Pavan Sai Reddy Pendry

Software Engineer · Machine Learning

pavansaipendry2002@gmail.com | (785) 813-7825 | Irving, TX
pavansaipendry.dev | github.com/pavansaipendry | linkedin.com/in/pavansaireddypendry

PROFILE

Software engineer with production experience across full-stack systems and LLM/GPU infrastructure. Reimplemented DeepSeek's Native Sparse Attention from scratch in Triton/CUDA (22x faster window kernel at 64K context, A100-validated), built a from-scratch LLM inference engine (PagedAttention + continuous batching), and shipped **BabyJay** (82.4% user approval, 7,300+ courses) and **Note**, a developer-intelligence platform with 25+ REST APIs, WebSocket pub/sub, and a 24-command CLI. Strong across Python, TypeScript, PyTorch, React, FastAPI, PostgreSQL, Redis, Docker, and AWS. Published researcher (Springer, IEEE). M.S. in Computer Science, University of Kansas, May 2026.

TECHNICAL SKILLS

Languages: Python, Java, C++, TypeScript, JavaScript (ES6+), SQL, Bash

Machine Learning & AI: PyTorch, TensorFlow, scikit-learn, Hugging Face Transformers, LLM architecture & pretraining, attention mechanisms (FlashAttention, PagedAttention, GQA, RoPE, SwiGLU), Triton/CUDA GPU kernels, mixed precision (bf16), RAG, LangChain, LangGraph, multi-agent systems, fine-tuning, RLHF/GRPO, BPE tokenization, reinforcement learning (PPO), model evaluation

Backend & APIs: FastAPI, Node.js, Express, Next.js API Routes, REST, WebSocket, Server-Sent Events, JWT, OAuth, bcrypt, rate limiting

Frontend: React 19, Next.js 16, SvelteKit, Tailwind CSS, Framer Motion, responsive design, accessibility

Data & Infra: PostgreSQL (tsvector, pg_trgm, indexing), Redis, vector DBs (Qdrant, ChromaDB, pgvector), Supabase, Docker, Kubernetes, AWS (EC2, Lambda, S3), RunPod / A100 GPUs, CI/CD (GitHub Actions), Pytest, Jest, kernel profiling & benchmarking

Practices: distributed systems, system design, code reviews, testing, numerical debugging, performance optimization, Agile, Git

EXPERIENCE

University of Kansas | Lawrence, KS **Jan 2025 - May 2026**
Research Software Engineer

- Built **BabyJay** (babyjay.bot), a production AI campus assistant serving **7,300+ courses** and **2,207 faculty** through Claude, achieving **82.4% user approval** via a real-time feedback loop that drove model and prompt optimization.
- Engineered a multi-stage **RAG** pipeline (preprocessor, classifier, router, 8 specialized retrievers) over ChromaDB and pgvector, cutting average retrieval latency from 500-1000ms to 5-50ms, a **35x improvement** over pure vector search.
- Designed a **FastAPI** backend with 14+ authenticated endpoints, Supabase PostgreSQL persistence, JWT auth with per-user isolation, and a 3-tier rate limiter capped at a daily inference cost budget.
- Built 9 production **Python** data pipelines (BeautifulSoup4) with retry logic, deduplication, and schema validation, plus automated **Pytest** evaluation suites with CI regression checks; mentored **100+ students** as a graduate teaching assistant.

Note | USA (Remote) **May 2025 - Aug 2025**
Software Engineer Intern

- Built **Note**, a developer-intelligence platform capturing and analyzing Claude Code (LLM coding-agent) sessions, designing **25+ REST API endpoints** in Next.js 16 App Router covering auth, prompts, projects, search, analytics, and cross-session intelligence.
- Designed a normalized **15-table PostgreSQL** schema (prompts, sessions, projects, knowledge graph, audit log) using tsvector with pg_trgm trigram similarity for fuzzy search, composite indexes, and auto-updating search vectors via triggers.
- Built a **WebSocket** server with Redis pub/sub for real-time CLI-to-web session pairing using hashed 6-digit codes, plus JWT dual-token auth (7d access, 30d refresh) with bcrypt, token revocation, and rate limiting.
- Built a **Node.js CLI with 24 commands** (save, search, standup, report, capture, knowledge, share) and a 14-view **React 19** dashboard for browsing sessions, prompts, and analytics.

Amrita Vishwa Vidyapeetham | Kerala, India **Jun 2023 - May 2024**
Research Assistant

- Co-authored **2 peer-reviewed papers** (Springer LNNS ICT4SD 2024, IEEE i-PACT 2023) and shipped 2 production apps in Python, React, and PostgreSQL serving **500+ users** with REST APIs, JWT auth, and AWS in a 6-person Agile team; built a Bidirectional LSTM (491K parameters, TensorFlow) for next-word prediction and nutrient analysis.

