Retrieved-Augmented Generation (RAG) For Large Language Models

Leah Lu, Sr. Data Scientist

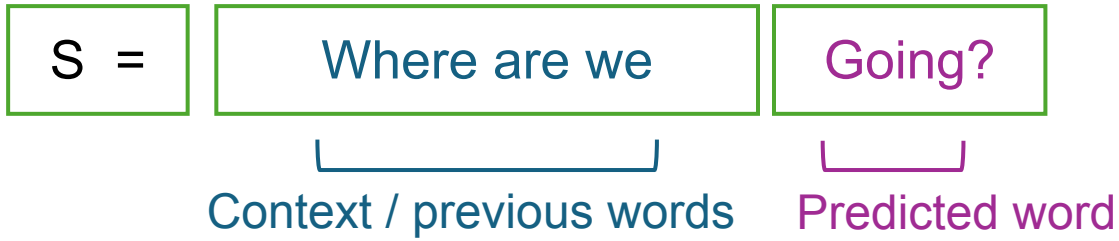
2024-03-08

Outline

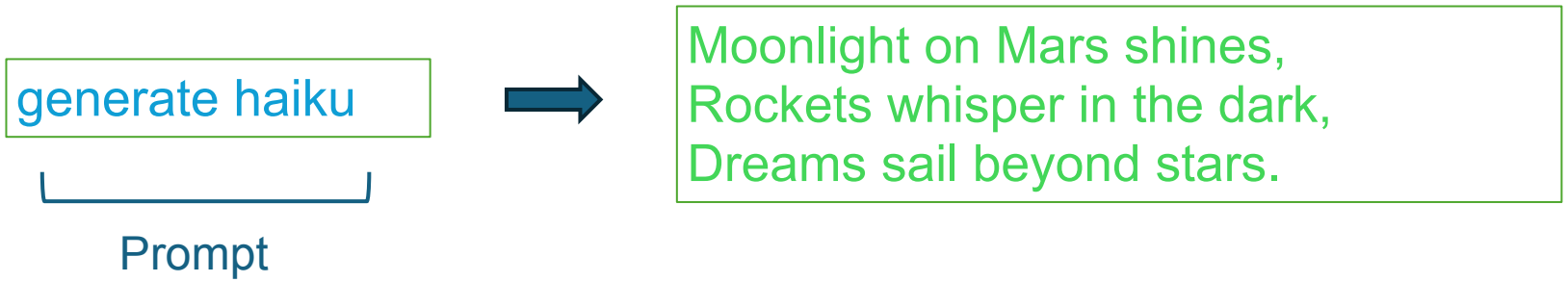
- Large Language Model
- RAG introduction and development
- RAG evaluation
- RAG ecosystem
- RAG implementation at LangChain and LLamaIndex
- Case Study: RAG (LLM + Target data) for generative recommendation
- References

The Age of Language Models

- Next Word Prediction

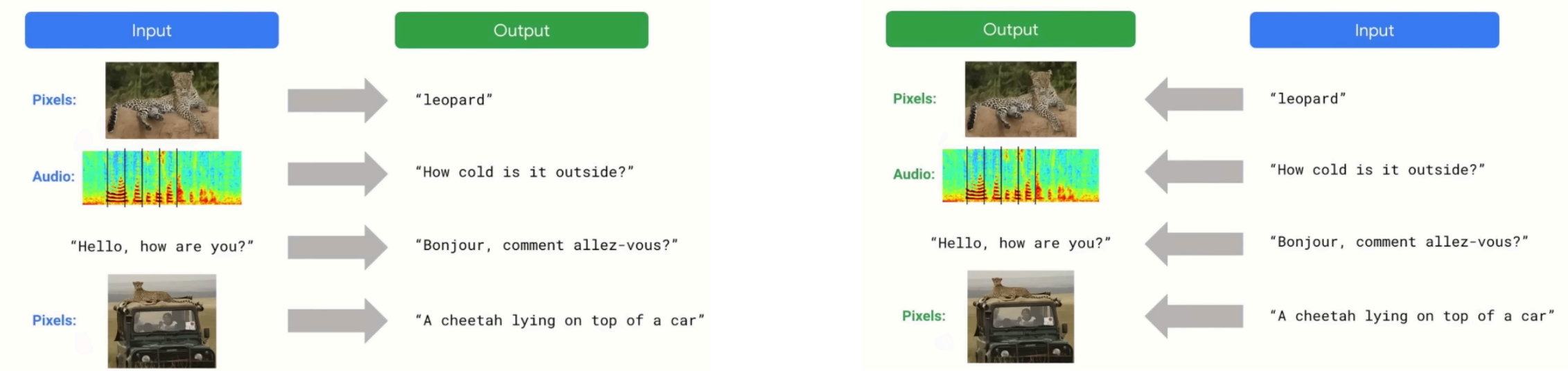


- Generative AI ← Large Language Model (tech)



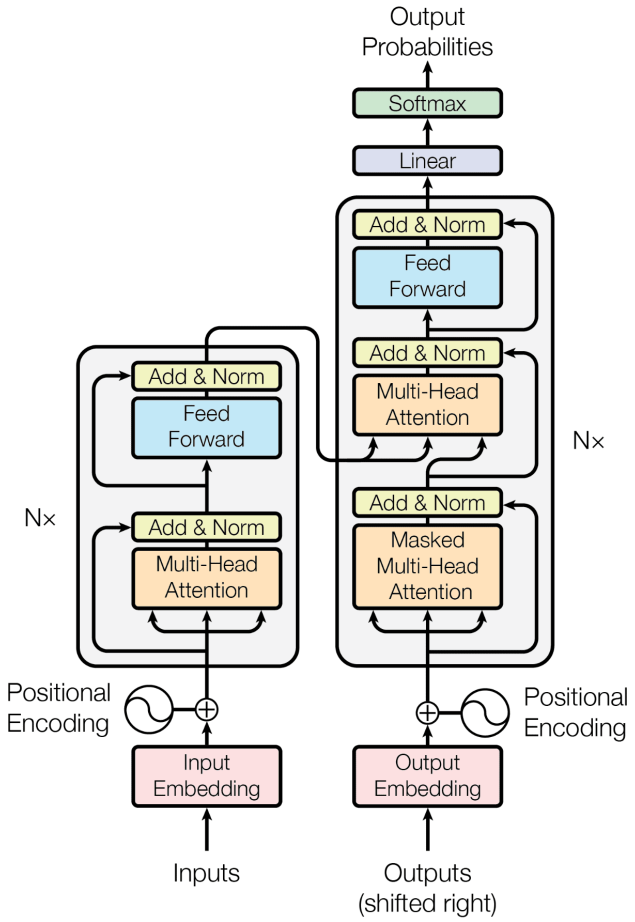
What is GenAI?

- capable of generating text, images or other data using generative models



Credit: [Jeff Dean (Google): Exciting Trends in Machine Learning]

