Mohammed Pathariya

linkedin.com/in/mjpathariya | mjpathariya.com | github.com/MohammedPathariya
mjpathariya7@gmail.com | +1 (930) 333-7212 | Indiana, USA

Education

Master of Science in Data Science, Indiana University — Bloomington, IN, USA
GPA: 3.7/4.0 | Coursework: LLMs, Deep Learning, MLOps, Cloud Computing — *Aug. 2024 – Exp. May 2026*

BE in Artificial Intelligence & Data Science, Pune University — Pune, MH, India
GPA: 3.8/4.0 | ML, Data Structures & Algorithms, OS, Statistics — *Aug. 2020 – May 2024*

Experience

Graduate Research Assistant, AI in Sports Analytics — Aug. 2025 – Present
Indiana University, Kelley School of Business — *Bloomington, IN*

- Parallelized **WhisperX** transcription across **5 Slurm-managed HPC nodes**, achieving a **30x speedup** (3.5m → 7s) to process **450+ games** with near real-time latency.
- Developed a hybrid NLP feature extraction system, combining **Sentence-BERT** embeddings with a lexical validation layer (**VADER**) to benchmark deep learning approaches against rule-based baselines.
- Architected a fault-tolerant orchestration layer using **SQLite file-locking** to manage thread-safe concurrency, ensuring zero data corruption during high-volume scraping of **50+ weekly games**.

Data Engineering Intern — Jan. 2024 – Jun. 2024
Sparkwood IT Solutions — *Pune, India*

- Optimized legacy **PostgreSQL** reporting queries by implementing composite indexing and refactoring joins, reducing report generation latency by **40%** (5m → 3m).
- Resolved data discrepancies between sales and inventory reports by auditing SQL logic, identifying inconsistency in 'cancelled order' handling, and standardizing definitions via a **SQL View**.
- Maintained daily ETL pipelines using **Apache Airflow**, implementing automatic retries to handle intermittent database locking issues during peak load windows.

Projects

The Digital Forge | *Multi-Agent Systems, AI Engineering* | Code — May 2025 – Aug. 2025

- Architected a **role-based** multi-agent system that achieved an **85% pass rate** on a custom test suite of Python algorithmic challenges, benchmarking performance against single-shot LLM baselines.
- Engineered a self-healing code execution environment using **Docker** that autonomously captures runtime errors and iteratively prompts agents to debug, resolving **70%** of defects without human intervention.
- Implemented a **RAG** layer using **ChromaDB** to ground agents in library documentation, reducing syntax hallucinations, and visualized the **Chain of Thought** via a **Streamlit** interface.

LearnLoop | *System Design, Full-Stack* | Code — Jan. 2025 – Apr. 2025

- Designed a **Session-Based RAG** architecture using in-memory **FAISS** indexes, ensuring strict user data isolation and $<$**50ms** retrieval latency for course materials.
- Optimized database concurrency by enabling **Write-Ahead Logging (WAL)**, validated via **Locust** load testing to support **500+ simultaneous users** without locking failures.
- Developed a deterministic output parser using **Pydantic**, creating a feedback loop that autonomously corrects invalid JSON from GPT-4, ensuring **99%** UI rendering stability.

AudioGroove | *Deep Learning, MLOps* | Code — Oct. 2023 – Apr. 2024

- Published **Tunes by Technology** in **IEEE ICC Robins**, demonstrating that sequential LSTM architectures outperform spatial DCGANs for symbolic music generation.
- Engineered a **Bi-Directional LSTM** in **PyTorch** that achieved a validation loss of **0.78** (vs. 4.8 baseline), implementing temperature sampling to ensure diverse, real-time melodic inference.
- Architected a distributed ETL pipeline using **Dask** to process **175,000+ MIDI files**, reducing data ingestion time by **90%** (20h → 2h), and tracked 50+ experiments via **MLflow**.

Technical Skills

Languages & Frameworks: Python, SQL, Bash, JavaScript (React), Flask, FastAPI, Pydantic, Streamlit
AI & GenAI: PyTorch, Hugging Face, LangChain, CrewAI, WhisperX, FAISS, ChromaDB, Scikit-learn
Data & Infrastructure: Docker, GitHub Actions, MLflow, Apache Airflow, Dask, Locust, GCP (Vertex AI)
Databases & Analysis: PostgreSQL, SQLite, Pandas, NumPy, Vector Databases
Concepts & Architecture: RAG, Multi-Agent Systems, Microservices, Event-Driven Architecture, CI/CD, TDD