three-pass experiment showing re-feeding shifts token-credit estimates **14–28 percentage points** above replica noise at low-margin decisions; published code, logs, and per-pivot data.