CSCI E-103

Data Engineering for Analytics to Solve Business Challenges

Developing & Deploying LLMs & Agents

Lecture 11

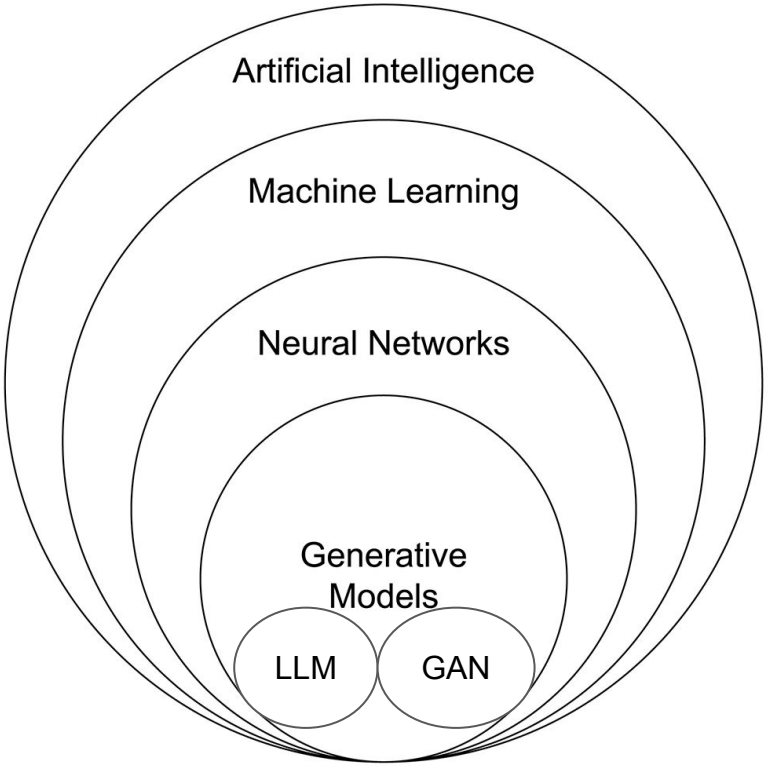
Anindita Mahapatra & Eric Gieseke

Harvard Extension, Fall 2025

Agenda

- Announcements
  - Quiz-1, Assignment-3 grades posted
  - Use Case-1, Assignment-4 will be graded soon
  - Quiz-2 will be posted soon
  - We are on the final stretch
  - Final Project will be release on Nov 18th - Please create your slack groups & let us know which challenge you chose
  - No section on Thursday due to Thanksgiving break
- Data & AI - two words in one breath
  - Software eats the world & AI eats Software ….
- GenAI - Yet another disruptive technology
- What happens to the older models?
- What use cases benefit from LLM?
- LLM Maturity curve
- How to choose the right first use case
- RAG
- LLM Ops
- Additional considerations when dealing with Gen AI product
- Lab
  - Demo LLM application

Evolution of AI



- Rule Based Systems
- Classical ML
- Deep Learning (unstructured data)
- Gen AI
  - Large Language Model (LLM) - text
  - Generative Adversarial Network (GAN) - images

Traditional AI aims to perform specific tasks based on predefined rules and patterns.

Generative AI goes beyond this limitation and strives to **create** entirely new data that resembles human-created content.

Why are LLMs so powerful?

LLMs are very capable because they are trained on massive amounts of data - giving them a grasp of how language works and a significant amount of knowledge

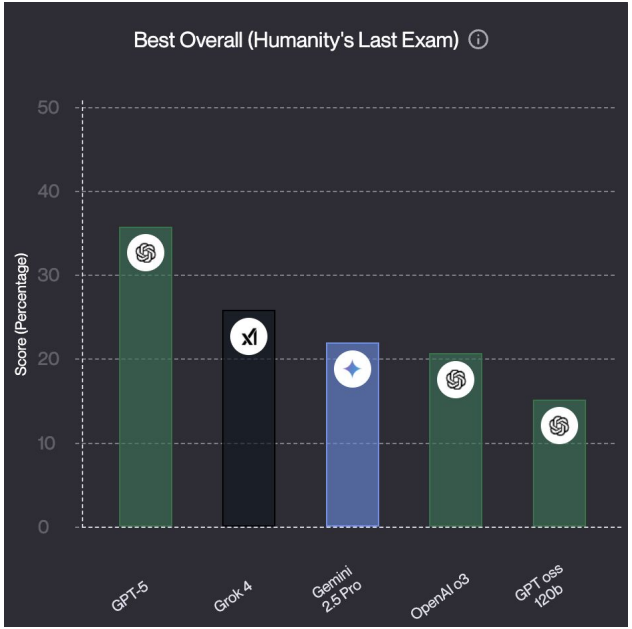
- Training data such as “all text on the internet”

