

YIORGOS PAPADOPOULOS

ML & LLM Systems Engineer

Thessaloniki, Greece • giorgospapadopoulos002@yahoo.com

giorgospapadopoulos.dev • linkedin.com/in/yiorgospapadopoulos/ • github.com/yiorgospapadopoulos

ABOUT ME

ML & LLM Systems Engineer and MSc student in Artificial Intelligence and Applications at the University of Thessaly. I build production Python backends for ML and LLM systems, from cost-guarded LLM generation to tested ML services, and trained medical imaging models for my thesis (graded 10/10). Strong on typed, tested code and multi-round code review.

PROFESSIONAL EXPERIENCE

Machine Learning and LLM Systems Engineer | Zimmerman February 2026 – Present

- Building backend services that integrate ML, NLP, LLMs, and embedding-based retrieval, exposed through typed APIs and React frontends.
- Owning backend features end-to-end (see Projects), from design through PR review to deployment.
- Working closely with senior engineers through multi-round code reviews on GitHub, raising the bar on code quality, typing discipline, and architectural clarity.
- Operating in an agile workflow (Jira sprints, shippable tickets) and presenting completed work to the engineering team.

EDUCATION

MSc in Artificial Intelligence and Applications, University of Thessaly October 2026 – Present

BSc in Computer Science & Telecommunications, University of Thessaly 2021 – 2026

- GPA: 7.5/10
- Specialization in Data Management and Artificial Intelligence
- Thesis: chest X-ray pathology classification, graded 10/10 (see Projects)

Erasmus+ International Exchange | ISLA Polytechnic Institute, Porto 2024 – 2025

- Completed an international study program in Portugal during the 2024–2025 academic year.

TECHNICAL PROJECTS

Thesis: Multi-Model Deep Learning for Chest X-Ray Pathology Classification (*Graded 10/10*)

- Designed and trained seven disease-specific models, each fusing MobileNetV2 and ResNet50 backbones, on the NIH ChestX-ray14 dataset (~112K images), with all models achieving 90%+ AUC.
- Implemented Grad-CAM visualization to produce per-class saliency maps, making model predictions interpretable for clinical review.
- Built a PyQt5 desktop application wrapping the trained models, enabling end-to-end inference and visualization from a single interface.

Model Pipeline: ML Service for Grant-Management Analytics *(Zimmerman)*

- Audited and corrected ML models across 24 use cases (regression, classification, clustering, anomaly detection, optimization, network analysis, causal inference) in a Python/Flask service: removed target leakage, fixed data aggregation and train/test splits, aligned prediction inputs with production data, and capped model size to cut saved artifacts from several GB to ~1MB.
- Added experiment tracking (per-run logging of config, winning model, hyperparameters, data and git versions), deterministic run replay, and a data migration script.
- Drove the service to 100% test coverage (1,300+ tests) behind a pre-commit gate of ruff, flake8, bandit, and type checking, over Pydantic-typed config and a Swagger-documented REST API.

Narrative Engine: Cited Country Narratives for the Global Fund *(Zimmerman)*

- Built a Python/Flask service that collects Global Fund country evidence and supporting documents and generates validated, cited written narratives through LiteLLM, saved as versioned SQLite bundles.
- Separated generation from serving: narratives are produced by an explicit, cost-guarded job with a persistent spend ledger, while authenticated API reads only serve saved content, so visitors never trigger a model call.
- Integrated the saved narratives into the Data Explorer frontend behind a feature flag, with documented output contracts and automated validation of generated content.
- Typed, class-based Python at 100% test coverage behind pre-commit formatting, linting, type, and security checks; Dockerized with Gunicorn.

AI Lead Generation Agent: Automated Sourcing from IATI Funding Data *(Zimmerman)*

- Built a Python/Flask pipeline querying the IATI Datastore (Solr, ~875K activities) to source and qualify sales leads from international aid funding data, replacing a manual process.
- Implemented a signal-detection layer of five org-level detectors (new funding, first-time publisher, activity spike, dormant publisher, large commitment), each running scoped Solr queries and returning structured evidence for review.
- Built a three-stage contact-enrichment module (cached prior runs, then IATI data, then OpenAI web search as fallback) that cut paid LLM calls by reusing earlier results, with per-run token and cost tracking.
- Ran a month-by-month historical replay to measure lead volume, filtering out existing clients, multilaterals, and large publishers; typed, class-based Python at 100% test coverage enforced via pre-commit.

CERTIFICATIONS

- AI in Healthcare Specialization, Stanford University. 5-course specialization covering clinical machine learning implementation, algorithmic fairness, and healthcare regulatory frameworks.
- Generative AI for Healthcare, Google Cloud (2026).

PROGRAMMING SKILLS

- **Machine Learning:** PyTorch, TensorFlow, Keras, scikit-learn, Pandas, NumPy, OpenCV, Grad-CAM
- **LLMs & Retrieval:** LiteLLM, OpenAI API, embeddings, semantic search, prompt evaluation
- **Backend:** Python, Flask, FastAPI, Pydantic, REST and Swagger, SQL, MySQL, SQLite, Solr
- **Engineering:** Git/GitHub, code review, pytest, 100% test coverage, pre-commit, ruff, Docker, uv, Jira
- **Programming:** Python, SQL, C++, JavaScript, HTML

LANGUAGES AND INTERESTS

- **Languages:** Greek (native), English (ECPE C2 Proficiency, with honors)
- **Interests:** Strong interest in fashion and visual arts, incorporating creativity into technical projects.