

Sijan Paudel

Pokhara, Nepal | Phone: +977 9846906893 | Email: sijanpaudel44@gmail.com

GitHub: github.com/sijanpaudel14 | LinkedIn: linkedin.com/in/sijanpaudel14

AI/ML Engineer | LLM & Generative AI | Agentic Systems, RAG, Fine-tuning

PROFESSIONAL SUMMARY

AI/ML Engineer with production experience shipping LLM and generative AI systems end to end: multi-agent workflows, retrieval-augmented generation, model fine-tuning, and predictive modelling on large relational datasets. Engineered 11 AI modules for a live healthcare SaaS platform, including LangGraph multi-agent pipelines, hybrid BM25 + vector RAG, and statistical behaviour analysis, cutting LLM token costs by 74% and query latency by up to 2.64x. Fine-tuned open-source LLMs (Llama, Mistral, Qwen2, HyperCLOVA X) with QLoRA/LoRA, and built offline LLM evaluation harnesses gated in CI. Strong foundation in classical ML (scikit-learn, XGBoost, SciPy), feature engineering, and EDA, with backend delivery in Python, FastAPI, and PostgreSQL.

TECHNICAL SKILLS

LLM & Generative AI: LangChain, LangGraph (StateGraph multi-agent), LCEL, RAG (Hybrid BM25 + Vector, RRF, pgvector), Prompt Engineering, Agentic Tool Use, QLoRA/LoRA Fine-tuning, PEFT, TRL/SFTTrainer, BitsAndBytes, HuggingFace Transformers, OpenAI GPT-4o, LLM Evaluation (LLM-as-judge), Guardrails, Text-to-SQL

Machine Learning: Scikit-learn, XGBoost, PyTorch, TensorFlow, SciPy (KDE), Supervised & Unsupervised Learning (regression, tree ensembles, SVM, clustering), Neural Networks, BERT, Imbalanced Data (SMOTE, Isolation Forest, cost-sensitive learning), Feature Engineering, EDA, Model Evaluation

Languages & Backend: Python, SQL, FastAPI, Django/DRF, REST APIs, Async/asyncpg, Microservices, SSE Streaming, JWT/OAuth2, RBAC

Data & Databases: PostgreSQL (pgvector, tsvector FTS), MongoDB, Redis, Vector Search, ETL, Pandas, NumPy, Apache Spark (basic)

MLOps & Infrastructure: Docker, CI/CD (GitHub Actions), AWS EC2/RDS, Azure ML, Celery, Weights & Biases, pytest, Git/GitHub

Tools: Jupyter, Postman, Streamlit, PaddleOCR, sqlglot

PROFESSIONAL EXPERIENCE

AI Engineer — Neurithum AI, AgeCare Platform

Nov 2024 – Present

Healthcare SaaS (Aged Care) | Remote | Python · FastAPI · LangGraph · pgvector · PostgreSQL · Redis · AWS · Docker

- Agentic EHR Chatbot (25-tool LangGraph pipeline):** Shipped a production dual-agent system for natural-language querying over Electronic Health Record data, where an LLM intent router (gpt-4o-mini, temp=0) dispatches to a data agent (25 async SQL tools across 11 EHR domains) or a clinical reasoning agent (4 extraction tools). Added real-time SSE streaming, PostgreSQL-persisted memory with sliding-window LLM summarisation, anti-hallucination guardrails (forced first tool call + output blacklist), and PHI audit logging on every data access. *LangGraph, LangChain LCEL, FastAPI, gpt-4o-mini, PostgreSQL, asyncpg*
- Hybrid RAG retrieval system:** Designed a two-stage BM25 + pgvector semantic search with Reciprocal Rank Fusion for clinical entity extraction from free-text notes, achieving a **74% reduction in LLM token cost** through targeted retrieval. Built the ingestion pipeline for guideline PDFs using semantic chunking and OpenAI **text-embedding-3-small** embeddings. *pgvector, tsvector BM25, LangChain, LangGraph*
- Behaviour pattern analysis — 5-layer ML pipeline:** Engineered SQL signal extraction → SciPy KDE clustering to model behaviour frequency distributions → LangGraph reasoning agent for causal hypothesis generation → RAG intervention retrieval → human-review draft. *SciPy, scikit-learn, LangGraph, pgvector*
- Clinical deterioration prediction (CDTR):** Computed patient baselines from longitudinal EHR data and detected statistical drift in vital-sign and behavioural signals, auto-generating structured clinical notes and facility-wide dashboards for proactive escalation. *FastAPI, GPT-4o, SQLAlchemy async, PostgreSQL*
- Data pipeline and SQL optimisation:** Replaced N+1 Python loops with PostgreSQL set-based views, benchmarked across 18 runs: **2.64x peak speedup**, **44.4% payload reduction**, and **75% LLM cost savings**. Designed 5 AI-specific SQL views powering unified patient timeline, clinical profile, behaviour intelligence, and compliance features. *PostgreSQL, SQLAlchemy, Pandas*
- Care plan intelligence (ICP):** Built a deterministic gap-detection engine, preference scanner, and LangGraph reasoning agent for care plan generation with compliance checking, 4-layer Redis caching, multi-tenant scoping, and RBAC across 5 roles. *LangGraph, Redis, FastAPI*
- Production security and compliance:** Implemented JWT/RBAC middleware with PHI access audit logging, OWASP security headers, sliding-window rate limiting, parameterised SQL, and LLM output guardrails blocking unsafe clinical recommendations. *python-jose, FastAPI middleware, SQLAlchemy*