LLMs excel at language related tasks, such as:

- Answering questions or chatting
- Summarizing longer form content
- Writing computer code such as writing SQL or HTML or Java
- Generating content such as marketing copy and legal documents
- Translation

There are 1000s of different LLMs, each with different skills & capabilities

- GPT family (e.g. ChatGPT), Grok, Gemini, Claude



https://www.vellum.ai/llm-leaderboard

ML/AI has been around for a while - why should I care now?

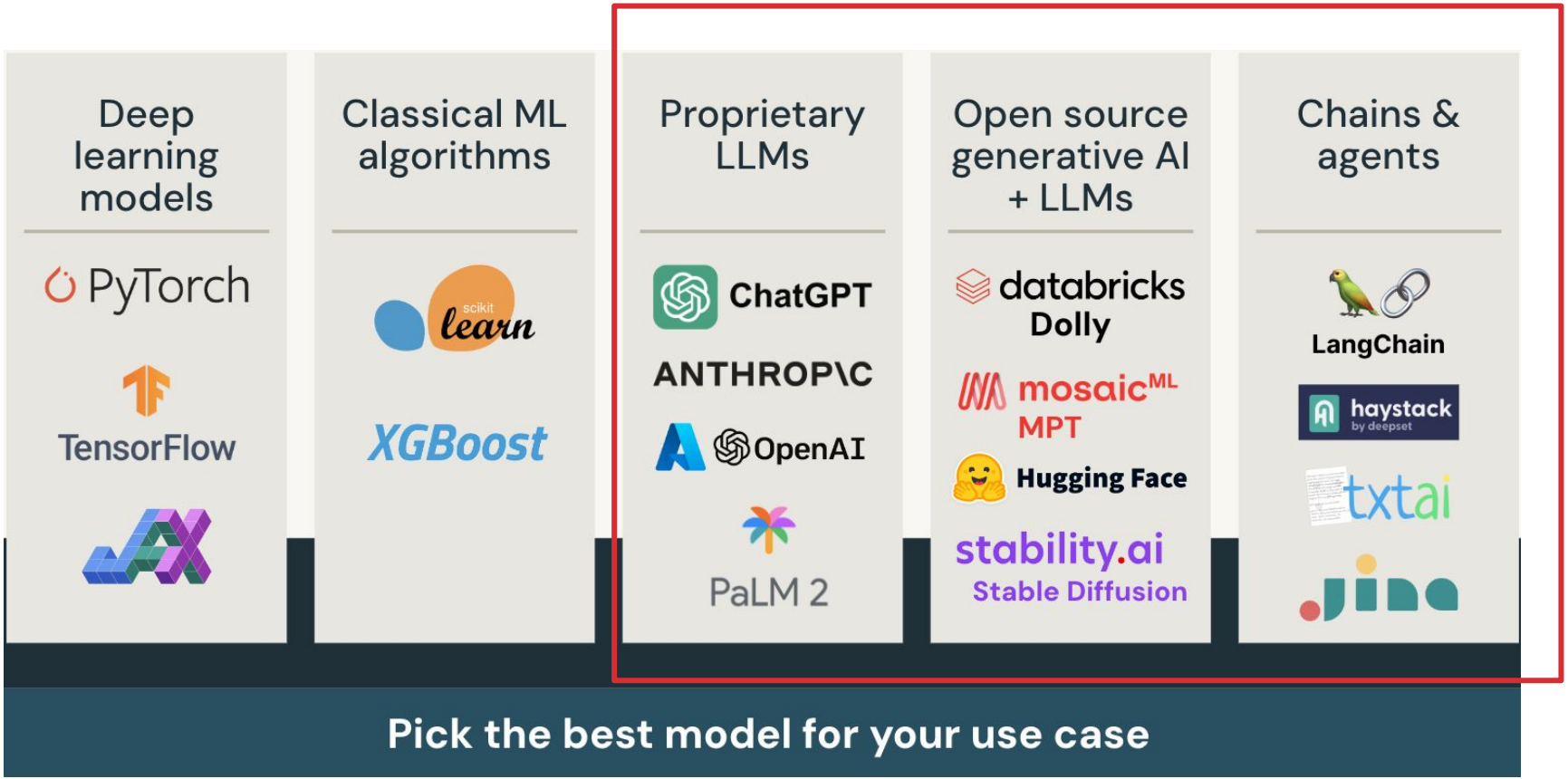
LLM accuracy and effectiveness has hit a tipping point

- Powerful enough to enables use cases not feasible even a year ago
- Yet economical enough to access and use - even by non-technical business users
- Artificial General Intelligence (AGI) and Super Intelligence may be reached as soon as 2027!

