Sergio Adan Bermudez, PhD

Applied AI Scientist · LLM agents, fine-tuning and evaluation

Barcelona, Spain · sadanbermudez@gmail.com · linkedin.com/in/sadanbermudez · github.com/sadanbermudez · sergioadan.dev

Profile

Applied AI scientist with a PhD in sensing and measurement from the University of Edinburgh. I design AI systems around large language models (LLMs) and other neural networks, build them quickly with AI coding agents, and measure them with held-out evaluation sets that decide every design choice: agents with tools, retrieval with checked citations, models fine-tuned on my own data, and local models tuned to the hardware they run on. Before AI I spent nine years in photonics and optical sensing, in research and in industry, the last two with customers. That mix suits applied AI, LLM engineering, AI for science and forward-deployed roles.

Projects

2026. Each project links to a live demo or its code.

Argus, my local agent system, and a llama.cpp latency fix

- A personal assistant on a local 35B mixture-of-experts model (Qwen3.6-35B-A3B, llama.cpp on Vulkan, AMD integrated GPU) inside the open-source Odysseus hub. I wrote the MCP memory server (seven tools, confined paths, no deletes), the voice pipeline, a supervisor that restarts failed services, and a learned router that sends hard jobs to Claude Code and keeps private data local.
- A weekly report card on hidden test items, with confidence intervals, compares the bare model, the assistant and the assistant with deliberation. A new prompt habit is adopted only after it also wins on a separate held-out set.
- Settings chosen by evaluation on 13 everyday tasks with deterministic checks. Turning the model's thinking off scored 13/13 at a 2.6 s median, against 10/13 at 19.8 s with it on.
- Traced a 3-second pause before each spoken answer to the Vulkan backend reading uncached GPU memory. A one-line condition cut a state snapshot from 0.61 s to 0.03 s, bringing the first spoken word to about 0.7 s with identical output. The same fix had reached upstream two weeks earlier; I found it independently.

sergioadan.dev/argus · sergioadan.dev/work/llama-cpp-stall

Caïssa, a chess training app with models fine-tuned on my own games

- A Windows and Android app for my family's chess lessons and my own tournament preparation: guided lessons, an AI coach, opening repertoire checks and opponent preparation from an index of 13.8 million games.
- Fine-tuned Maia-2 (PyTorch) on 10,143 of my online games. On my last 1,015 games, held out by date and never trained on, it predicts my exact next move 55.7% of the time, up from 52.9%, and 84.0% within three guesses.
- Measured the gap to grandmaster play with Stockfish: 21.5 centipawns lost per move in my recent games against 14.9 for players rated 2600+, and grouped my biggest losses by theme to drive training.

Private app; demo on request

Can a local model replace a frontier coding agent?

- Built 23 real coding tasks with hidden tests and gave them to four agent setups. Claude Code with Opus 5.5 solved 23, a free cloud model (DeepSeek V4.1 Flash in OpenCode) solved 22, and Qwen3.6 on my laptop solved 17, at a median 195 s a task.

sergioadan.dev/work/coding-bench

LabDesk, a documentation assistant delivered like a client project

- Took a simulated client brief (support for PyVISA instrument control) from requirements to handover: six architecture decision records, an evaluation spec, a FastAPI service with health and metrics endpoints, and a runbook.
- BM25 retrieval over 98 documentation sections. Each sentence cites its source, and citations are checked against the passages the model actually saw. Includes a read-only tool that reads a power meter.
- On a held-out set frozen before the first run: 26 of 31 answers correct, 27 of 31 citations correct, all 8 off-topic questions declined, p95 latency 4.8 s on a local model.

github.com/sadanbermudez/labdesk-copilot

photonics-lab-mcp, an AI agent that runs a lab experiment

- An MCP server with ten tools over a simulated version of my PhD optics bench. Safety lives in code: a power cap, an arm-then-emit interlock, permission tiers, a safe state on any fault and an append-only lab notebook. 22 tests in CI.
- With the answer hidden in the server, a local model planned the experiment and measured a photon order of 2.004 ± 0.03, and 1.003 ± 0.002 on a one-photon control run.

sergioadan.dev/work/lab-agent · github.com/sadanbermudez/photonics-lab-mcp

sergioadan.dev and the public Jarvis

- A hand-built static site on a Cloudflare Worker, with no framework. The build stops on any Content-Security-Policy violation or missing file, and changes go through pull requests and CI.
- A public assistant that answers from the site's pages: hybrid retrieval (BM25 and bge-m3 embeddings with reciprocal rank fusion), streamed answers with citations, Gemma 4 26B on Workers AI, and voice in and out.

