YVES DONATO

Toronto, Ontario

(905) 966-1994 | yvesdonato@pm.me | linkedin.com/in/yvesd | github.com/YvesDonato | yvesdonato.com

EDUCATION

Honours Bachelor of Computer Science (Data Analytics) — Sept 2023 – Expected Dec 2026
Sheridan College | GPA: 3.41/4.0 — Oakville, ON

EXPERIENCE

Research Assistant — *July 2026 – Present*
Sheridan College's Centre for Applied AI — Oakville, ON

- Build software that trains and tests large AI models, helping researchers save their work and compare experiments.
- Built sharded PyTorch checkpoints with atomic saves, restoring model, optimizer, and RNG state for resumable MoE training.
- Preserved FP32 optimizer moments when restoring BF16 models, adding byte-exact checks for parameter and optimizer state.
- Weighted SFT losses by global supervised-token counts across ranks to keep gradient accumulation consistent on packed data.
- Handled all-zero SFT loss masks with synchronized rank skips, preserving token and dataset cursors without optimizer updates.
- Reassembled striped MoE expert shards for Hugging Face export, checking tensor shapes and dtypes to preserve expert order.
- Tuned checkpointing and fused operations; reports measured 50.22K to 66.40K training tokens/s across four H100 GPUs.
- Built Python reports for 113-task coding evaluations, separating strict solves, partial-test scores, agent errors, and patch output.

Machine Learning Engineer Intern — *Apr 2026 – Present*
StereoLOGIC — Toronto, ON

- Built AI that identifies the computer window someone is using, helping software understand their work from screenshots.
- Fine-tuned GUI-Owl with LoRA and GRPO in Unsloth to identify active windows in screenshots and evaluate model behavior.
- Added SFT dataset checks for mixed input styles, missing images, marker conflicts, and metadata to catch invalid training rows.
- Fingerprinted metadata-renderer code in dataset manifests to detect drift between baked training inputs and live inference.
- Preserved image and video preprocessing configs when merging GUI-Owl adapters for export to avoid serving-time input drift.

Co-Founder & CTO — *Mar 2026 – Present*
IgnisXR — Waterdown, ON

- Lead a 4-person team building Ember, a mask to help firefighters find their way through smoke, tested by Toronto Fire.
- Fixed an OpenCV CLAHE data race with C++ thread-local state, isolating image processing across render and inference threads.

AI Engineer — *Sept 2026 – Present*
Sheridan College — Oakville, ON

- Build an AI tool that helps Sheridan professors create, review, and publish teaching materials for their college classes.
- Build the platform's Python backend and Next.js frontend while co-authoring a research paper on AI-assisted curriculum workflows.

PROJECTS

Effects of Zero-Shot Chain-of-Thought on Medium Language Models — *Jan 2025 – April 2026*
Thesis — Final Grade: 91%

- Tested four AI models on hard math problems; step-by-step reasoning did not clearly improve how often they were right.
- Built streamed Ollama inference with per-problem timing and append-only JSONL prediction logs for reproducible evaluation.

Proxmox LLM Server | *Personal project: Rust, Python, Docker, Ollama* — *May 2025 – Present*

- Built a server to run and test AI models, with separate software setups so each experiment can be managed on its own.
- Built Rust APIs for Ollama on Proxmox, isolating local AI model workloads with Docker and LXC for serving and experiments.

AWARDS

2× Track Winner (NeuRizz @ Hackville, Jan 2026) · 2× Track Winner (Ember @ HackCanada, Mar 2026) · 1st Place (Ember @ Pitchathon by SFC, Apr 2026) · 5th Place Overall (Prismarine @ C+H+D, Feb 2026) · IBM Track Winner (Voxel @ GenAI, Mar 2026) · Valsea Track Winner (NailedIt.ai @ Cursor, May 2026)

RELEVANT SKILLS

ML & Training: Python, PyTorch, MoE, SFT, LoRA, GRPO, Unsloth, vLLM, Ollama, Pandas, Polars, R
Systems & Tooling: C++, C, Rust, Linux, NixOS, Docker, Proxmox, Git/GitHub, OpenCV, Benchmarking
Software & APIs: FastAPI, REST APIs, TypeScript, JavaScript, React, Next.js, WebRTC, SQL, PostgreSQL