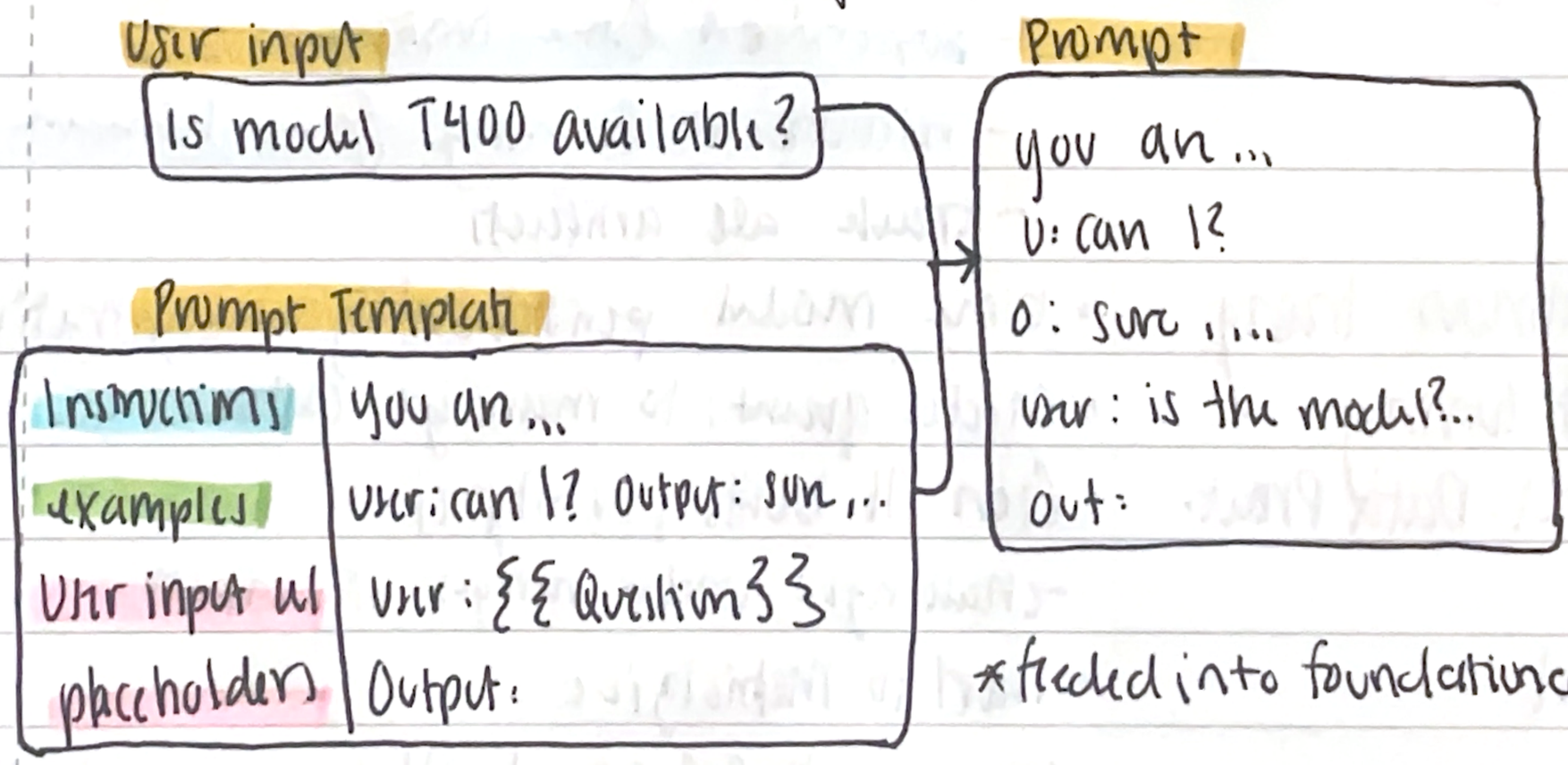
[Day 57] - "MLOps for Gen AI" (Unit 5)

"Operationalizing Gen AI or ~~non AI~~ using MLOps" Podcast

- Agent MLOps - new frontier
- DevOps: making software dev smoother & faster
- MLOps: same principle but applied to ML
  - challenges: data valid, eval. models, drift, nproducibility, etc..
  - Gen AI life cycle phases
    - Discovr: no one size fits all model depends on use case & quality, latency, legal/comp $
    - Dev & exp: iteration, refining data, eval results & going back to rework
    - Eval: can be automated once mature enough (ex AutoSS)
    - Deploy:
    - Govern:
  - Emphasis of tuning and adapting models
  - SOT: whole model guard - collection of all models.