LLMs and tooling are readily available

- Many LLMs are open source and customizable
- Requires powerful GPUs, but are available in the cloud

LLMs are an addition to the existing ML arsenal



Gen AI Terminology (I)

- **LLM** - Large Language Model (NLP and beyond)
- **GAN** - Generative Adversarial Network (images)
- **Diffusion** - simulate the dynamics of complex systems over time (Lip sync)
- **Foundational LLM** - a pre-trained lang model that is the starting point for more specific models
- **Hallucination** - a confident response by an AI that it has not been trained on (Temperature)
- **Grounding** - process of associating words with their real-world entities and concepts.
- **Prompt Engineering** - process of designing effective NL prompts for use with LLMs
- **Zero-shot Learning**: An input text + prompt that describes the expected output from the model
- **Few-shot Learning**: Zero-shot + few examples of in/out

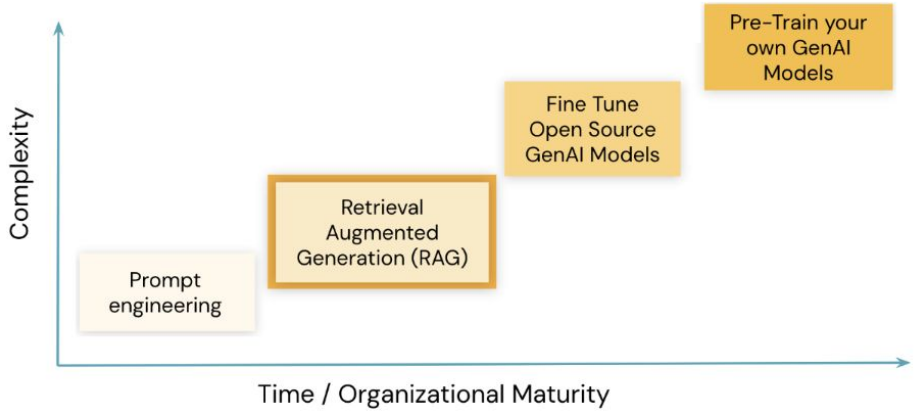
Gen AI Terminology (I)

- **Chain of Thought**: improves the reasoning ability of LLMs by prompting them to generate a series of intermediate steps that lead to the final answer of a multi-step problem.
- **Modality** - Multiple types of data - text, image, audio, video
- **Transformers**: NN arch for NLP - Encoder, Decoder, Embedding(transform from high dim to lower) (basis for GPT, BERT, T5)
- **Tuning**:
  - Instruct Tuning (e.g., prompts)
  - Fine Tuning
- **RLHF** - Reinforcement Learning with the Human Feedback
- **RAG** - Retrieval Augmented Generation

Some Examples of How do LLMs enhance use cases

Data Q&A: democratize access to knowledge	Simplify structured insights about unstructured data	Improve efficiency of knowledge worker's basic tasks	Improve existing machine learning models
Enable call center staff to ask questions of all previous support tickets Let users ask which Delta table best meets their analysis needs	What are the 5 top issues based on the call center transcripts this week Which customer reviews mentioned an issue with defects? Has that spiked in the last 2 weeks	Ask a data question, get a draft SQL query Describe a landing page, get draft HTML code Automated personalized marketing messages Create legal documents	Include customer forum posts in our fraud models Tune our product recommendation model based on customer's written feedback

LLM Maturity Progression



LLM Type	Word Volume	Quality	Cost	Latency	Privacy
Prompt Eng	(No domain data)			No control	
RAG	100s of K				
Fine Tune	Millions/Billions				
Pre-Train	Billions/Trillions			Max control	Most secure

