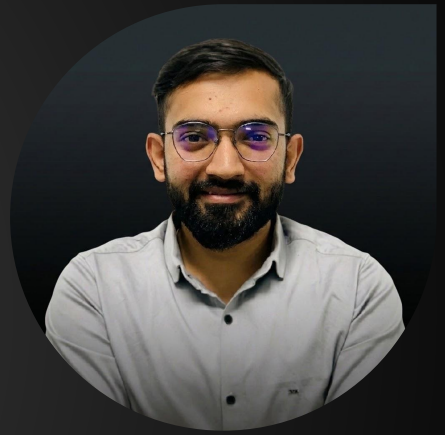
Bypassing the Cloud: Building an Offline Assistant with Gemma

By: Savan Padaliya

Meet The Presenter

I’m **Savan Padaliya**

- An Associate Backend Lead At VE3 Global
- Builder, Problem Solver
- Organiser at GDG Rajkot



Gemma 101: Free, Private, and Offline

- What is Gemma?
- Gemma Family!
- Running bigger models with less GPU!!
- Why LLM’s need chat template?
- Live Demo!!!

Why Gemma?

Data Vulnerability

Financial Overhead

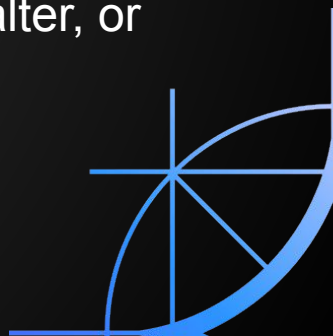
Network Latency

What Is Gemma?

- **Gemini DNA:** Built using the same research, technical infrastructure, and safety alignments as Google's flagship Gemini models.

- **Open Weights Philosophy:** Google provides public access to the fully trained model parameters, allowing for deep customization.

- **Commercial Flexibility:** Developers can freely deploy, self-host, alter, or fine-tune Gemma models across varied hardware stacks.



Gemma Family!

Introducing Gemma 4

E2B & E4B

- A new level of intelligence for mobile and IoT devices
- Audio and vision support for real-time edge processing.

12B, 26B, 31B

- Frontier intelligence, efficient performance Advanced reasoning for IDEs, coding assistants, and agentic workflows.
- These models are optimized for consumer GPUs — giving students, researchers, and developers the ability to turn workstations into local-first AI servers.

Running bigger models with less GPU!!

- Gemma 2B contains two billion active weights.
- Stored at raw standard precision (FP16), loading the model requires roughly 4.0 GB of VRAM.
- The Bottleneck: Running uncompressed models leaves zero memory headroom for token context processing, causing immediate Out-of-Memory (OOM) browser crashes.

Running bigger models with less GPU!!

The Concept: Quantization scales the precision of neural network weights to reduce their memory footprint.

Running bigger models with less GPU!!

The Math: High-fidelity 16-bit floating-point decimals (FP16) are compressed down to highly efficient 4-bit integers (INT4).

Running bigger models with less GPU!!

The Optimization: Model size drops from 4.0GB to a lean 1.4GB with negligible loss in reasoning accuracy.

Running bigger models with less GPU!!

Hardware Gain: Smaller data types allow consumer GPUs to process mathematical matrices faster, accelerating tokens-per-second.

The Mechanics of Chat Templates

- Base LLMs operate as pure predictive text completion auto-engines.
- They require formal data formatting to distinguish between user intent and model replies.
- **Control Tokens** define structural boundaries within the multi-turn context window:

<start_of_turn>user [User PromptPayload]<end_of_turn> <start_of_turn>model

Take a look at Live Demo

Web Gpu Demo

Take a look at Live Demo

Offline Gemma Demo

Get Started With Building

JS Boiler Plate

Python Boiler Plate

Thank you!!

Be In Touch

Linkedin

Portfolio

Topmate