Foundational Model Paradigm

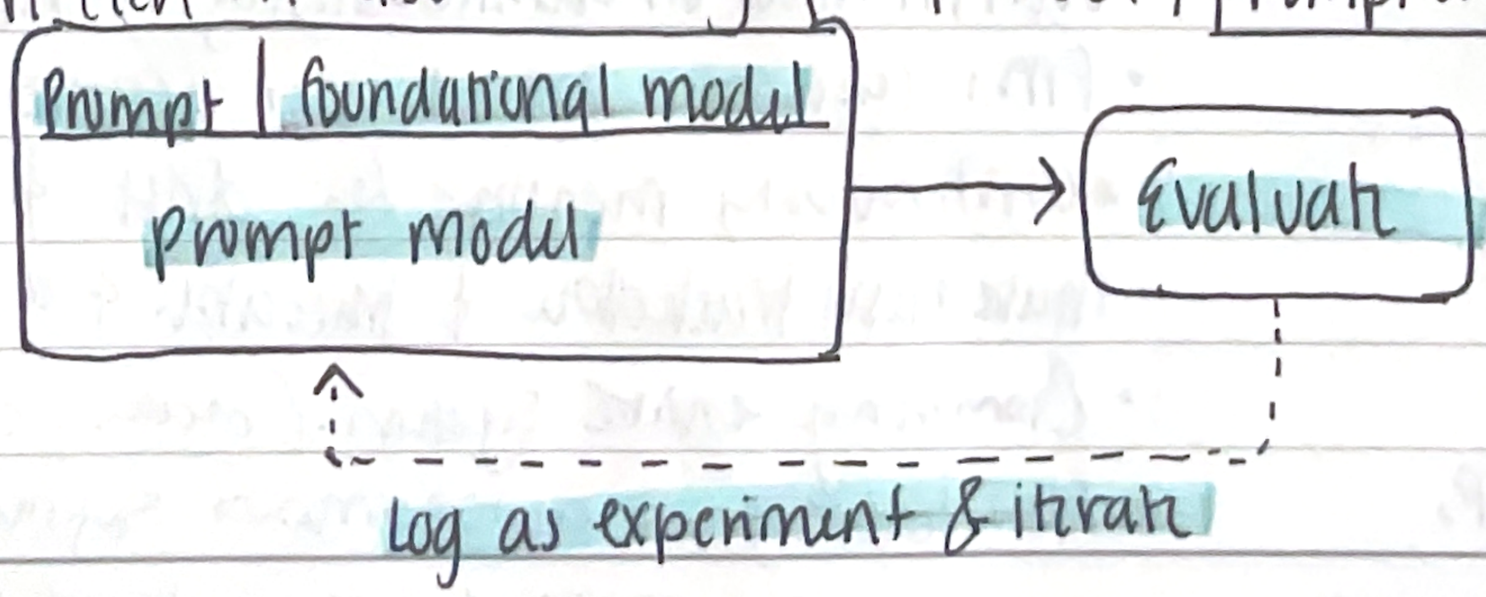
- Different from trad. prediction models:
  - multipurpose, not trained for a specific tasks & display emergent properties
  - input = everything, sensitive to prompt
- Prompted model comp using <u>prompt template</u>