Transformer Architecture



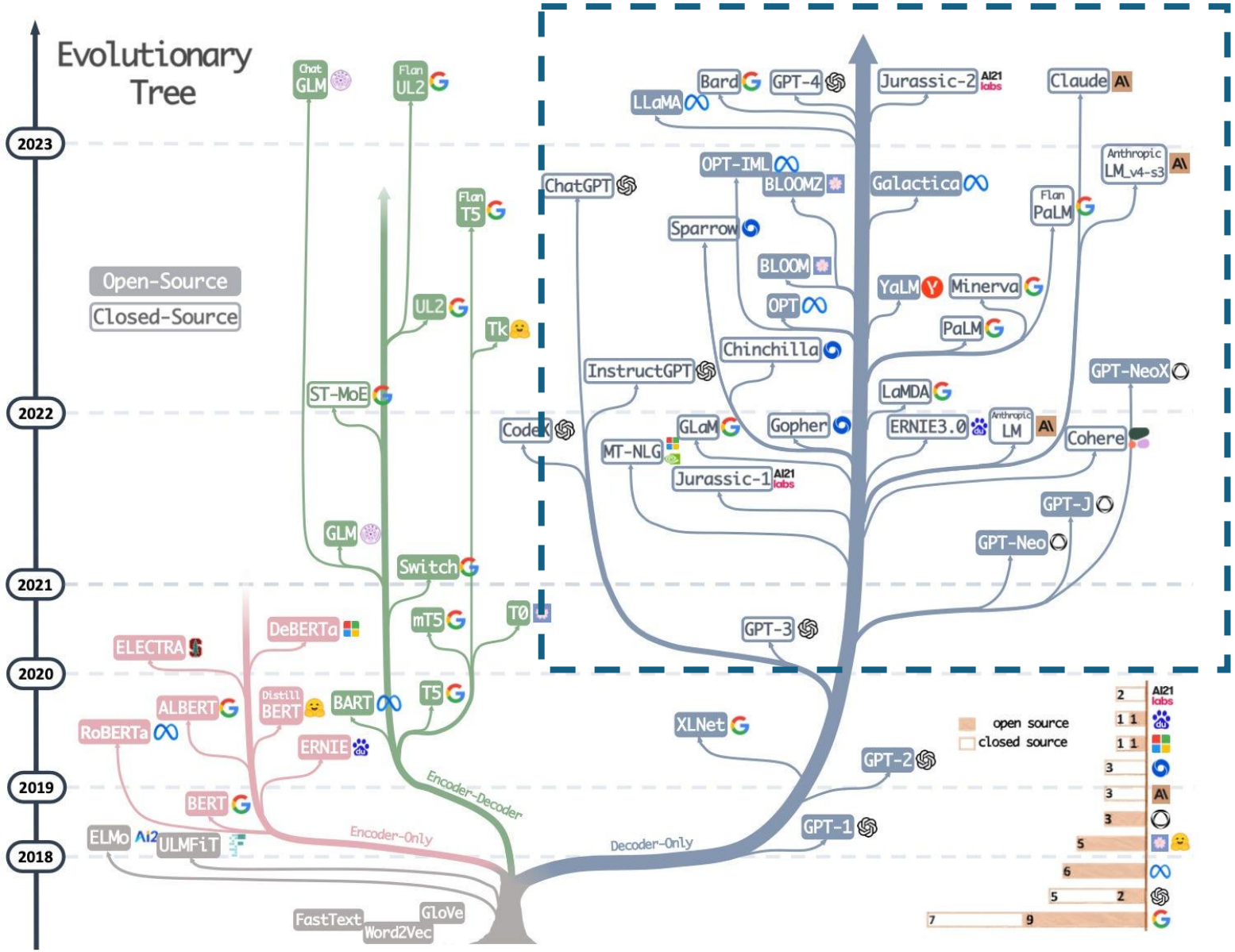
Key components:

- Self-Attention: learn the context of tokens
- Multi-Head attention: combine information across subspaces
- Positional Encoding: the position vector for tokens
- Feed Forward: fully connected, predict next token

Advanced LLM:

- Encoder-only: Bert , etc
- Decoder-only: GPT, etc
- Encoder and Decoder: T5, etc

Source: Attention Is All You Need (2017)



The evolutionary tree of modern LLMs via https://arxiv.org/abs/2304.13712.

Source: State of GPT: Andrej Karpathy – State of GPT | BRK216HFS

Current standard of intelligence

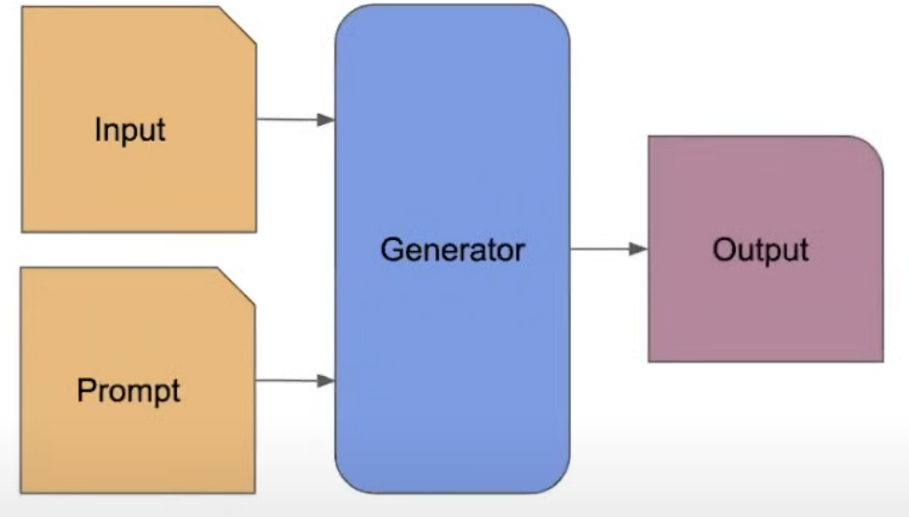
	Claude 3 Opus	Claude 3 Sonnet	Claude 3 Haiku	GPT-4	GPT-3.5	Gemini 1.0 Ultra	Gemini 1.0 Pro
Undergraduate level knowledge *MMLU*	86.8% 5 shot	79.0% 5-shot	75.2% 5-shot	86.4% 5-shot	70.0% 5-shot	83.7% 5-shot	71.8% 5-shot
Graduate level reasoning *GPQA, Diamond*	50.4% 0-shot CoT	40.4% 0-shot CoT	33.3% 0-shot CoT	35.7% 0-shot CoT	28.1% 0-shot CoT	–	–
Grade school math *GSM8K*	95.0% 0-shot CoT	92.3% 0-shot CoT	88.9% 0-shot CoT	92.0% 5-shot CoT	57.1% 5-shot	94.4% Maj1@32	86.5% Maj1@32
Math problem-solving *MATH*	60.1% 0-shot CoT	43.1% 0-shot CoT	38.9% 0-shot CoT	52.9% 4-shot	34.1% 4-shot	53.2% 4-shot	32.6% 4-shot
Multilingual math *MGSM*	90.7% 0-shot	83.5% 0-shot	75.1% 0-shot	74.5% 8-shot	–	79.0% 8-shot	63.5% 8-shot
Code *HumanEval*	84.9% 0-shot	73.0% 0-shot	75.9% 0-shot	67.0% 0-shot	48.1% 0-shot	74.4% 0-shot	67.7% 0-shot
Reasoning over text *DROP, F1 score*	83.1 3-shot	78.9 3-shot	78.4 3-shot	80.9 3-shot	64.1 3-shot	82.4 Variable shots	74.1 Variable shots
Mixed evaluations *BIG-Bench-Hard*	86.8% 3-shot CoT	82.9% 3-shot CoT	73.7% 3-shot CoT	83.1% 3-shot CoT	66.6% 3-shot CoT	83.6% 3-shot CoT	75.0% 3-shot CoT
Knowledge Q&A *ARC-Challenge*	96.4% 25-shot	93.2% 25-shot	89.2% 25-shot	96.3% 25-shot	85.2% 25-shot	–	–
Common Knowledge *HellaSwag*	95.4% 10-shot	89.0% 10-shot	85.9% 10-shot	95.3% 10-shot	85.5% 10-shot	87.8% 10-shot	84.7% 10-shot

- On Mar.4, 2024, Claude 3 model from Anthropic showed increased capabilities in analysis, forecasting and generation.

source: https://www.anthropic.com/news/claude-3-family

Eliciting Outputs from LLM

- Problems (but hopefully getting better)
  - Hallucination
  - Attribution
  - Staleness
  - Bias
  - Reliability



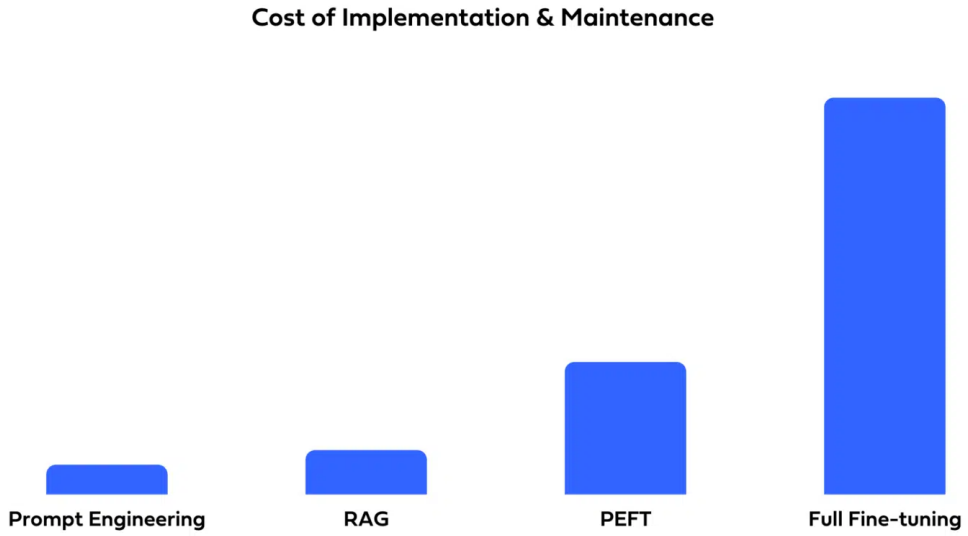
Q: "What is the capital of the ocean?"

