Job Description: Senior LLM Engineer (AWS Bedrock | Agentic AI | RAG | Prompt Engineering | Vector Databases)

Location: Kolkata On-site
Seniority: Senior (hands-on; expected to lead implementation choices and mentor others)
Team: AI Engineering / Platform Engineering
Role Type: Full-time

Role Summary

We are hiring a **Senior LLM Engineer** to build and productionize GenAI capabilities using **AWS Bedrock**. You will own prompt engineering practices, build **RAG (retrieval-augmented generation)** workflows with **vector databases**, and (where needed) enable **custom model adaptation** (fine-tuning / continued pre-training / instruction tuning, or Bedrock-supported customization approaches). You will work closely with product, backend, and security teams to deliver reliable, cost-aware, and safe AI features.

Key Responsibilities

Prompt Engineering & Evaluation

- Design, test, and maintain prompt templates for key use cases (task decomposition, tool use, structured outputs, summarization, classification).
- Build prompt guardrails: schema constraints, deterministic formatting, injection resistance patterns, and safe fallback behavior.
- Create evaluation datasets and run systematic experiments (A/B prompts, regression checks, quality scoring).

RAG & Vector Search

- Design and implement RAG pipelines: ingestion → chunking → embedding → indexing → retrieval → generation.
- Select and operate a **vector database** (e.g., OpenSearch vector, Aurora pgvector, Pinecone, Weaviate, Milvus, etc. depending on constraints).
- Improve retrieval quality through hybrid search, metadata filters, re-ranking, and caching strategies.

AWS Bedrock Implementation

- Implement Bedrock-based services (model selection, routing, prompt management, orchestration patterns).
- Optimize for cost, latency, and reliability (token budgeting, streaming, batching, retry policies, fallbacks).
- Integrate with AWS services for security and operations (IAM, KMS, CloudWatch, VPC endpoints where applicable).

Model Customization (as needed)

- Enable customization workflows: data preparation, labeling strategies, safety filtering, and fine-tuning or adapter-based approaches.
- Build and maintain a pipeline for model updates, versioning, and rollback (MLOps-lite for LLM features).
- Work with stakeholders to define acceptance criteria and reduce hallucination risk with grounding strategies.

Engineering Excellence & Security

- Build secure, multi-tenant-ready services (data isolation, least privilege, secrets management).
- Implement observability for AI features: prompt/response tracing (redaction-aware), quality metrics, drift monitoring, incident playbooks.
- Document architecture, model/prompt choices, and operational runbooks; mentor teammates.

Must Have

- Strong hands-on experience in **prompt engineering** with measurable outcomes (quality improvements, reduced error rates, structured outputs).
- Solid experience building **RAG systems** and working with **vector databases** (indexing, retrieval, filtering, relevance tuning).
- Hands-on experience with **AWS Bedrock** (invoking foundation models, orchestration patterns, production considerations).
- Strong programming skills in **Python** (preferred) and/or JavaScript/TypeScript/Java for service integration.
- Experience designing **evaluations** for LLM features (test sets, automated scoring, regression testing, human-in-the-loop reviews).
- Understanding of LLM failure modes and mitigations: hallucinations, prompt injection, data leakage, unsafe outputs.
- Ability to ship production-quality systems: reliability, latency, cost optimization, and monitoring.

Should Have

- Experience with embeddings, chunking strategies, hybrid retrieval (BM25 + vectors), and re-ranking approaches.
- Experience integrating LLMs into real workflows: tool/function calling, agentic patterns, state management, and workflow orchestration.
- Familiarity with model governance practices: dataset lineage, privacy-by-design, PII redaction, access controls.

- Experience with AWS-native choices relevant to GenAI (OpenSearch, S3, Lambda/ECS/EKS, Step Functions, DynamoDB, CloudWatch).
- Basic understanding of ML model training lifecycle and fine-tuning constraints (even if not training from scratch).

Nice to Have

- Hands-on experience with **fine-tuning / instruction tuning** open-source LLMs (e.g., Llama-family) and serving them (vLLM/TGI).
- Familiarity with Bedrock Knowledge Bases / Agents (if used) and trade-offs vs custom RAG pipelines.
- Experience building internal prompt libraries, governance workflows, and prompt review processes.
- Exposure to security testing for GenAI (red teaming, jailbreak testing, prompt injection testing).
- Experience with languages and NLP tasks relevant to Indian and European contexts (multilingual content).

Experience & Qualification

- Typically **5–10+ years** engineering experience, including 2+ years in applied AI/LLM systems (flexible for strong candidates).
- Bachelor's/Master's in Engineering/CS (or equivalent practical expertise).

How to Apply

Please share your resume and (if available) one or more of the following:

• (a) links or short summaries of AI systems you built (RAG, agents, evaluation harnesses, or GenAI features in production), including your role and what you optimized for.

• (b) a brief tool/platform evaluation you led (e.g., Bedrock model choice, vector DB selection, retrieval strategy, guardrails/evaluation approach) and the decision outcome.

• (c) a short write-up explaining a technical trade-off you made (e.g., quality vs latency vs cost, safety vs usability, retrieval depth vs speed) and how you measured it.