*fed into foundational mt

Prompt is Data & Code

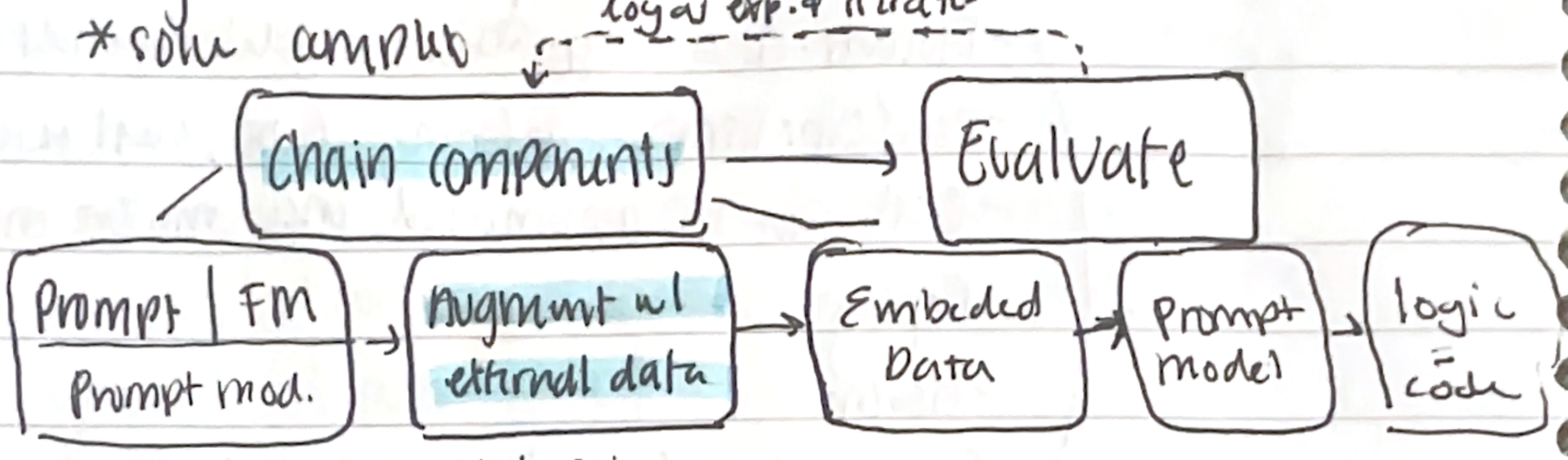
- In Gen AI: also tweaking prompt itself, <u>prompt engineering</u>



Prompt = Data
- Fewshot ex, any knowledge base, prompt = data

Prompt = Code
- Instruction, template, needs version control etc
  * track which versions of prompts work best for which versions of models for reproducibility.

Chain & Augment
- connecting multiple prompted model components together along w/ calls to external API. etc.
  * solve complex



Gen AI chain
- Agents use LLM as brain.
- Nature of inputs is much harder to do upfront
  - sol: think about entire chain as one unit
  - eval & versioning = more complex

Vertex AI
- Grounding (retrieval) → great for chaining

LangChain
- Open source framework

Tuning FM
- Adapting to perform better on task/domain
  - supervised fine tuning
  - reinforcement learning from human feedback
  - track all artifacts

Continuous training & tuning
- tune model periodically
- synthetic data
- Model garden: to manage costs

FUEL: Data Prod.
- Gen AI builds prototypes
  - challenge: wider ranges of input

Eval
- Hard to Metricize
- essential to safeguard AI apps against Adversarial attacks)

Deploy
- Gen AI: focus on operationalizing a complete sys. for specific use
- FM: centers around user accessibility
  * continuously monitor for drift & skew

Govern:
- Make easy trackable & traceable & accountability
- Governing entire system. (data, code, prompts)

Agents OPs
- Now dealing w/ autonomous systems.
  - require solid frameworks & trust: tool orchestration + registry

[Live Stream]

Presentation

- Why it takes long to dev. an agent?

Repo = Quick Start

~kaggle agent-starter-pack

Pop Quiz

- not a core practice of MLOP's? → C. training model from scratch
- Prompt temp. in context of GenAI? → set of instructions (B)
- Purpose of chaining? → B. avoid hallucination & maintain accuracy
- Eval = critical step in dev of GenAI → D. Measure quality better.
- which vertex AI offers managed exec of eval. jobs in prod., skew, and drift ? → B. Pipelines.

[Capstone]

Goal: create notebook demo. use case using GenAI capabilities: agents, vector databases, embeddings, or all above.

* Bonus: create blog./YT vid.