crossml

Taking GenAI and LLMs from POCs -> Production

*Yes, It's definitely a tough question to answer. Buy **Why**?*

Challenges

Business:

1. Data Privacy and Security
2. Fairness & Responsible AI
3. Alignment with Business Goals
4. Value Proposition Clarity

Technical:

1. Skilled Workforce
2. Data & LLMOps Pipelines
3. Throughput & Latency
4. Evaluation is Challenging

*So does it end the road? Actually **No**!*

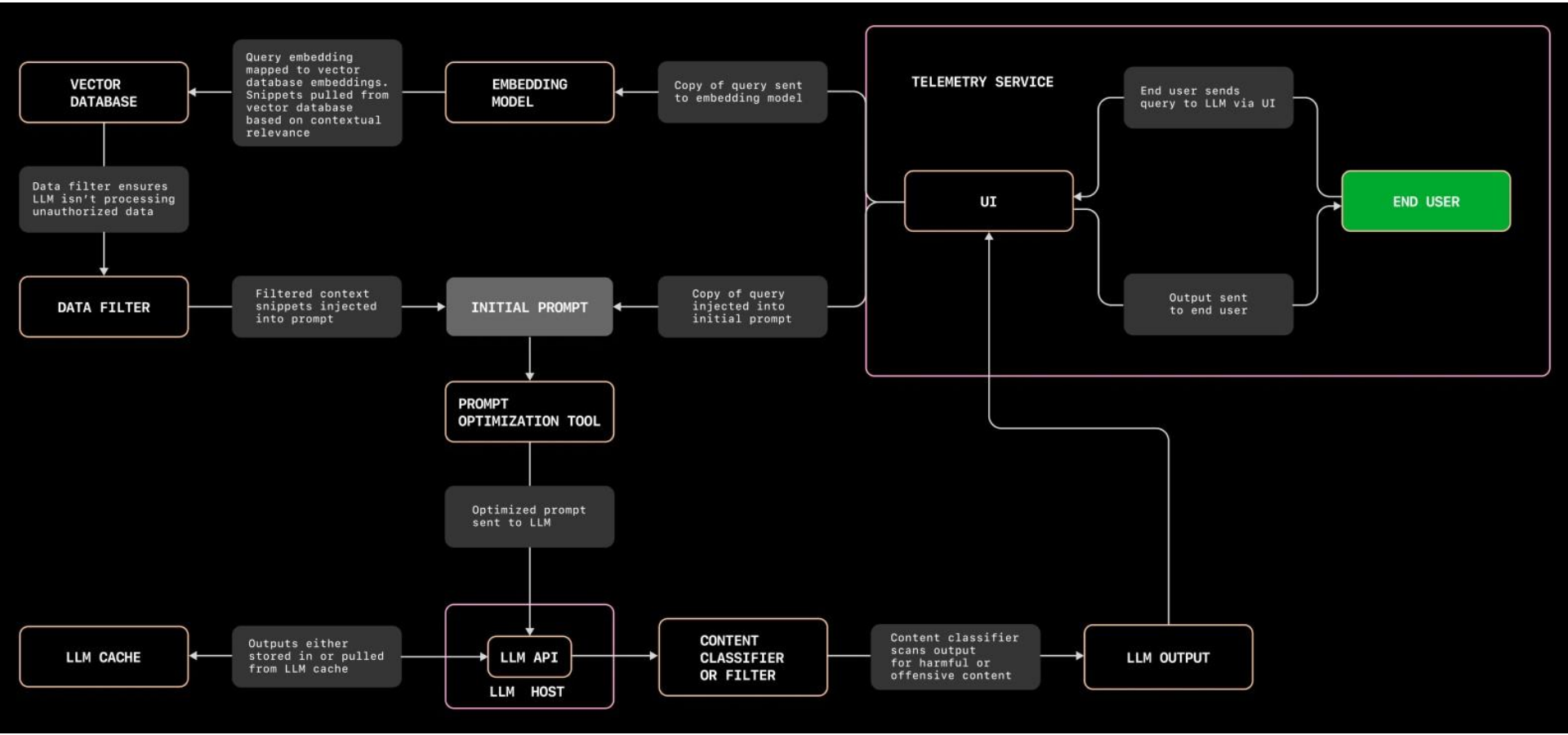

Image Source: GitHub

LLMs Optimisation - For Prototype to Production

1. **Prompt Engineering**: Tailoring the prompts to guide the model's responses.
2. **RAG**: Enhancing the model's context understanding through external data.
3. **Fine-tuning**: Modifying the base model to better suit specific tasks.

A very common mistake:

We think that this is a linear process and should be done in that order!

Instead, we should think this as a two axes depending on where you view the issues to be:

Context optimization – The model does not have access to the right knowledge?

LLM optimization – is the model not generating the right output in a particular style or format?

Is Fine-Tuning a LLM Worth?

We often heard that prompting is enough and fine-tuning **complicated**, it's **expensive**, it's **worthless**, and many other reasons. Reality says fine tuning can get a model performance much better than GPT4 and up to 30x cheaper!

LORA (Low Rank Adaptation) Popular method for Parameter Efficient Fine-Tuning (PEFT) of LLMs:

1. Reduces trainable parameters and memory usage while maintaining comparable performance.
2. 4-bit LoRA fine-tuned models could outperform base models (including GPT-4).
3. Investigated the best base models for fine-tuning per use case.

LLM Inference is Complex

LLM inference **engines** and **servers** are designed to optimize the memory usage and performance of LLMs in production.