A: "The capital of the ocean is Atlantis."

Hallucination: plausible but false or not factual

Tweak LLM for needs

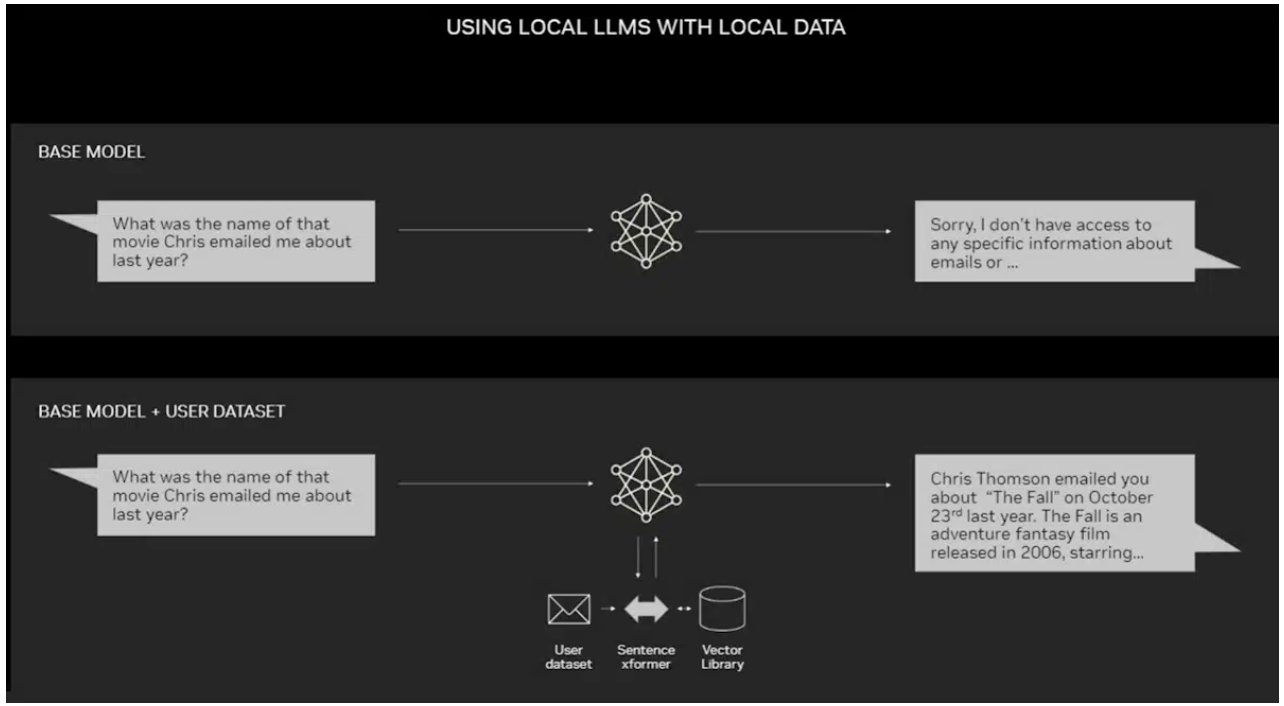
- Prominent approaches: Full Fine-tuning, Parameter-efficient Fine-tuning (PEFT), Prompt Engineering, RAG (Retrieval Augmented Generation)
- They vary in expertise needed, cost, and suitability for different scenarios.



Source: https://deci.ai/blog/fine-tuning-peft-prompt-engineering-and-rag-which-one-is-right-for-you/

Retrieval-Augmented Generation (RAG)

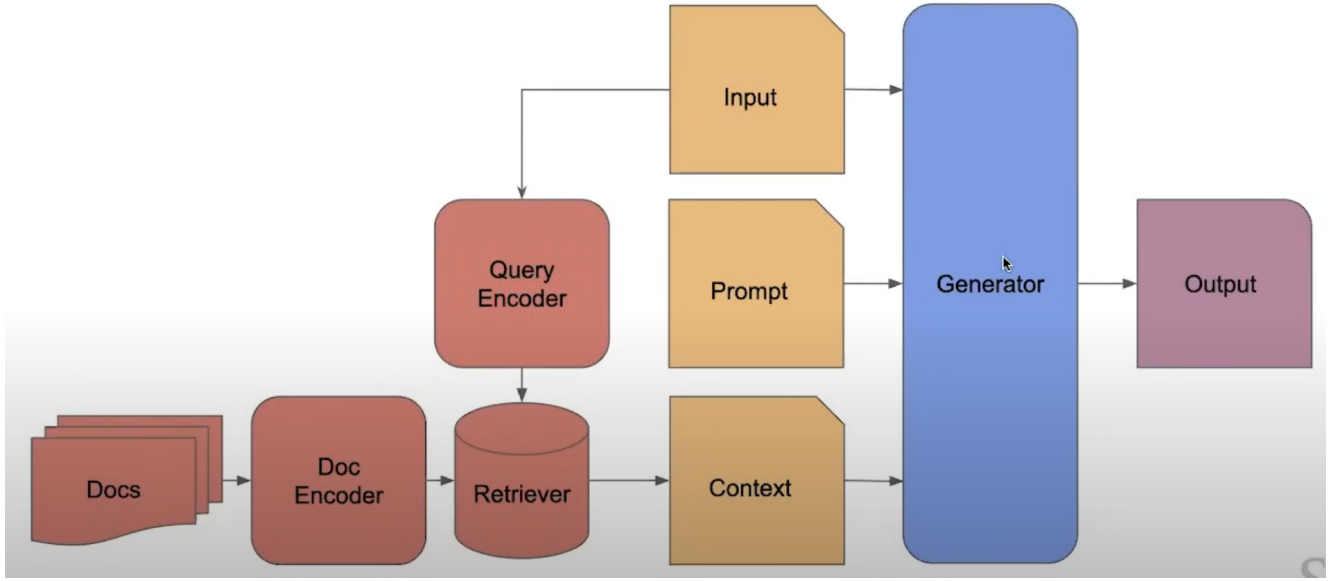
- a technique for enhancing the accuracy and reliability of generative AI models with facts fetched from external sources



source: https://blogs.nvidia.com/blog/what-is-retrieval-augmented-generation/

Advantages:

- Have customization and avoid staleness
- Less hallucination and attribute to the source
- Precision and Relevance
- Scalability and Versatility



