Yan Gaev

Research Engineer / ML Engineer

LLM post-training & reinforcement learning · Agents & evaluations · ML systems

yangaev.engineer@gmail.com · github.com/YanGaev2 · t.me/yangaev1

Open to relocation to London / USA and remote opportunities

Research and ML engineer working across LLM post-training, agents and production inference. AI Head at TurboCloud; AI/LLM roles since 2023 and 30+ AI projects shipped. Experience includes Russian-language reasoning models, SFT and GRPO, token-level RL research, Rust agent runtimes and CUDA optimization.

PROFESSIONAL EXPERIENCE

AI Head / Lead ML Engineer — TurboCloud

2024–Present · Progression: Systems Analyst → AI Architect → AI Head

- Built a corporate agent harness and platform in **Rust and React**, adopted by **over 500 internal users**; received direct recognition from company leadership.
- Integrated a code sandbox, web search and tracing; tracing made agent iteration **2× faster**.
- Fine-tuned open models that matched GPT-5.5 (xhigh) on internal company benchmarks using two GPUs.
- Modified SGLang to achieve **30% higher production throughput**; deployed dozens of on-premises LLM servers.

AI/LLM Engineer — Smartseeds | 2023–2024

Lead Systems Analyst — T1 Innovations | 2022–2023

Systems / Data Analyst | 2019–2022

Moy Dom · Cherkizovo Group · INFOLine.

SELECTED WORK — POST-TRAINING & AGENTS

Russian-language reasoning model — SFT, GRPO & token-level RL

Instruct-to-reasoning conversion · 20M+ training tokens

- Converted an instruct model into a reasoning model through **SFT**; adapted the `tool_call` template and `reasoning_parser`.
- Trained on **more than 20 million tokens** and ran **GRPO** (Group Relative Policy Optimization).
- Researched token-level RL updates, training a neural network to target specific tokens within a rollout.

Tool-use SFT evaluation — 10B mixture-of-experts model

GigaChat3.1-10B-A1.8B · PyTorch, TRL, PEFT / LoRA · H200

- Measured SFT run: **132 agent traces / 2.2M tokens**, contexts up to 250k. Corrected a loss-masking bug involving role headers.
- On a held-out evaluation of **30 episodes / 102 decisions**, improved call/no-call accuracy from **66.7% to 89.2%** and correct-tool selection from **29.2% to 52.8%**.

Rust agent harness — model-specific tools & reliable execution

Rust · github.com/YanGaev2/rust-harness

- Built an agent runtime with tool schemas informed by model probes, bounded concurrent execution, tool-call repair and execution traces.
- Benchmarked **26 tasks across 6 models, with two repeats**: no tool-execution failures across **1,199 real tool calls**. Adapting tools to DeepSeek-V4 priors reduced tool calls by **46%** in the measured comparison.

AGENT APPLICATIONS & DEVELOPER TOOLS

SlideML — an agent skill and compiler for editable presentations

Rust, Python, Open XML · github.com/YanGaev2/slideml

- Built a compact XML interface and Rust compiler that let agents create native, editable PowerPoint slides, charts, tables and animations.
- Added font-based text measurement, layout warnings and JSON geometry manifests for programmatic checks; a Python validation workflow renders slides in PowerPoint for visual review.

Graph-enhanced retrieval-augmented generation (RAG)

Python, FastAPI, Neo4j, Ollama, Next.js, Docker · github.com/YanGaev2/hse-vector-system

Built a document ingestion and chat application combining vector retrieval, entity extraction and a knowledge graph, with a web interface and containerized deployment.

INFERENCE & ML SYSTEMS

LLM inference engine from scratch

Qwen3.5-2B · Custom CUDA kernels, CUDA Graphs · RTX 5090

- Achieved **322 tokens/s at batch 1** and **7.3k tokens/s at batch 32**; batch-1 decode was **21× faster** than the measured Hugging Face + FLA baseline.
- Accelerated 262k-token prefill by **2×** and verified **100% greedy-token agreement** with the reference `generate()` output in the evaluated runs.

Speculative decoding, model surgery & KV-cache compression

llama.cpp · SGLang

- Grafted an MTP head from Qwen3.8-27B into BAAI AREX-2; measured **0.79 peak draft acceptance** and approximately **100 tokens/s** on one 32 GB GPU.
- Implemented TurboQuant in SGLang with **5.6× KV-cache compression** in the measured setup.

RESEARCH EXPERIMENTS & TRAINING DATA

Verifiable-reward and preference data for post-training

Created **1.5M tokens** of records from arXiv ML papers and rewrote agent traces containing **892 tool calls** for post-training datasets.

Concept learning and discovery of an unseen physical law

Designed a synthetic world with gravity proportional to $1/r^{2.4}$, three 20M-token corpora and evaluation thresholds fixed before training. Nine 19.5M-parameter GPT baselines failed to state the law in 2,700 attempts; an MDL concept library recovered exponent **−2.39998**. Ongoing world-model research.

TECHNICAL SKILLS

Training: PyTorch, Transformers, TRL, PEFT / LoRA, SFT, GRPO, token-level RL, MoE models, reasoning parsers, agent trajectories, verifiable-reward and preference data.

Agents & evaluation: tool calling, execution harnesses, sandboxes, tracing, held-out evaluations, BFCL, τ^2-bench, SWE-bench, LLM-as-judge, RAG.

Systems: Python, Rust, CUDA, CUDA Graphs, SGLang, vLLM, llama.cpp, speculative decoding, KV-cache optimization, H100 / H200, Docker.

EDUCATION & LANGUAGES

HSE University, Faculty of Computer Science
Artificial Intelligence & Computer Vision · 2026

Languages: English C1 · Russian native