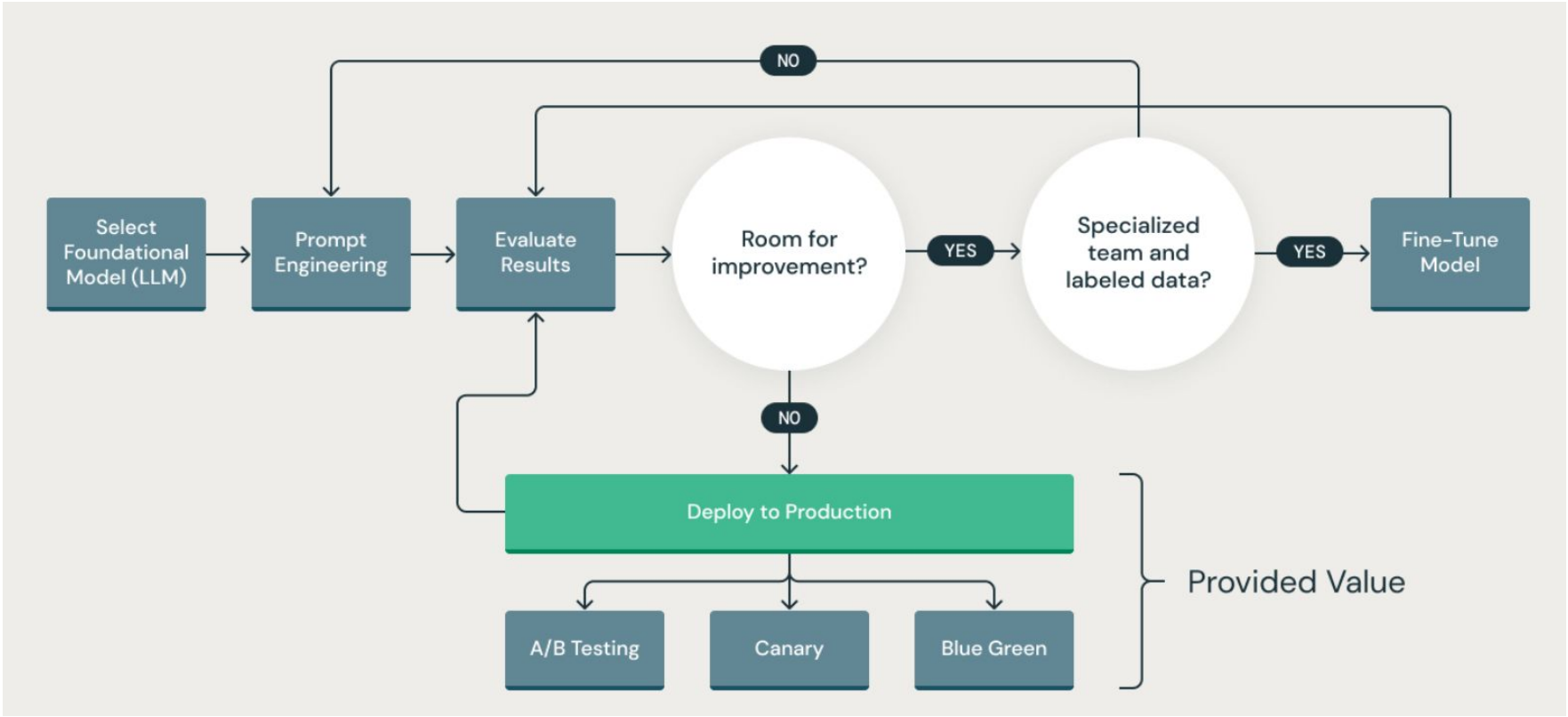
Challenges implementing LLMs

- Need to move quickly
  - Your competitors are also jumping into LLMs, and you need ensure you aren't left behind your peers—how to quickly tackle high value use cases?
- Need to customize, control, and secure your LLMs
  - Using proprietary SaaS LLMs requires you to send your data to 3Ps and may leave you without a competitive edge. How to customize LLMs that you own & control with your proprietary data?
- Need to connect LLMs with your existing data
  - Just like other forms of machine learning, LLMs require a tight coupling with your existing data strategy—how to best connect LLMs with all your existing data sources?

Known Limitations of LLMs

- Hallucination (can be controlled by temperature, prompt)
- Bias (limited or biased training data)
- Adversarial Tokens (inaccurate tokens fed to cause malfunction)
- Malicious content authoring and social engineering
- Train an LLM for Malicious Reward Hacking or train an LLM for Malicious Reward Hacking - LLMs have the possibility of finding loopholes in real world systems, but rather than fix them, might end up exploiting them.