RAG example

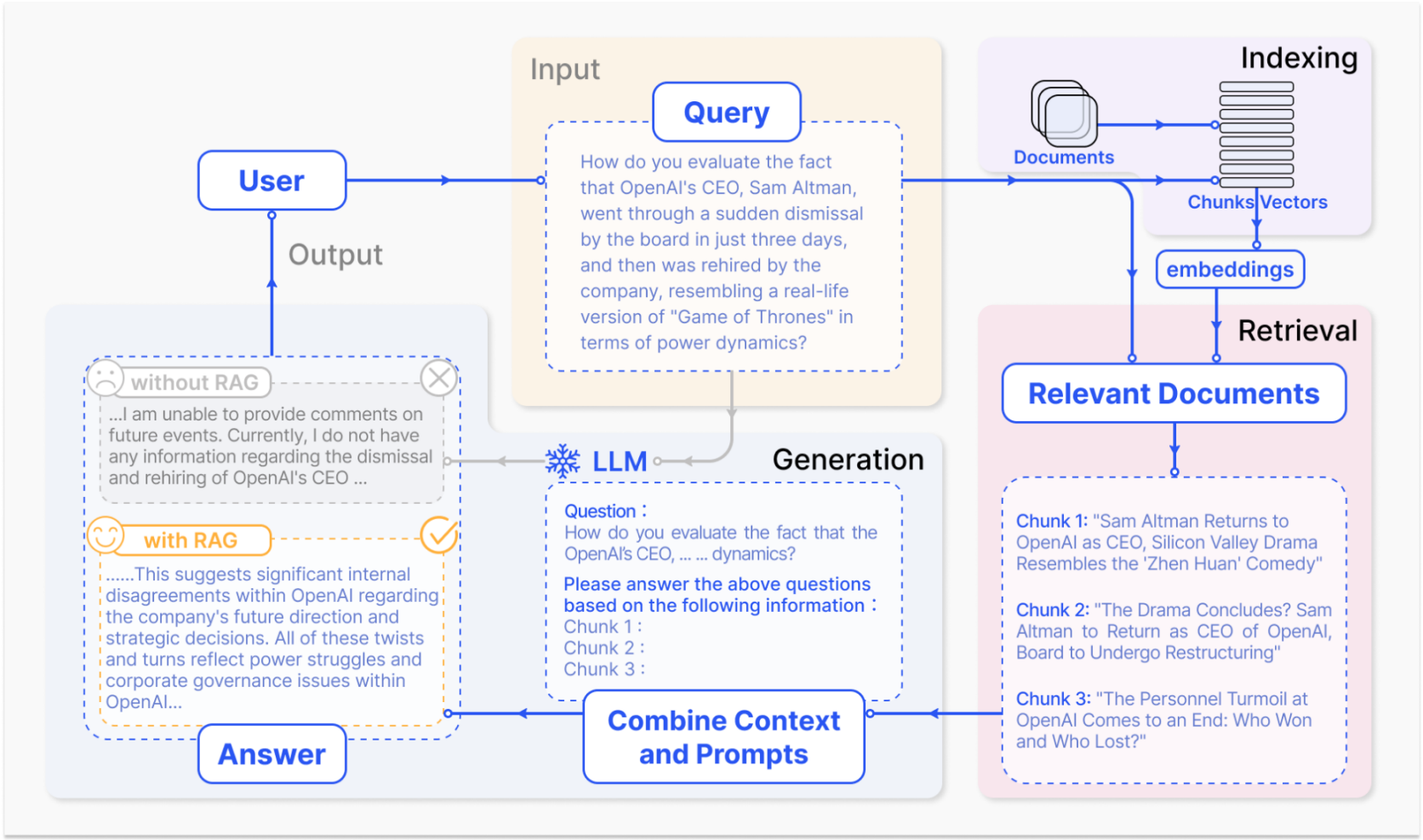


Figure 2: A representative instance of the RAG process applied to question answering

RAG development

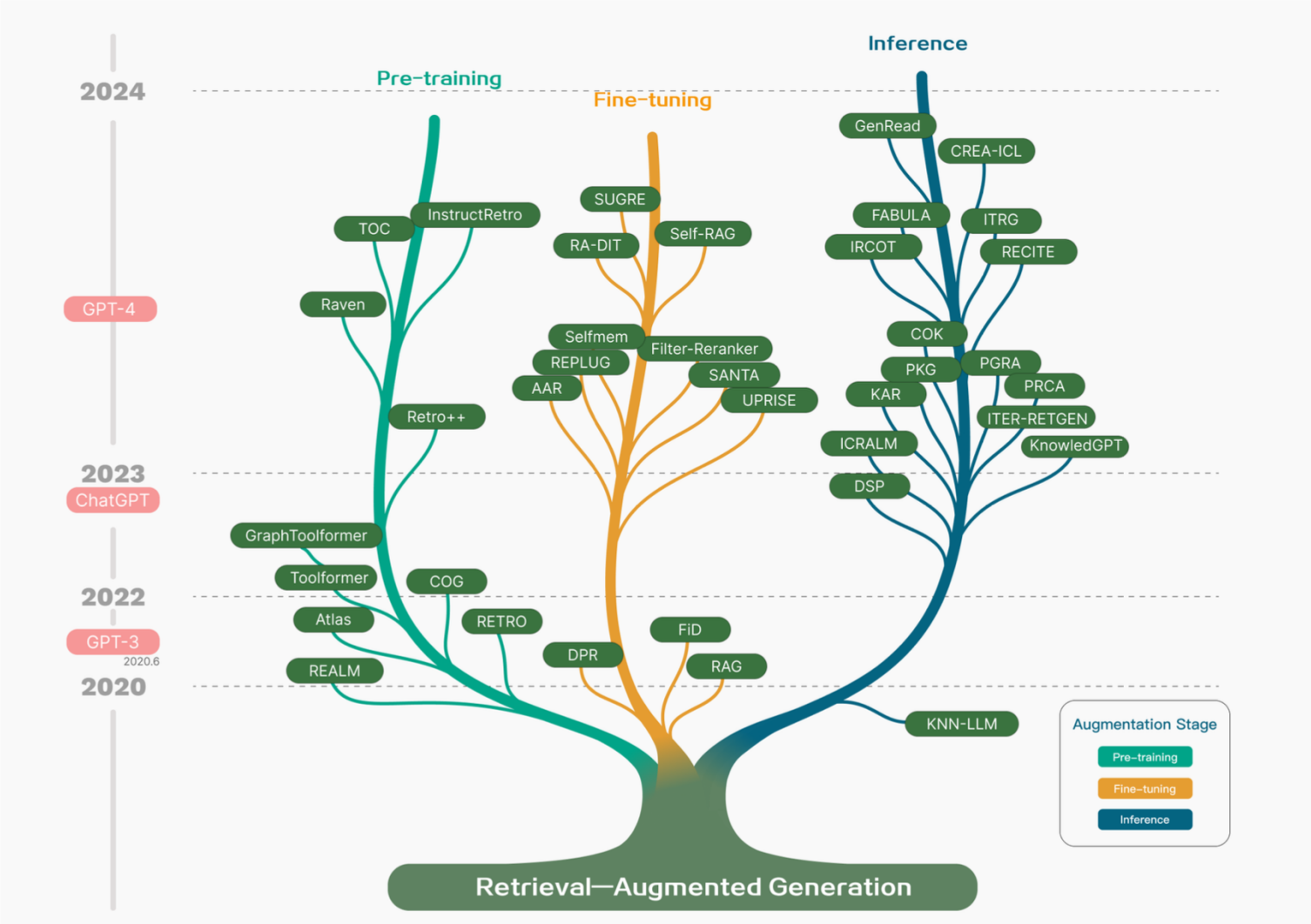


Figure 1: Technology tree of RAG research development featuring representative works

AI model milestones eras

- 2020 → 2022 → 2023 → 2024

Model development process

- Pre-training
- Fine-tuning
- Inference

Model development process

Stage	Description	Model Examples
Pre-training	Bolsters PTMs for open-domain QA with retrieval-based strategies, focusing on knowledge-intensive tasks and domain-specific model development.	REALM, RETRO, Atlas, COG
Fine-tuning	Tailors retriever and generator to specific scenarios, aligning them with the preferences of LLMs for improved adaptability in multi-task scenarios.	PROMPTAGATOR, LLM-Embedder
Inference	Integrates RAG with LLMs, using advanced techniques to introduce contextually rich information for complex task execution.	DSP framework, PKG, CREA-ICL, RECITE, ITRG