- Evaluated on fixed questions plus two held-out sets: 16 of 17 on development, 7 of 8 held out. Writing the evaluation turned up five defects, all fixed.

sergioadan.dev/jarvis · github.com/sadanbermudez/portfolio-site

jarvis-chess, chess commentary checked by an engine

- An MCP server where Stockfish and python-chess compute the facts, and a claim checker marks each statement true, false or uncheckable. On the site, Stockfish 19 runs as WebAssembly in a Web Worker, and an explanation names a tactic only when the engine's line shows it.
- As a FIDE Master (peak 2402) I was the domain judge: each rule in the checker comes from a mistake I spotted in commentary on one of my own games.

sergioadan.dev/work/chess · github.com/sadanbermudez/jarvis-chess

moe-offload-lab, why a 284B model runs at 0.5 tokens per second

- Modelled bytes per token and benchmarked the SSD: it delivers 4.8 GB/s, but the engine read only 0.72. Wrote an AVX-512 FP4 kernel in C++20, profiled the engine and found the shipped binary had no AVX-512 instructions.
- Rebuilt it with clang and changed one variable at a time. AVX-512 was worth about 9%; the bigger win, about 55%, came from how threads wait at roughly 500 barriers per token on Windows. Reported upstream.

github.com/sadanbermudez/moe-offload-lab

Three Android apps, with working browser demos

- La Piña (Kotlin, Jetpack Compose): a rules engine for a trading method, with 263 tests. Spanish Tutor (Kotlin, llama.cpp through JNI): an offline tutor on a 2B on-device model. Olympus (Flutter, Riverpod, encrypted SQLite): training and nutrition, with a guard that flags any number the model states without a tool behind it.
- Ported each app's logic to JavaScript for the browser demos and checked it against the apps' own test cases. Picked the live tutor's model with a 20-turn evaluation, where Gemma 4 26B scored 20 of 20.

sergioadan.dev/apps

Experience

Independent Technical Consultant

Self-employed, Barcelona · 2026 – present · client confidential

- Technical assessment of a precision measurement instrument for a manufacturing company: what it measures, how well, and what it takes to turn it into a product.
- Market analysis: who buys this kind of instrument, the competing systems, and the size of the market.
- Business plan: phased development with go/no-go gates, team, costs and financing.
- Web design for the product's presentation.

sergioadan.dev/work-with-me

Key Account Manager and Product Specialist

Sensofar-Tech S.L., Barcelona · 2024 – 2026

- Designed and built an internal LLM assistant: an open model fine-tuned with LoRA, retrieval over the company's product and scientific documents, frontend and backend, deployed with Docker and Cloudflare.
- Led technical product support and key-account management for optical metrology systems (3D confocal and interferometric) across the UK, Ireland, Italy, Poland, Australia and South-East Europe.
- Ran workshops, demonstrations and training for technical and commercial audiences, including Rolls-Royce, GE Aviation, ST Microelectronics and Red Bull, and brought customer needs back to product and R&D.

Postdoctoral Researcher, Biomedical Photonics and Sensing

IQS, Universitat Ramon Llull, Barcelona · 2022 – 2024

- EU Horizon 2020 project Light4Lungs: measured singlet oxygen inside bacteria with ultrasensitive photon detection (photodiodes and photomultiplier tubes).
- Designed, built and validated sensing and spectroscopy systems, and coordinated the optics, biochemistry and computational-modelling teams.

Research analysis and postdoc search · Wiesbaden, Germany · 2022, two months: Analysed my PhD results and applied for postdoctoral positions.

Education

PhD in Engineering, Intelligent Sensing and Measurement · University of Edinburgh · 2017 – 2022

- Thesis: Detection of two-photon photosensitised singlet oxygen in hollow-core photonic crystal fibres. The first quantitative measurement of two-photon singlet oxygen in water, using the fibre as a nanolitre sample cell, refined over seven versions of the experiment.
- Leadership Award, University of Edinburgh (2019). Talks at the European Society for Photobiology and the Max Planck Institute for the Science of Light.

MSc in Engineering · University of Glasgow · 2016 – 2017 · first year of the Centre for Doctoral Training in Intelligent Sensing and Measurement

MSc in Nanoscience · University of Copenhagen · 2014 – 2016 · photophysics major; machine learning with Python and R

BSc in Biochemistry · Universidad Autónoma de Madrid · 2010 – 2014

Skills

AI and LLMs: LLM application engineering, retrieval-augmented generation (RAG): BM25, embeddings, reciprocal rank fusion, AI agents and tool use, Model Context Protocol (MCP), fine-tuning: LoRA, PyTorch, evaluation design: held-out sets, deterministic checks, Inspect AI, local inference: llama.cpp, GGUF quantisation, Vulkan, speech: Whisper, Kokoro, MeloTTS, observability: OpenTelemetry, Arize Phoenix

Software: Python (daily), C++20, JavaScript, WebAssembly and Web Workers, FastAPI and Cloudflare Workers, web security: Content-Security-Policy, Subresource Integrity, profiling and benchmarking, Kotlin, Jetpack Compose, Flutter and Dart; Git and GitHub Actions, Docker, Linux and Windows, R, MATLAB

Science and customers: photonics, spectroscopy, experimental design, statistics and curve fitting, instrument control (PyVISA, SCPI) and chess (FIDE Master), key-account management, technical support, workshops and training

Languages and publications

Spanish (native), English (C2), German and French (basic).

S. Adan Bermudez et al., "Designed modular proteins as scaffolds to stabilize fluorescent nanoclusters", *Biomacromolecules* 16, 3836–3844 (2015).