An iterative journey of refinement and improvement



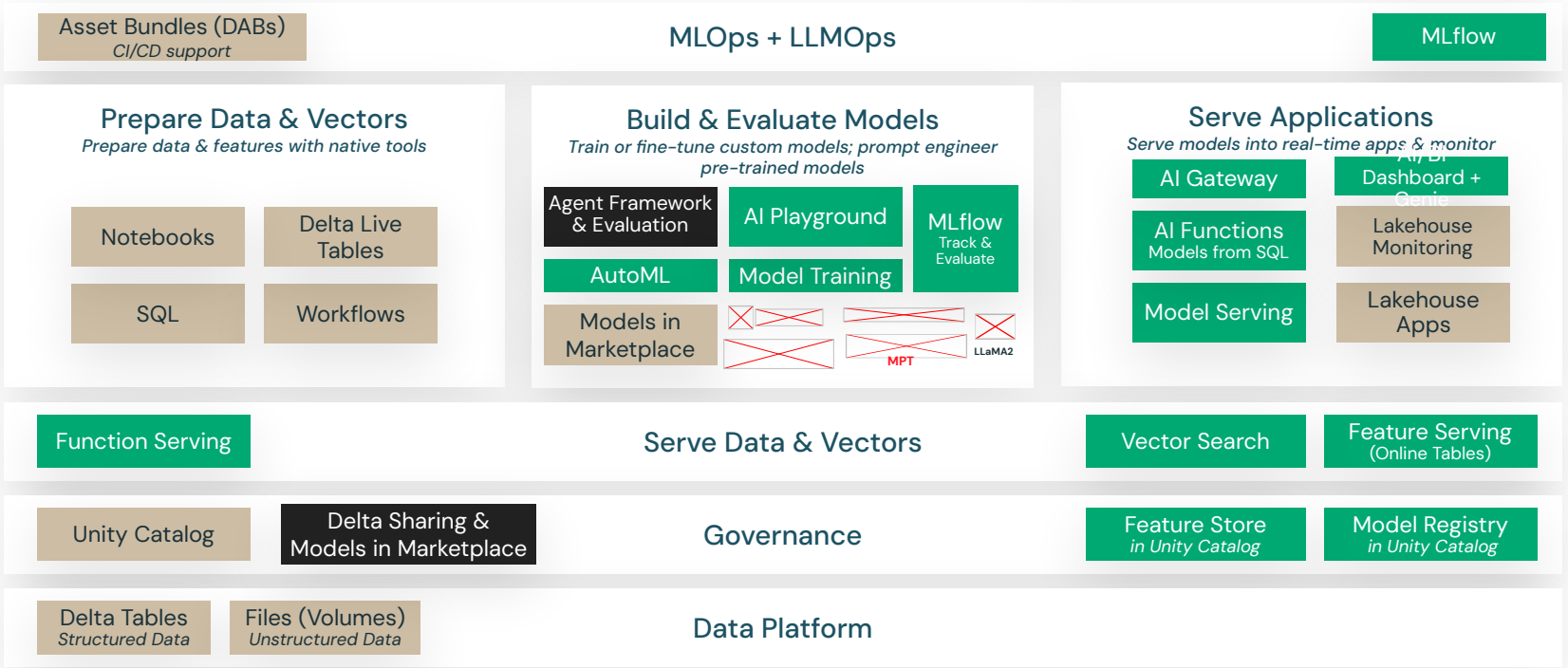
Demo Application

Mosaic AI