Three paradigms of RAG

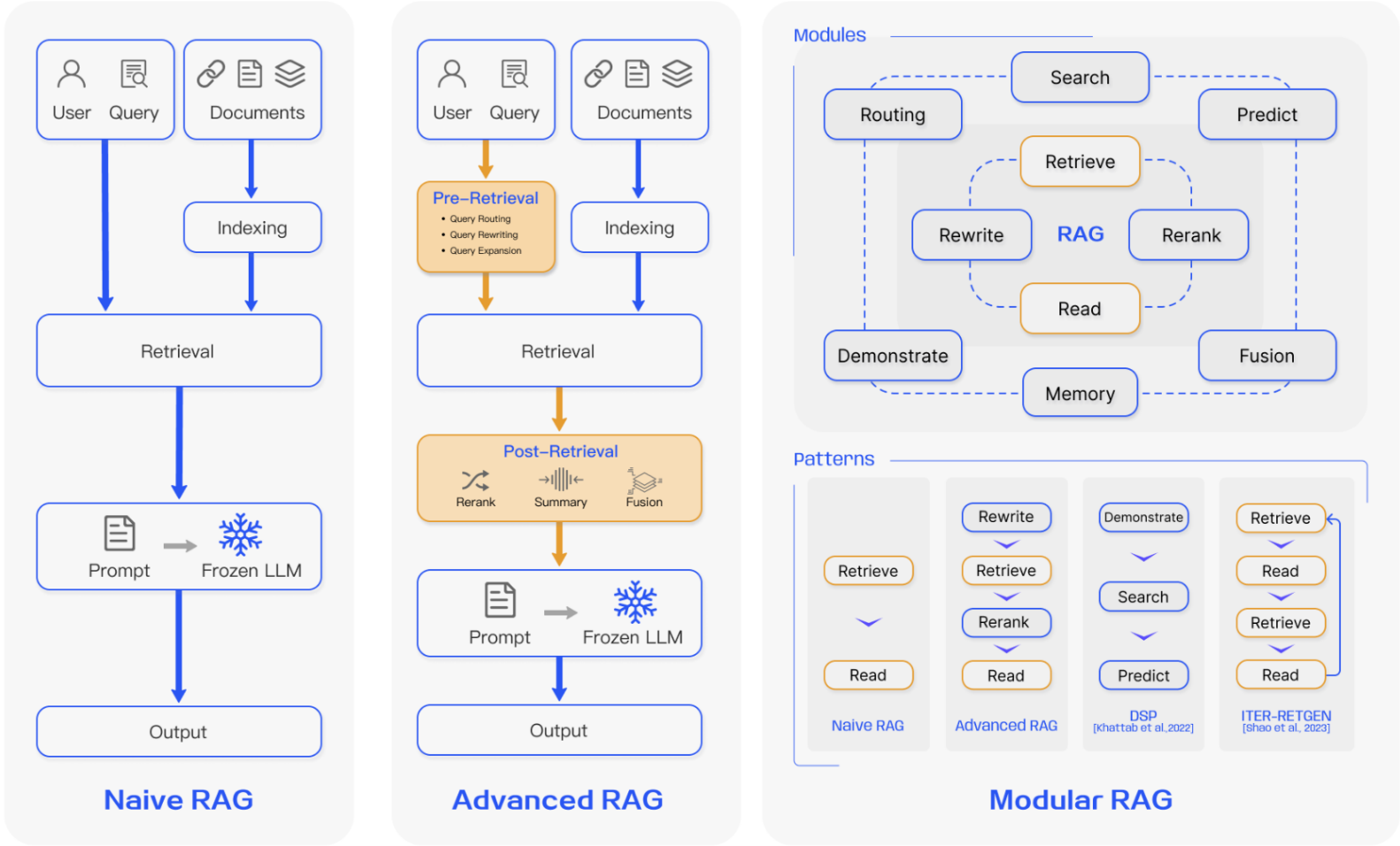


Figure 3: Comparison between the three paradigms of RAG

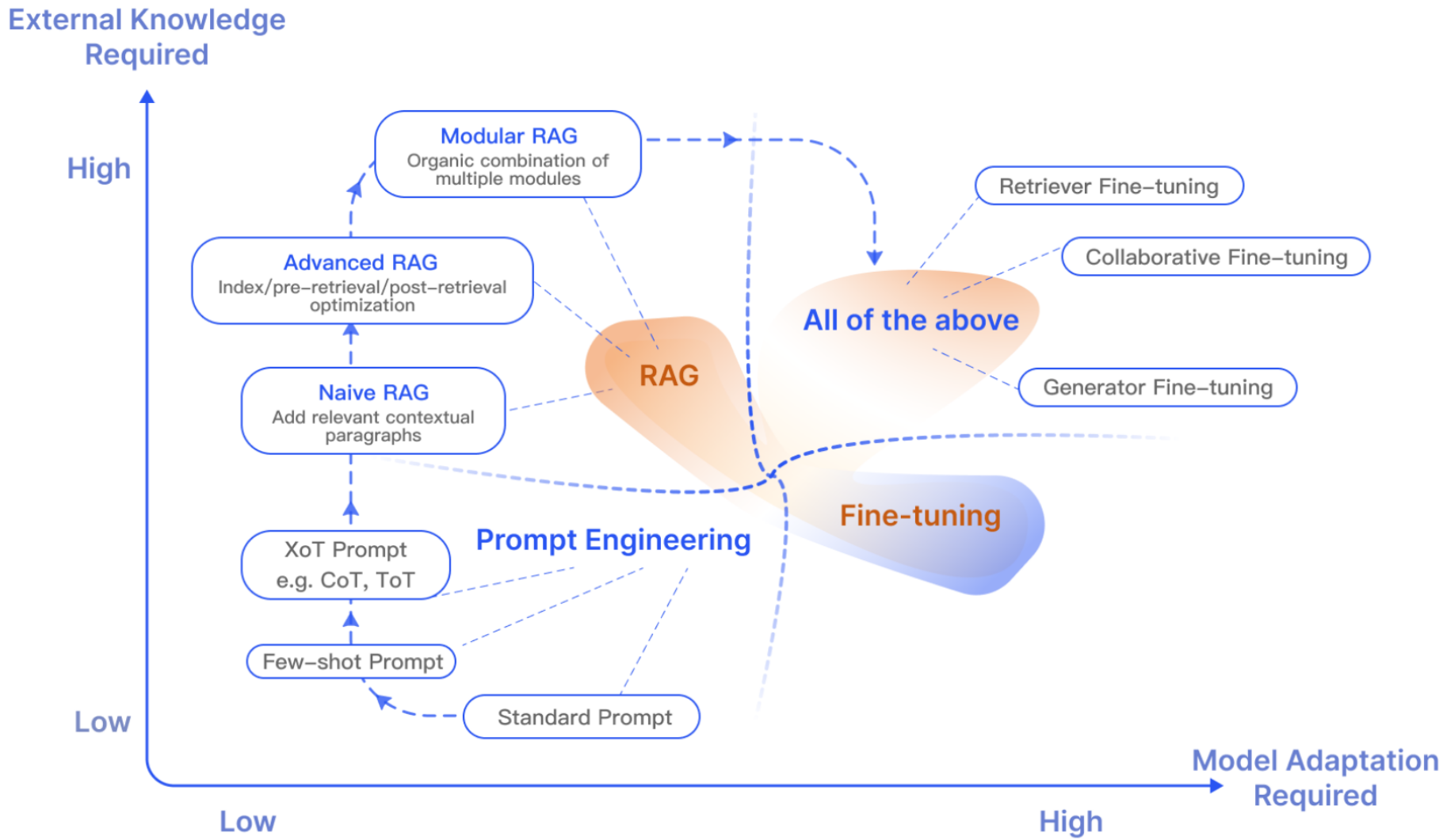


Figure 6: RAG compared with other model optimization methods

RAG evaluation

Table 2: Summary of metrics applicable for evaluation aspects of RAG

	Context Relevance	Faithfulness	Answer Relevance	Noise Robustness	Negative Rejection	Information Integration	Counterfactual Robustness
Accuracy	✓	✓	✓	✓	✓	✓	✓
EM					✓		
Recall	✓						
Precision	✓			✓			
R-Rate							✓
Cosine Similarity			✓				
Hit Rate	✓						
MRR	✓						
NDCG	✓						

