NIKET JAIN

niketj@andrew.cmu.edu | GitHub | LinkedIn | +14122848345

EDUCATION

Carnegie Mellon University (CMU) | School of Computer Science **Pittsburgh, PA**
Master of Computational Data Science **- GPA: 3.96/4.0** Aug 2024 - Present
Selected Coursework: Introduction to ML, LLM applications, Cloud Computing, Deep RL, LM Inference, DL Systems

Vellore Institute of Technology (VIT) **Vellore, India**
Bachelor of Technology in Computer Science and Engineering **- GPA: 8.96/10.0** Jul 2018 - May 2022

SKILLS

Programming Languages & Databases - Python, SQL, Java, R, Scala, C++, SQLite, MongoDB, PostgreSQL
Tools - Git, Anaconda, Azure (Microsoft AZ-900, DP-900 & AI-900 Certifications), AWS, GCP Vertex AI, Databricks, MLFlow, Langgraph, Langsmith, Docker, Redis, GitHub Actions, High-Performance Computing Clusters, Kafka, Samza
Frameworks & Libraries - Flask, Django, Pyspark, Tensorflow, Pytorch, Pytorch Lightning, HuggingFace, vLLM, Numpy, Pandas, OpenCV, Langchain, Kubernetes, SLURM, RAG, Google Agent Development Kit, MCP, CUDA

EXPERIENCE

Honeywell Inc. **Atlanta, GA**
Data Science Intern Jun 2025 – Aug 2025

- Optimized transformer model inference through ONNX export, knowledge distillation techniques, and Kubernetes HPA with PVC storage architecture for model caching across pods, achieving 2x speedup while maintaining model accuracy for production deployment.
- Developed multimodal RAG system leveraging MCP server for native image-based document processing, replacing traditional chunking/OCR workflows to reduce context length overhead and achieve 35% improvement in generation performance while maintaining visual document fidelity.
- Designed fault detection agent utilizing random forest classification on agitator sensor frequencies, winning company hackathon and reducing maintenance costs by 41% through predictive scheduling optimization.

Carnegie Mellon University **Pittsburgh, PA**
Research Assistant | Language Technologies Institute | Advisor: Prof. Carolyn Rose Jan 2025 – May 2025

- Engineered an agentic workflow leveraging Multi-modal LLMs and Code LLMs, with vLLM serving, that streamlined webpage UI editing, achieving 40% faster modifications through automated visual and code updates.

UBS **Mumbai & Pune, India**
Software Engineer Jul 2022 - Jul 2024

- Built robust ETL pipelines using Java to process enterprise-scale Credit Risk data, enhancing risk assessment capabilities, and designed financial dashboards with MicroStrategy and Power BI.
- Established scalable data infrastructure with Oracle databases, implementing data replication and optimization techniques to support large-scale financial data integration efforts, post UBS-Credit Suisse merger.
- Developed Nucleus document processing system integrating OCR and OpenAI APIs, processing 500+ financial documents monthly with 92% accuracy rate, reducing analysis time from 4 hours to 30 minutes per document.

PROJECTS

Data Attribution Benchmark for LLMs | Capstone Project | Advisor: Prof. Chenyan Xiong | CMU **Apr 2025**

- Developed DATE-LM unified benchmark for evaluating data attribution methods across training data selection, toxicity filtering, and factual attribution tasks, enabling systematic comparison of 8+ methods (LESS, MATES, gradient-based) on diverse LLM architectures from Pythia-1B to Llama-3.1-8B.
- Conducted large-scale evaluation revealing no single method dominates across tasks and established public leaderboard for community engagement; submitted findings to NeurIPS 2025. (arxiv)

Twitter Analytics on Cloud | Cloud Computing | CMU **Apr 2025**

- Built end-to-end ETL pipeline and recommendation engine using Apache Spark to process millions of tweets and optimizing database queries to serve real-time user recommendations under 200ms latency.

Enhancing LLM Math Solving Abilities via Code Generation | LLM Methods & Applications | CMU **Sept 2024**

- Implemented distributed QLoRA PEFT training of Llama (3B) & Qwen (7B) models using data parallelism across four GPUs, enabling parameter-efficient fine-tuning with reduced memory overhead.
- Evaluated model performance against in-context learning techniques (zero, 3, and 10-shot prompting), achieving improved perplexity (1.28) and accuracy (0.68) over base models.