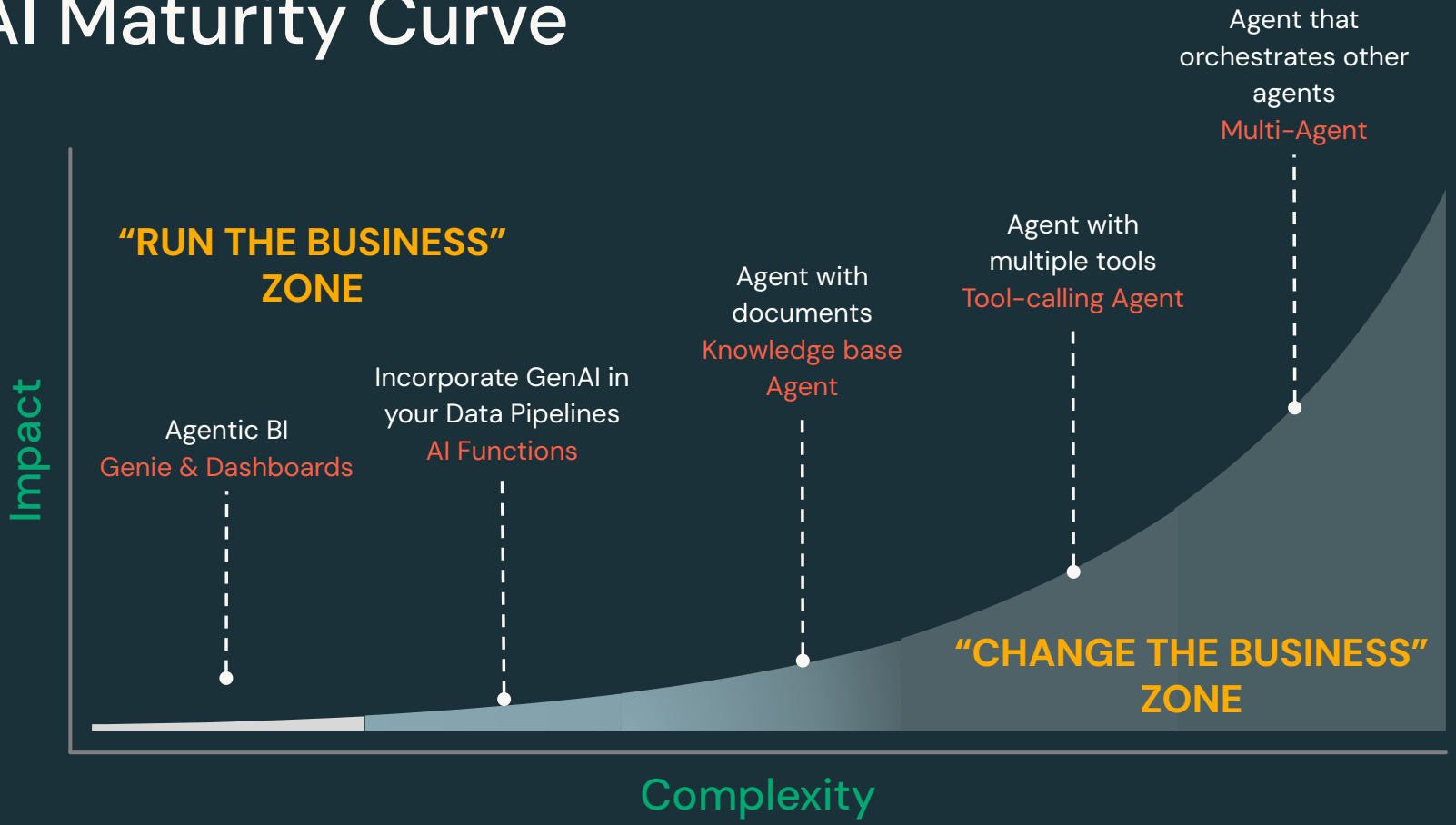
GenAI fully integrated into the Lakehouse

Lakehouse capability (Data + AI)

Mosaic AI capability (AI)



