Cătălin Friptu

Targeting AI / LLM engineering roles: evals, ML + LLM systems, agentic engineering

friptu.05@gmail.com | +373 767 85 221 | github.com/malmurk | linkedin.com/in/catalin-friptu-437026270
Chișinău, Moldova (UTC+2/+3) | Remote | EU (Romanian) citizen | B2B contract or EOR

SELECTED RESULTS

- Product categorization into a 3,520-leaf taxonomy: **84.8% acceptable-set accuracy vs 47.4%** for the previous model, on a pre-registered blind test of 599 supplier rows (95% CI 81.7–87.5).
- **60.8% of test rows end auto-assigned and correct with no human touch**: 63.6% go out automatically at 95.5% precision, the rest to an LLM or a reviewer, for **$0.04 of OpenAI spend per 1,000 rows** and ~110 ms/row on CPU.
- Marketplace buyer app, release candidate: **~2,800 test cases and 72 Maestro E2E flows** behind one command.

How I work. I design the system and set the evaluation bar. Claude Code agents write most of the code; I own verification (test suites, blind evals) and decide what ships. Both main repos are private; I can show the code on a call.

PROJECTS

Hybrid ML + LLM product categorization May 2026 – present

Pilot project for a Moldovan e-commerce marketplace, alongside my Stefanini job. RO/RU/EN listings. Evaluated offline; deployment unconfirmed.

- **Design.** Six candidate sources (a fine-tuned multilingual encoder, lexical match, bge-m3 kNN and others) feed an MLP ranker with isotonic calibration. The calibrated score routes each row to auto-assign, a constrained LLM pick among the candidates (20.5% of test rows), or a person.
- **Metric.** Acceptable-set accuracy: the taxonomy has duplicate leaves, so a row can have more than one right answer. Gold labels from blind LLM double-labelling with adjudication; test set fixed in advance and read once.
- **Limits I found.** Without a supplier's own rows in the kNN index, the local ranker's top-1 on that supplier's dev split fell from 80.7% to 45.7%, so 84.8% is an in-supplier figure. On 300 rows from sources the system had never indexed it scores 79.0%, short of my target. I dropped a pseudo-label retrain that failed the supplier hold-out.
- **Previous version (v5).** Fine-tuned the encoder on 527,086 rows with ROCm on an AMD RX 9070 (~5.5 h); INT8 ONNX at 34.6 ms/row; FastAPI service with a MariaDB poller in Docker; 327 tests.

Python, PyTorch, Transformers, scikit-learn, ONNX Runtime, ROCm, bge-m3, OpenAI API, FastAPI, MariaDB, Docker

Agentic engineering: buyer app for a Moldovan multi-seller marketplace May 2026 – present

Same marketplace, same pilot. React Native / TypeScript, two developers, release candidate, not yet in stores. No AI features in the app.

- **App decisions.** The mobile API wraps the existing backend rather than rewriting it; money converts to minor units in the app's adapter layer. An audit found the HTTP client's retry default would resend a checkout POST after a network drop and could duplicate an order, so I limited retries to idempotent requests and pinned it with a test.
- **Delivery process.** An orchestrator hands work to builder, API-contract-auditor and adversarial-reviewer subagents, each in its own git worktree. 26 campaigns with ledgers; 1,202 of the 1,550 commits carry a Claude Code co-author trailer.
- **Checks.** One command runs tsc, eslint, jest, import-cycle and i18n-parity checks, then the 72 Maestro flows on a fresh emulator. A device-lock plugin keeps parallel agents off the same emulator. 42 ADRs. A pre-launch risk audit flagged that the checkout's order button and terms fell short of consumer-law rules; fixed in the release candidate.

TypeScript, Expo / React Native, TanStack Query, Zustand, axios, Jest, Maestro, EAS, Claude Code subagents

Supporting: agent harness and federated RAG May 2026 – present

- Federated retrieval over 5 knowledge bases (bge-m3 on my own GPU); a prompt hook auto-injects the strongest matches from the main one. On a 20-query eval with a RO/RU subset, the model switch plus a chunking-bug fix took recall@1 from 0.467 to 0.733 and MRR from 0.598 to 0.771.

EMPLOYMENT

Support Engineer, Stefanini EMEA, Chișinău Mar 2026 – present

- Live EMEA ticket queue: reproduce the problem, narrow it down, write an escalation an engineer can act on.

SKILLS

ML and evals: PyTorch, scikit-learn, calibration, ONNX, pre-registered blind tests, Wilson CIs, McNemar, group hold-out
LLM and retrieval: OpenAI API, prompt design, constrained LLM selection, RAG, embeddings (bge-m3), llama.cpp, MCP
Agentic engineering: Claude Code subagents, hooks and skills, git worktrees, model routing, ADRs
Engineering: Python, TypeScript, SQL (MariaDB, SQLite), FastAPI, REST APIs, Docker, React Native / Expo, Jest, Maestro, Bash
Languages: Romanian and Russian (native), English (B2–C1)

EDUCATION

Database Administration, Centre of Excellence in Informatics and IT (CEITI), Chișinău 2021 – 2025