Evaluation Framework	**Evaluation Targets**	**Evaluation Aspects**	**Quantitative Metrics**
RGB†	Retrieval Quality	Noise Robustness	Accuracy
		Negative Rejection	EM
	Generation Quality	Information Integration	Accuracy
		Counterfactual Robustness	Accuracy
RECALL†	Generation Quality	Counterfactual Robustness	R-Rate (Reappearance Rate)
RAGAS‡	Retrieval Quality	Context Relevance	*
		Faithfulness	*
	Generation Quality	Answer Relevance	Cosine Similarity
ARES‡	Retrieval Quality	Context Relevance	Accuracy
		Faithfulness	Accuracy
	Generation Quality	Answer Relevance	Accuracy
TruLens‡	Retrieval Quality	Context Relevance	*
		Faithfulness	*
	Generation Quality	Answer Relevance	*

*† represents a benchmark, and ‡ represents a tool. * denotes customized quantitative metrics, which deviate from traditional metrics. Readers are encouraged to consult pertinent literature for the specific quantification formulas associated with these metrics, as required.*

RAGAS evaluation

- Retriever: Offers **context_relevancy** that measures the performance of your retrieval system
- Generator (LLM): Provides **faithfulness** that measures hallucinations and **answer_relevancy** that measures how relevant the answers are to the question

	question	ground_truths	answer	contexts	context_relavency	faithfulness	answer_relevancy
0	How to deposit a cheque issued to an associate...	[Have the check reissued to the proper payee.J...	\nThe best way to deposit a cheque issued to a...	[Just have the associate sign the back and the...	0.867	1.0	0.922
1	Can I send a money order from USPS as a business?	[Sure you can. You can fill in whatever you w...	\nYes, you can send a money order from USPS as...	[Sure you can. You can fill in whatever you w...	0.855	1.0	0.923
2	1 EIN doing business under multiple business n...	[You're confusing a lot of things here. Compan...	\nYes, it is possible to have one EIN doing bu...	[You're confusing a lot of things here. Compan...	0.768	1.0	0.824

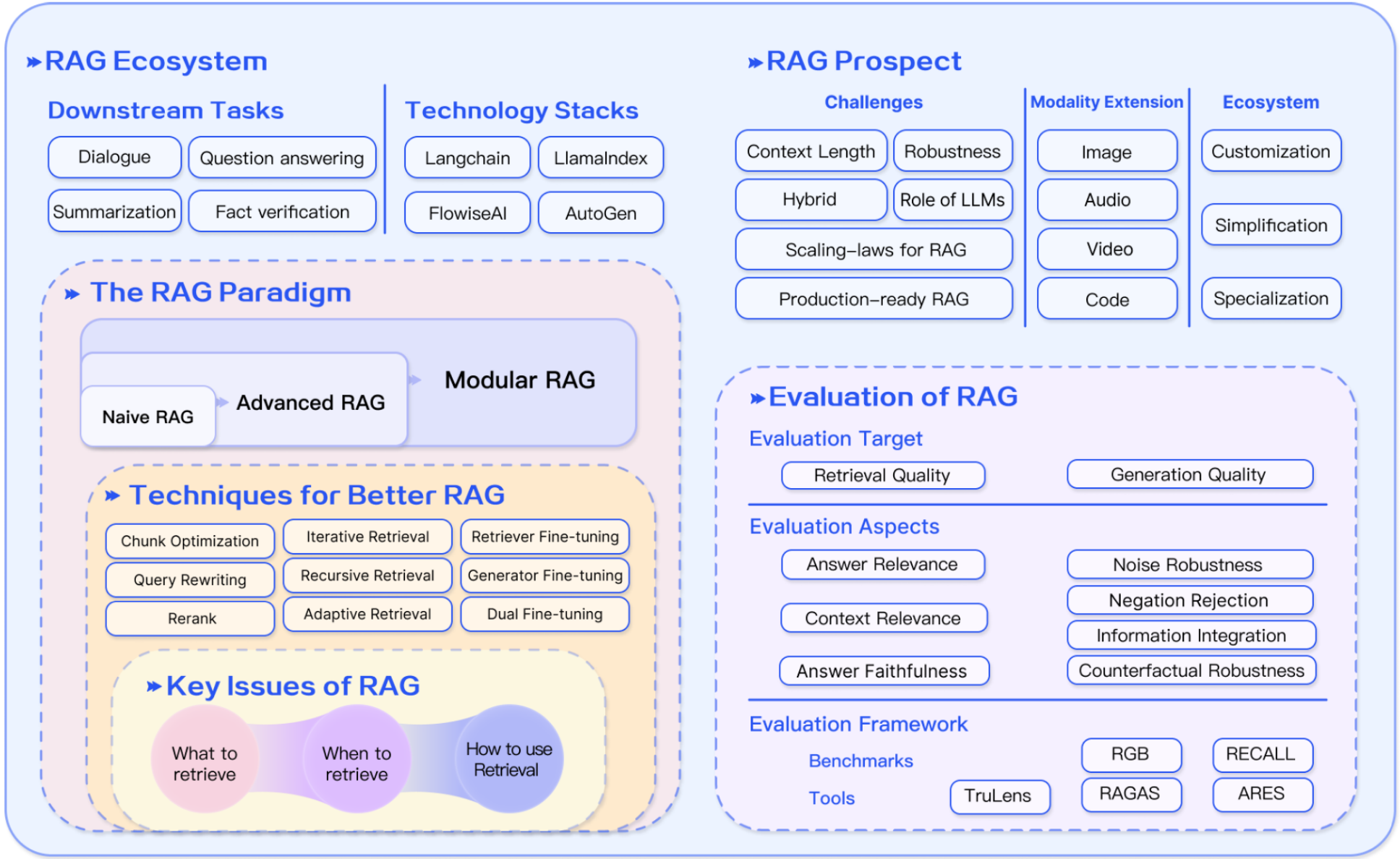
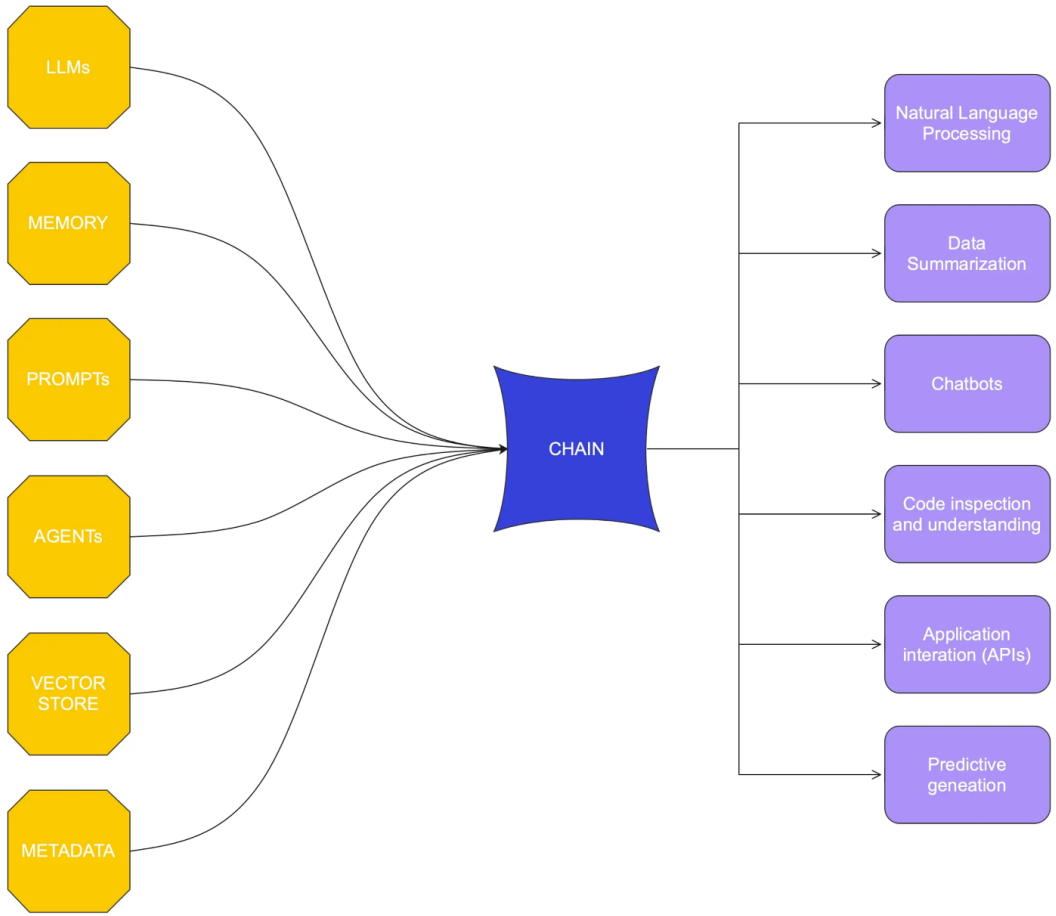