AI Maturity Curve



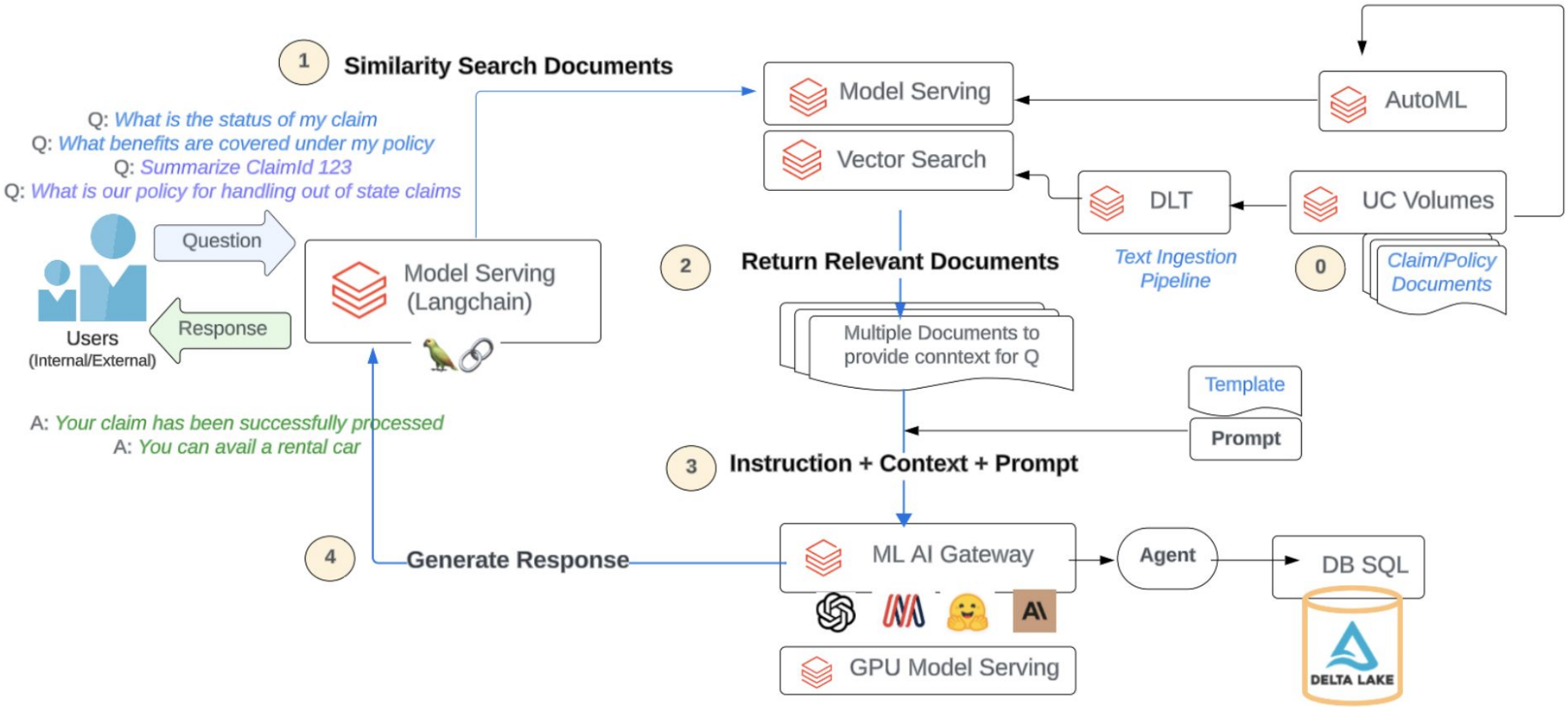
RAG - Retrieval Augmented Generation

RAG is a potential solution for LLM limitations

- LLMs are “stuck” at a particular time (of training) - It is not feasible to update their gigantic training datasets - but RAG can bring them into the present.
- LLMs are trained for generalized tasks, meaning they do not know your company’s private data.
- It’s not easy to understand which sources an LLM was considering when they arrived at their conclusions.
- Few organizations have the financial and human resources to produce and deploy foundation models.

RAG is one of the most cost-effective, easy to implement, and lowest-risk path to higher performance for GenAI applications.

