Gabriel Pedde Ungureanu | AI Engineer & ML Scientist

Brescia, Italy
gabry9815@gmail.com • gabriel-pedde.github.io/Gabriel-Pedde
Gabriel-Pedde

Profile

Physicist by training (PhD, 6 papers), now focused on AI/ML. My doctorate was large-scale computation at its core — parallel jobs on EuroHPC clusters, MPI debugging, automated pipelines. That transfers well to ML infrastructure. Finishing the MHPC at SISSA/ICTP; looking for Applied AI Scientist or ML Engineer roles.

Professional Experience

SISSA **Trieste, Italy**
Doctoral Researcher *2022–2026*
Ran **large-scale distributed computations** on EuroHPC-class clusters: writing code, managing SLURM jobs, debugging MPI failures. Six peer-reviewed papers.

Education

SISSA / ICTP Joint Programme **Trieste, Italy**
Master in High Performance Computing and AI (MHPC) *2025–2026*
Machine learning, deep learning, reinforcement learning, LLM agents, and distributed GPU computing. Hands-on work on **Leonardo (CINECA)**, Ulysses (SISSA), Argo (ICTP).

Università Cattolica del Sacro Cuore **Brescia, Italy**
MSc Physics *2020–2022*
Theoretical physics, quantum field theory, condensed matter.

Università Cattolica del Sacro Cuore **Brescia, Italy**
BSc Mathematics *2017–2020*

Awards & Publications

Best graduate of the year, MSc Physics — Università Cattolica del Sacro Cuore

110/110 *cum laude* — MSc Physics | 110/110 *cum laude* — BSc Mathematics

Istituto Toniolo merit fellowship — BSc, Università Cattolica del Sacro Cuore

6 peer-reviewed papers in theoretical physics — INSPIRE-HEP | Google Scholar

Selected Projects — AI & ML

LLM RL fine-tuning: Implemented RLOO/PPO from scratch to fine-tune **Llama-3.2-1B** on GSM8K with a rule-based reward signal (KL regularisation was the tricky part); **+11 pp answer accuracy** (35 % → 46 %), format compliance from 51 % to 100 %. Distributed on **Leonardo (CINECA)**. Stack: PyTorch, TRL, SLURM.

Multi-Agent LLM System: Opinion-dynamics experiments: N = 4 Qwen2.5 agents, **5 model sizes** (0.5B–14B), 10 rounds. Models ≥3B converged to $< 10^{-3}$ **error** at round 0 — a sharp transition driven by raw capacity, not training rounds. Stack: Ollama, Python.

Deep Learning: Built the full stack in PyTorch: autograd through CNNs, autoencoders, and representation learning. Also studied intrinsic dimensionality of learned representations — what the network stores is surprisingly low-dimensional.

Distributed GPU: CUDA kernel development through to cuBLAS; distributed dense linear algebra with **MPI + CUDA**. Distributed and federated training workflows on SISSA/CINECA infrastructure.

Technical Skills

AI / ML: PyTorch, Hugging Face Transformers & TRL, scikit-learn, cuML

LLM / Agents: RLHF, REINFORCE/RLOO, PPO, GRPO, fine-tuning, RAG, MCP, CrewAI, Ollama

ML methods: CNNs, autoencoders, representation learning, dimensionality reduction, Bayesian inference, clustering

Programming Languages: Python, C++, C, Fortran, Mathematica, Java, LaTeX

Scale / HPC: CUDA, cuBLAS, OpenACC, MPI, OpenMP, SLURM, PETSc/SLEPc

MLOps: Docker, Kubernetes, Singularity, AWS, Git, HDF5/NetCDF

Clusters: Leonardo (CINECA), Ulysses (SISSA), Argo (ICTP)

Languages: Italian (native) | English (C1) | Romanian (A2)