Computer Operator

Oct 2022 – Jan 2024

Pokhara, Nepal

- Automated data-processing workflows with Python scripts, reducing manual processing time by **40%**

PROJECTS

- **Deep Research Agent & LLM Evaluation Harness:** Built a multi-tool web research agent with planning and tool use over web search, RAG, and structured APIs. Designed an offline evaluation harness (LLM-as-judge plus regression tests) measuring answer quality, hallucination rate, and tool-selection accuracy, then gated it in CI so prompt and model changes must pass the eval suite before deployment. *LangGraph, FastAPI, GPT-4o, pgvector, pytest, GitHub Actions*
- **Amazon Product Price Prediction (QLoRA fine-tuning):** Fine-tuned open-source LLMs (Llama, Mistral) to predict prices from product descriptions using 4-bit quantisation and LoRA adapters, giving a **90% reduction in trainable parameters**, **75% lower GPU memory**, and training time cut from 15h to 5h. Ran the full loop: base-model evaluation, fine-tuning with W&B tracking, pre/post comparison, and production validation. *PyTorch, HuggingFace, PEFT, QLoRA, BitsAndBytes, W&B*
- **NL→SQL Analytics Copilot (safe text-to-SQL):** Natural-language-to-SQL copilot over a PostgreSQL warehouse using schema-aware prompting, sqlglot AST analysis, and an allow-listed template library to enforce read-only queries, tenant scoping, and column-level guards. Added an LLM self-correction loop and a labelled NL/SQL evaluation set to track execution accuracy and failure modes. *FastAPI, PostgreSQL, sqlglot, pgvector, GPT-4o, Redis*
- **Multimodal Document Intelligence (invoices & forms):** End-to-end extraction pipeline combining OCR, vision-language models, and deterministic rule checks. Renders PDFs to page images, runs OCR for text and bounding boxes, prompts a VLM for typed fields, then verifies totals, tax, and mandatory fields with a Decimal-arithmetic rule engine; every extracted value links back to pixel-level bounding-box evidence. *PaddleOCR, GPT-4o Vision/Qwen2-VL, FastAPI, Celery, PostgreSQL, pgvector*
- **Korean Math Solver (HyperCLOVA X fine-tuning):** Fine-tuned **HyperCLOVA X 1.5B** (NAVER) on a custom Korean mathematics dataset using QLoRA and SFTTrainer on a Kaggle P100 GPU, generating a synthetic instruction-following dataset of structured question/answer pairs. *HuggingFace, PEFT, QLoRA, TRL, BitsAndBytes*
- **Nepali Recipe Generator (Qwen2 fine-tuning):** Built an English→Nepali translation pipeline to create a custom training corpus, then fine-tuned **Qwen2-1.5B** with 4-bit quantisation and **DataCollatorForCompletionOnlyLM** to train on assistant responses only; tracked runs in W&B. *HuggingFace, PEFT, QLoRA, TRL, BitsAndBytes, W&B*
- **Credit Card Fraud Detection (EDA + predictive modelling):** Full EDA on a highly imbalanced dataset (0.17% fraud rate); feature engineering with missing-value imputation, outlier handling, scaling, and SMOTE; trained Isolation Forest and XGBoost with cost-sensitive learning, reaching **73% AUC-ROC**. *XGBoost, scikit-learn, Pandas, FastAPI*
- **Movie Genre Classification (BERT fine-tuning):** Fine-tuned BERT on 42K+ movies for multi-label classification, improving a TF-IDF baseline from 52% to **60% micro-F1**, and handled class imbalance with weighted loss. *PyTorch, HuggingFace Transformers, scikit-learn*
- **Storefront E-commerce API:** Production REST API with JWT auth, Redis caching (25% latency reduction), Celery async tasks, an 85%-coverage pytest suite, CI/CD, and Docker. *Django, DRF, PostgreSQL, Redis, Celery*

EDUCATION

Bachelor of Computer Engineering
Paschimanchal Campus, IOE, Tribhuvan University, Pokhara, Nepal

May 2022 – May 2026 (Graduated)
Aggregate: 81% (Distinction)

Relevant Coursework: Artificial Intelligence, Machine Learning (Data Mining), Big Data Technologies, Probability & Statistics, Database Management Systems, Data Structures & Algorithms, Distributed Systems, Operating Systems, Computer Networks, Software Engineering, Object-Oriented Analysis & Design, Theory of Computation, Computer Organization & Architecture

LEADERSHIP & VOLUNTEERING

Microsoft Learn Student Ambassador (Beta)
Microsoft Learn

Jan 2024 – Present

- Led **15+** technical workshops on Azure AI, RAG systems, and ML deployment for **500+** students
- Developed tutorials on LLM integration with FastAPI, Azure ML, and containerised AI services

CERTIFICATIONS

- **Applied Data Science Lab** — WorldQuant University (2023)
- **Supervised Machine Learning: Regression & Classification** — Coursera / Stanford (2023)
- **Generative AI Concepts, LLM Fundamentals, AI Ethics** — DataCamp (2023)
- **FastAPI: Building Production APIs** — Async programming, Pydantic v2, JWT auth, Redis caching, Docker (2025)
- **The Ultimate Django Series (Parts 1–3)** — DRF, authentication, Celery, testing, caching, deployment (2025)