Figure 7: Summary of RAG ecosystem

RAG challenges and open questions

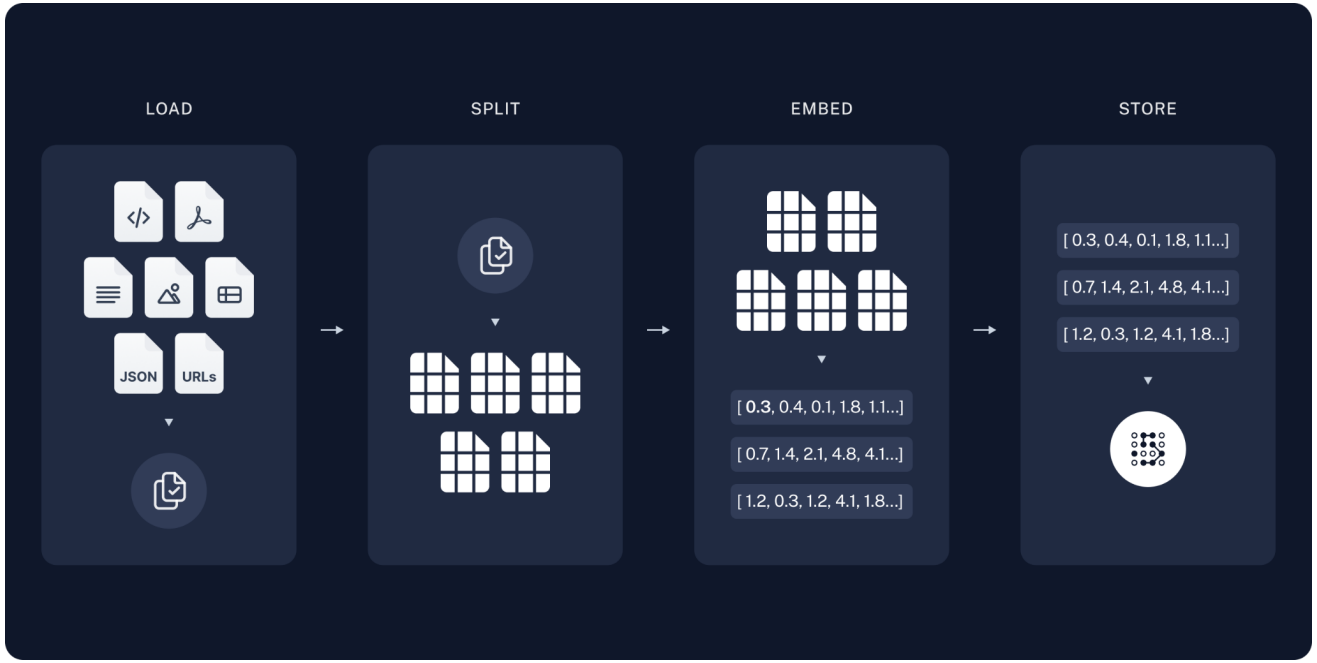
- Retrieval Quality
- Integration of Retrieved Data
- Data Currency and Relevance
- Latency
- Interpretability and Traceability
- Mitigating Hallucinations
- Bias and Fairness
- Multimodal Retrieval and Generation
- Scalability, Accuracy and Relevance
- Domain-specific Challenges
- Human-in-the-loop Integration

LangChain -- A Framework for LLM Applications

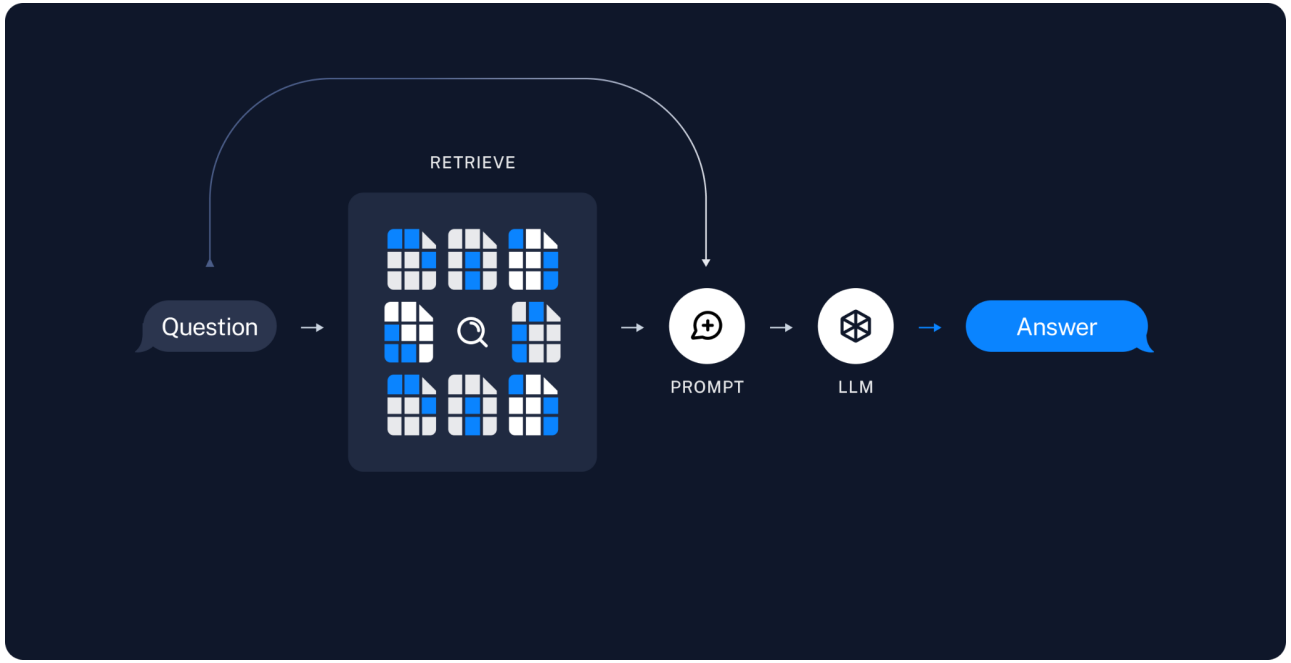


'Wheel' for multi-purpose LLM applications
Key components:

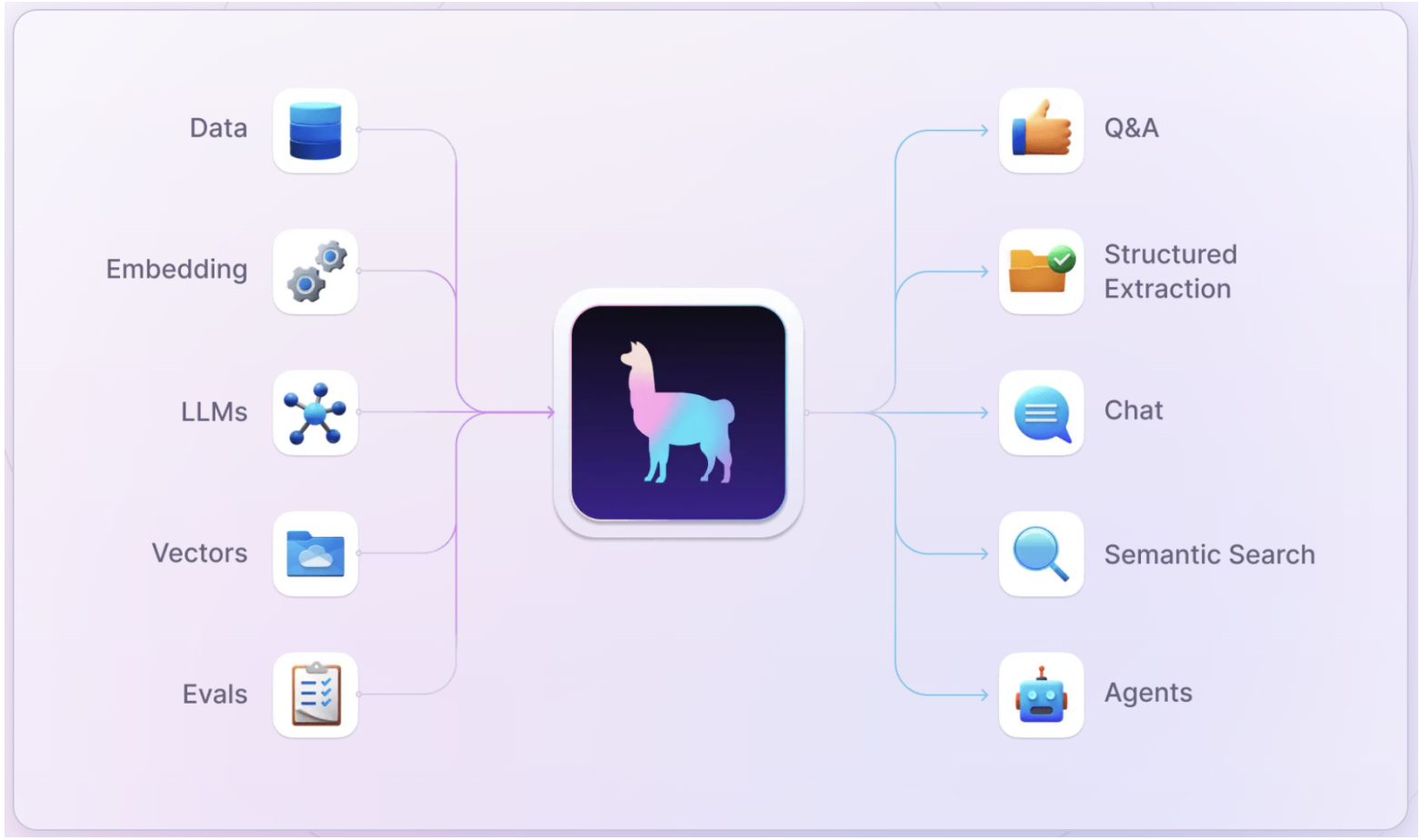
- Chains
- Memory
- Agents
- Prompts



RAG for QA example



LlamaIndex -- framework for building LLM applications



Key Applications

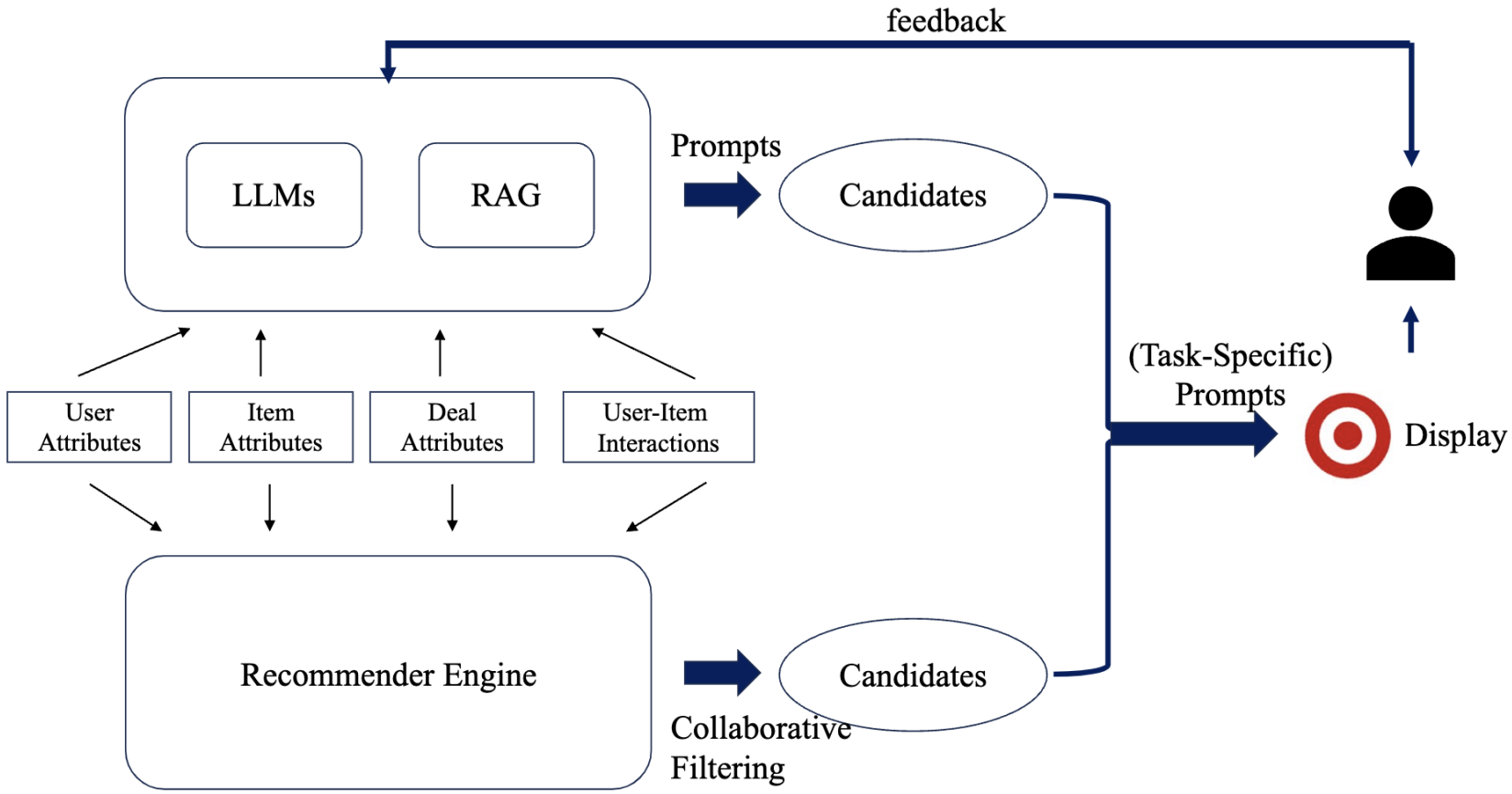
LangChain:	LlamaIndex:
• *Text generation* • *Translation* • *Question answering* • *Summarization* • *Classification*	• *Document search and retrieval* • *LLM augmentation* • *Chatbots and virtual assistants* • *Data analytics* • *Content generation*

Source: https://medium.com/@zahidali133/comparison-of-llamaindex-and-langchain-4900989752ac

Take home messages

- LLM advancements
- How RAG works
- Ideas to make improvements at RAG
- There is RAG eco-system
- Reach out to LangChain for RAG implementation
- Reach out to LlamaIndex for RAG implementation

Case Study: RAG (LLM + Target data) for generative recommendation



Innovation week slides, credit shared with Dong Li

References

- Gao, Yunfan, et al. "Retrieval-augmented generation for large language models: A survey." arXiv preprint arXiv:2312.10997 (2023).
- Retrieval Augmented Language Models Speaker: Douwe Kiela, Contextual AI from CS25: Transformers United V3 at Stanford, Lecture Video
- LangChain RAG Cookbook
- LLamaIndex RAG