Inference engines such as **vLLM** and **TensorRT-LLM** are occupied with great features:

- Paged Attention
- Efficient KV Caching
- Dynamic Batching
- Concurrent Model Execution

Evaluation is Challenging

Solid evaluation framework is core of any successful Generative AI application and it is not as easy as it sounds because of:

1. Answer Relevancy
2. Faithfulness
3. Context Precision
4. Answer Correctness

There are four primary LLM evaluation methods: **Vibe Checks**, **Human Evaluations**, **Benchmarks**, and **LLM-as-a-Judge**.

Security and Guardrails

Address potential security risks and challenges associated with the development, deployment, and use of GenAI-based applications such as:

1. Prompt Injection
2. Insecure Output Handling
3. Data Leakage or Poisoning
4. Model Theft
5. Denial of Services
6. Overreliance

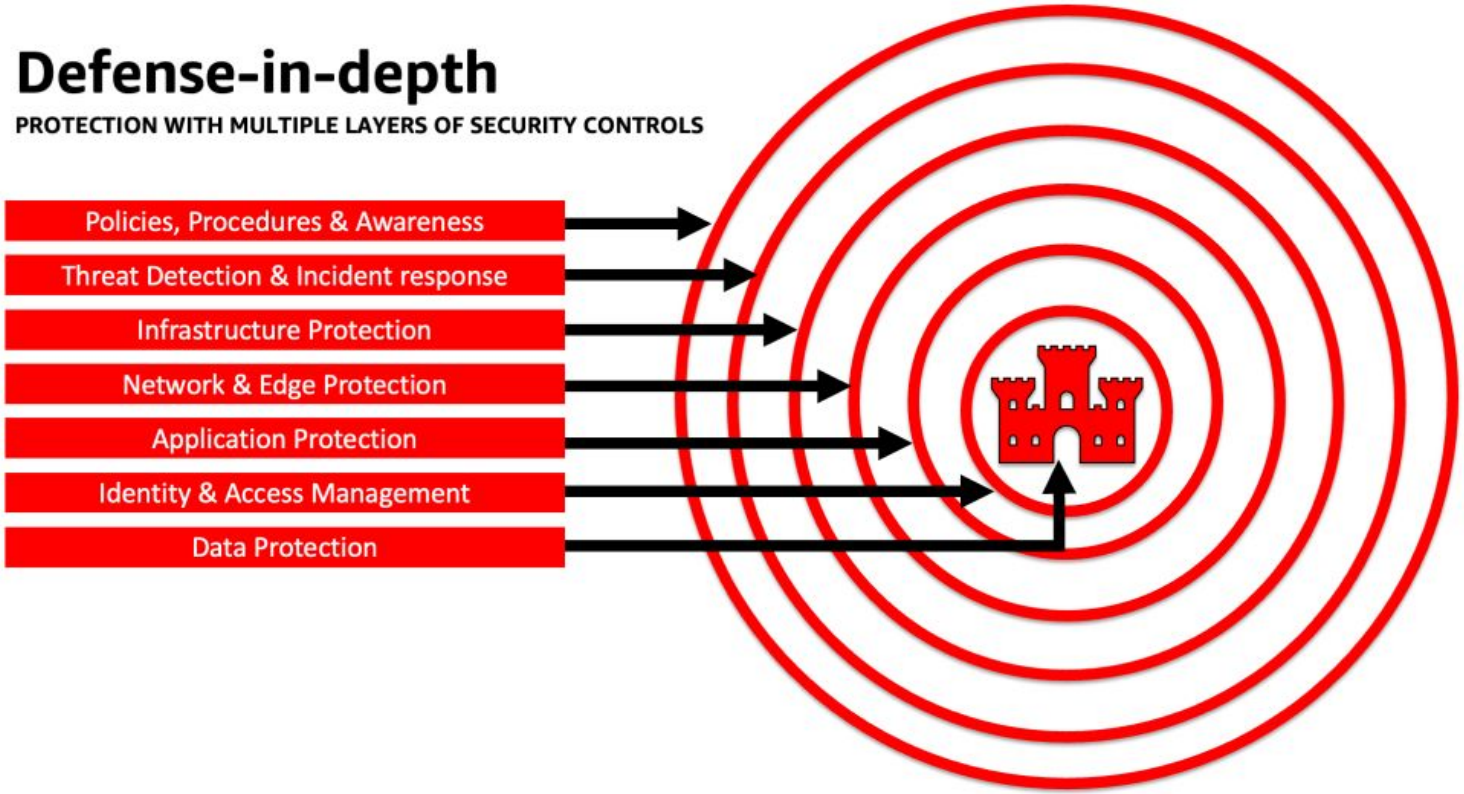

Image Source: AWS

Private LLMs

The primary purpose of Private LLMs are user privacy and data protection. But the question is do we really need this due to one of the following common reasons:

1. Data Privacy & Security
2. Reduced Dependency
3. Maintaining Regulatory Compliance

Industries where Private LLMs seems relevant:

- Legal Firms
- Healthcare Industry
- Government Agencies

Private LLMs

A typical project flow for a Private LLM would include:

1. Discovery
2. Data Preparation
3. Model Training
4. Evaluation
5. Deployments
6. Maintenance

Question to Ask:

Does your business really need a Private LLM ?

Thank You!

Ankit Aggarwal

Co-Founder at CrossML Pvt Ltd

Leading the AI Revolution | AI Awards Winner | Top Voice

LinkedIn: https://www.linkedin.com/in/ankitaggarwal1990/

I'm a driven entrepreneur with around two decades of experience in Technology.

I believe in the magic of technology - its power to transform, inspire, and bring about change. I'm fascinated by the possibilities of Artificial Intelligence, Digital Transformation, Cloud and its transformative impact on businesses.

My passion lies in nurturing innovation, building businesses, fostering entrepreneurship, and driving customer success.