PROJECTS

NSA-mini: Native Sparse Attention from Scratch | *PyTorch, Triton, CUDA, A100*

- Reimplemented DeepSeek's **Native Sparse Attention** (ACL 2025 best paper) from scratch in PyTorch and Triton: a three-branch mechanism (compression, top-n block selection, sliding window) with GQA-shared block selection and learned per-head gates.
- Wrote custom **Triton GPU kernels** (FlashAttention-2-style online softmax, group-centric sparse gather) reaching **22x faster forward** vs FlashAttention-2 on the window kernel at 64K context (A100 80GB); verified quality parity on enwik8 (2.164 vs 2.163 bpc) with a 38-test correctness harness.

mini-vLLM: From-Scratch LLM Inference & Serving Engine | *Python, PyTorch, Triton, Hugging Face, Qwen2.5*

- Built an LLM inference engine from scratch implementing **PagedAttention** (block-pooled KV cache) and **continuous batching**, reproducing the core of vLLM; reimplemented the Qwen2.5 transformer (RMSNorm, RoPE, GQA, SwiGLU), matching Hugging Face logits to under 5e-4.
- Designed a paged KV-cache allocator with per-step block recycling and an admit/decode/evict scheduler delivering **1.77x throughput** over sequential decoding (token-for-token verified); scaffolded a fused Triton paged-attention kernel validated to 7e-4 vs dense ground truth.

Reasoning SLM: End-to-End LLM Training Pipeline | *PyTorch, RunPod, A100, NumPy*

- Engineered an end-to-end LLM training pipeline (data → tokenizer → pretraining) for a math/code reasoning model: curated a **189M-token corpus** with from-scratch MinHash + LSH near-duplicate detection and Gopher-style quality filters.
- Trained a **118M-parameter** decoder transformer from scratch with a custom gated-attention block at ~33% MFU and ~146K tokens/sec on a single A100 (bf16); trained a 32K-vocab byte-level BPE tokenizer and automated A100 provisioning via the RunPod REST API.

PiqJob | *React, Next.js 16, TypeScript, Node.js, Chrome MV3, Supabase, OpenAI*

- Architected a full-stack job discovery platform end-to-end as a solo product: a Vite React SPA, a Next.js 16 SSR rebuild, a Node.js/Express REST API, a Chrome Manifest V3 extension, and a shared Supabase PostgreSQL database enforcing RLS across 7 tables.
- Built a **5-strategy ATS extraction pipeline** covering Schema.org JSON-LD, Greenhouse, Lever, Workday, and generic DOM fallback across 10+ ATS platforms, with a backend LLM proxy converting raw career-page text into normalized JSON job records.

AttentionFM | *SvelteKit, FastAPI, async Python, WebSocket, Claude Sonnet 4, Docker, RunPod*

- Built a 24/7 AI podcast platform with a fully async **FastAPI** backend handling concurrent room management and **WebSocket** multiplexing; containerized with Docker Compose (backend, PostgreSQL, Redis, Qdrant) with a RunPod serverless handler for GPU workloads.

AI City | *Python, FastAPI, TypeScript, PixiJS v8, PostgreSQL, Redis, Qdrant, Multi-LLM*

- Built an autonomous AI civilization simulator where **10+ LLM agents** live, work, commit crimes, stand trial, and die without human input - 10-migration PostgreSQL schema, Redis messaging, Qdrant vector memory, and an isometric PixiJS v8 frontend.

HACKATHONS

HackKU 2025 | University of Kansas ACM, 36-Hour Hackathon **Apr 2025**

- Shipped a cloud-based AI application in 36 hours using Claude API, Python, and TypeScript, earning recognition for the **most innovative use of AI** among 60+ competing teams.

Hack K-State 2025 | Kansas State University, 36-Hour Hackathon **Oct 2025**

- Shipped a real-time collaborative dashboard (Python, WebSocket streaming, React) automating cross-functional task routing with AI classification.

EDUCATION

University of Kansas | Lawrence, KS **Aug 2024 - May 2026**

M.S. Computer Science - Machine Learning, Algorithms, Distributed Systems, Database Systems, Computer Architecture, Software Engineering

Amrita Vishwa Vidyapeetham | India **Oct 2020 - May 2024**

B.Tech. Computer Science & Engineering - Deep Learning, Data Structures, Algorithms, Operating Systems, Computer Networks, OOP, Cloud Computing