

SAJAL REGMI

Tempe, AZ 85281



Applied AI engineer working end-to-end on language models: pretrained a 1.04B-parameter bilingual LLM from scratch in a custom C/CUDA stack, post-trained it for instruction following and tool calling, and shipped it behind a production agent platform. Four papers; open-source artifacts on HuggingFace and GitHub.

SKILLS

Languages	Python, C, C++, CUDA, TypeScript, JavaScript, SQL
Frameworks & Platforms	PyTorch, Hugging Face, vLLM, Node.js, React Native
Databases & Storage	PostgreSQL, MySQL, Redis, MongoDB, Neo4j
AI & Agents	LLM agents, tool calling, agent orchestration, MCP, RAG, GraphRAG, vector search, memory systems, evaluation harnesses, guardrails, fine-tuning (SFT), inference optimization, vLLM, CUDA, tokenizers, PyTorch
Software Architecture	Microservices, CQRS, event sourcing, message brokers, caching, REST APIs, gRPC/RPC, API security, knowledge graphs, OT/CRDT, real-time collaboration
Cloud & Infrastructure	AWS, ECS, Terraform, CI/CD, blue-green deployment, zero-trust security, infrastructure automation

PUBLICATIONS

But How Would AI Agents Run a Town's Economy? *arXiv preprint*, 2026. arxiv.org/abs/2609.11108 · hf.co/datasets/sajalregmi4/agent-town-economy

Arkios: An Open Bilingual English-Nepali Language Model Trained From Scratch, with a Devanagari-Aware Tokenizer. *arXiv preprint*, 2026. arxiv.org/abs/2608.30092 · hf.co/sajalregmi4/arkios-1b-base · [arkios-1b-chat](https://hf.co/sajalregmi4/arkios-1b-chat)

Vowel Signs Are Not Letters: A Pre-tokenization Ceiling on Multilingual Tokenizer Fertility. *arXiv preprint*, 2026. arxiv.org/abs/2608.26449 · hf.co/sajalregmi4/arkios-tokenizer · github.com/sajalregmi/arkios-tokenizer

GPT Semantic Cache: Reducing LLM Costs and Latency via Semantic Embedding Caching. *arXiv preprint*, 2024. arxiv.org/abs/2411.05276 · github.com/sajalregmi/gpt-semantic-cache

EXPERIENCE

Lead Engineer	July 2025 - Present
Arkios.ai (arkios.ai) / Karela Technologies Inc.	Mesa, Arizona, United States
- Trained Arkios-1B (arXiv:2608.30092), a 1.04B-parameter bilingual English-Nepali model, from scratch on 150B tokens using a custom single-file C/CUDA training stack; it exceeds Pythia-1.4B, TinyLlama-1.1B, and OLMo-1B on ARC-Easy and ARC-Challenge with an order of magnitude fewer training tokens. - Raised training throughput from 3.1% to 43% model FLOPs utilization on H100 through fused attention, memory-layout, and kernel work. - Ran supervised fine-tuning end-to-end: built the chat/tool-calling corpus, swept learning rate and epochs, and selected checkpoints on held-out loss, finding chat-only metrics pick the wrong checkpoint without pretraining validation. - Built the post-training data pipeline - instruction, tool-call, and safety datasets with leak-audited splits - and diagnosed an alignment-tax regression where safety data fixed refusals but degraded tool calling. - Built evaluation harnesses for pretraining and post-training: bits-per-byte on held-out sets, standard benchmark protocols, and controlled ablations against compute-matched baselines. - Built the serving path on vLLM: KV caching and bf16 kernels cut decode latency 2.4x and 3.4x respectively. - Designed and published the model's tokenizer and the research behind it (arXiv:2608.26449), released on HuggingFace with a harness that reproduces every result on a laptop CPU. - Lead engineering for Arkios, an enterprise platform for building and deploying autonomous AI agents - agent runtimes, tool calling, long-term memory, and human-in-the-loop review in one system. - Designed a tool-calling protocol and foundry layer that lets non-technical teams compose agent tools and multi-step workflows, with governance, auditability, and least-privilege scoping. - Architected sandboxed agent execution runtimes with per-tenant isolation, hard execution boundaries, and scoped access to tools and memory. - Built a memory engine for enterprise-scale ingestion, indexing, and retrieval (RAG, GraphRAG) serving as agent long-term memory. - Shipped an in-platform application builder and document/call-center modules, and led the platform's infrastructure and zero-trust security architecture.	

EXPERIENCE

Engineer

January 2026 - Present

Arkios Code (code.arkios.ai) / Karela Technologies Inc.

Mesa, Arizona, United States

- Building Arkios Code, a desktop firmware IDE for hardware engineers with integrated project workflows, flashing, serial console, crash analysis, RTOS inspection, and device debugging.
- Developed low-level, machine-independent tooling to unify debug and programming workflows across three MCU architectures, reducing per-target integration complexity.
- Designing custom debug probe hardware and host software to capture high-frequency waveforms, power measurements, and multimeter-style telemetry for board bring-up, validation, and failure analysis.
- Built C++ instrumentation and orchestration layers across probes, programmers, embedded toolchains, and lab equipment to create a common harness used by hardware engineers.
- Working on native desktop systems, debugger backends, and IDE integrations for symbol inspection, memory and register views, trace pipelines, and automated firmware development workflows.

Lead Software Engineer

January 2025 - July 2025

Lecturely AI

Remote

- Architected multi-agent LLM workflows with knowledge-graph-enhanced RAG, improving response relevance and context quality across the product experience.
- Built a React Native to native iOS bridge using Objective-C++, Fabric synchronization, and JSI bindings to enable low-latency PencilKit-based note and doodle features.
- Implemented real-time collaborative note-taking with an OT/CRDT hybrid model to maintain consistent multi-user state under concurrent edits.
- Automated testing and deployment on Amazon ECS using CI/CD, health checks, blue-green rollout strategy, and rollback automation.

Entrepreneur in Residence

May 2024 - July 2024

Creatiphe LLC

Remote

- Built and validated MVPs through rapid experiments, improving activation and conversion by 30% using structured iteration and user testing.

Principal Engineer

May 2023 - May 2024

Ticketsewa Pvt Ltd

Remote, Nepal

- Architected and led a distributed ticketing platform using CQRS, event sourcing, and microservices to support high-concurrency purchase flows.
- Designed payment orchestration across airlines, events, and transport providers using idempotency and compensation patterns for resilient recovery.
- Built a real-time seat allocation engine with Redis locks and optimistic concurrency controls to prevent overselling under demand spikes.

Product Development Intern

August 2020 - July 2021

Karyathalo IT and Web Solutions Pvt Ltd

Kathmandu, Nepal

- Delivered an e-governance platform for Kathmandu Metropolitan City using WordPress multisite and custom plugin architecture.
- Optimized MySQL indexing and Redis caching, reducing page load time from 4.2s to 1.1s under peak usage.

Game Developer & Product Lead

February 2019 - February 2020

Karela Games

Remote

- Built and shipped Unity games in C#, improving acquisition through analytics-driven iteration and product experimentation.

EDUCATION

Arizona State University

August 2021 - May 2025

Bachelors of Science in Engineering (Biomedical Engineering)

Tempe, Arizona