Transform documents into a Knowledge Engine for Q&A



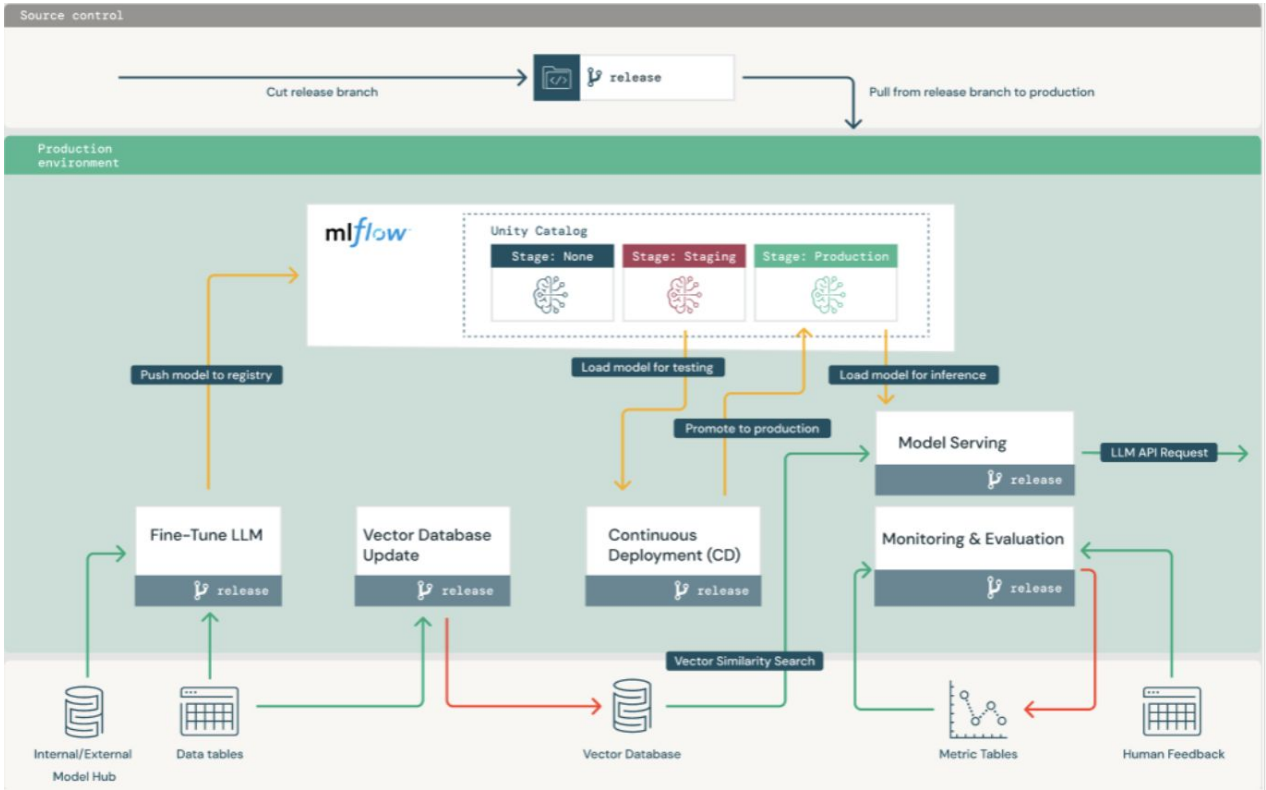
DataOps + MLOps + GenAI = LLMOps

LLM Operations for end-to-end production

- Databricks unifies LLMOps with traditional MLOps & DevOps
- Teams need to learn mental model of how LLMs coexist with traditional ML in operations

Differences to MLOps

- Internal/External Model Hub
- Fine-Tuned LLM
- Vector Database
- Model Serving
- Human Feedback in Monitoring & Evaluation



Use Case Selection Criteria

Lighthouse Use Case Desired Outcomes

- ★ Learning
- ★ Testing
- ★ Templating
- ★ Piloting
- ★ Creating excitement
- ★ Educating employees

- ❑ **Data accessibility and quality**
- ❑ Ease of implementation/adoption
- ❑ Reusability to other problems
- ❑ Expertise of AI org

The usual suspects & more …

- Potential for biased predictions
  - Gender, age, ethnicity
- Risk of misuse (more Black box)
  - Explainability is imp
  - Generation of harmful content (toxicity)
  - Hallucination
- Breaches of Privacy
  - Regulatory/Compliance
  - IP
  - GDPR

Discrimination, exclusion, and toxicity
Information hazards
Misinformation harms
Malicious uses
Human-computer interaction harms
Automation, access and environmental harms

Laws can be enforced but not Ethics

As a society, we have a collective moral responsibility

Focus on ESG score of a company- (Env/Social/Governance) By incentivizing organizations to prioritize fairness, transparency, privacy, and accountability, policies contribute to building ethical LLMs that benefit society as a whole.

Potential Safety Nets & Band-aids via iterative process

- Collect Interaction details
- Model Monitoring (Output/Results)
  - Automatic ML Scoring
  - RLHF
- Guardrail Models
  - To vet training data & possibly responses from ML
- Careful Prompt Design
  - Explicit instructions to be factual
  - Set temperature to be 0
- EU AI Act - Regulated by Disclosure of Data, Compute, Model, Deployment

Key Takeaways

- Every organization will be a Data & AI company in the future
- ‘Your Data’ & ‘Your Model’ will set you apart from your competition
- There are various levels of complexity of LLMs & You can adopt the one that best suits your needs and maturity
- A repo of narrow purpose fit LLMs are more useful to an organization as compared to a hunking large one for now
- Models are improving and it is getting cheaper to create them, so it is possible to have own foundational model in the near future
- LLMs in educational context can help promote plagiarism- but benefits far outweigh
- LLMs do pose the risk of automatic some routine tasks but the human is not going to be eliminated completely, at least not yet …

Demos

- Batch Inferencing
- Playground
- UC Functions as tools
- Agent Bricks
  - KIE
  - Q&A
  - Multi agent Supervisor
- AI Gateway

Appendix

First cut

Model & deployment type

- Open source models (as is/tune, commercial/non)
  - Pros
    - Data/model stays within your control, fine-tuning, faster inferencing
    - Quality is rapidly improving
  - Cons
    - Larger modes/datasets, in-house expertise
  - Llama 2 (FB), MPT(Mosaic ML), Dolly(Databricks)
- Proprietary model
  - LLM-as-a-service, May be fine-tuned - usually hosted elsewhere, you send the data
  - Pros
    - Faster development, better quality on routine tasks
  - Cons
    - Pay per request, Data privacy concerns, vendor lock-in
  - Eg. OpenAI, Anthropic