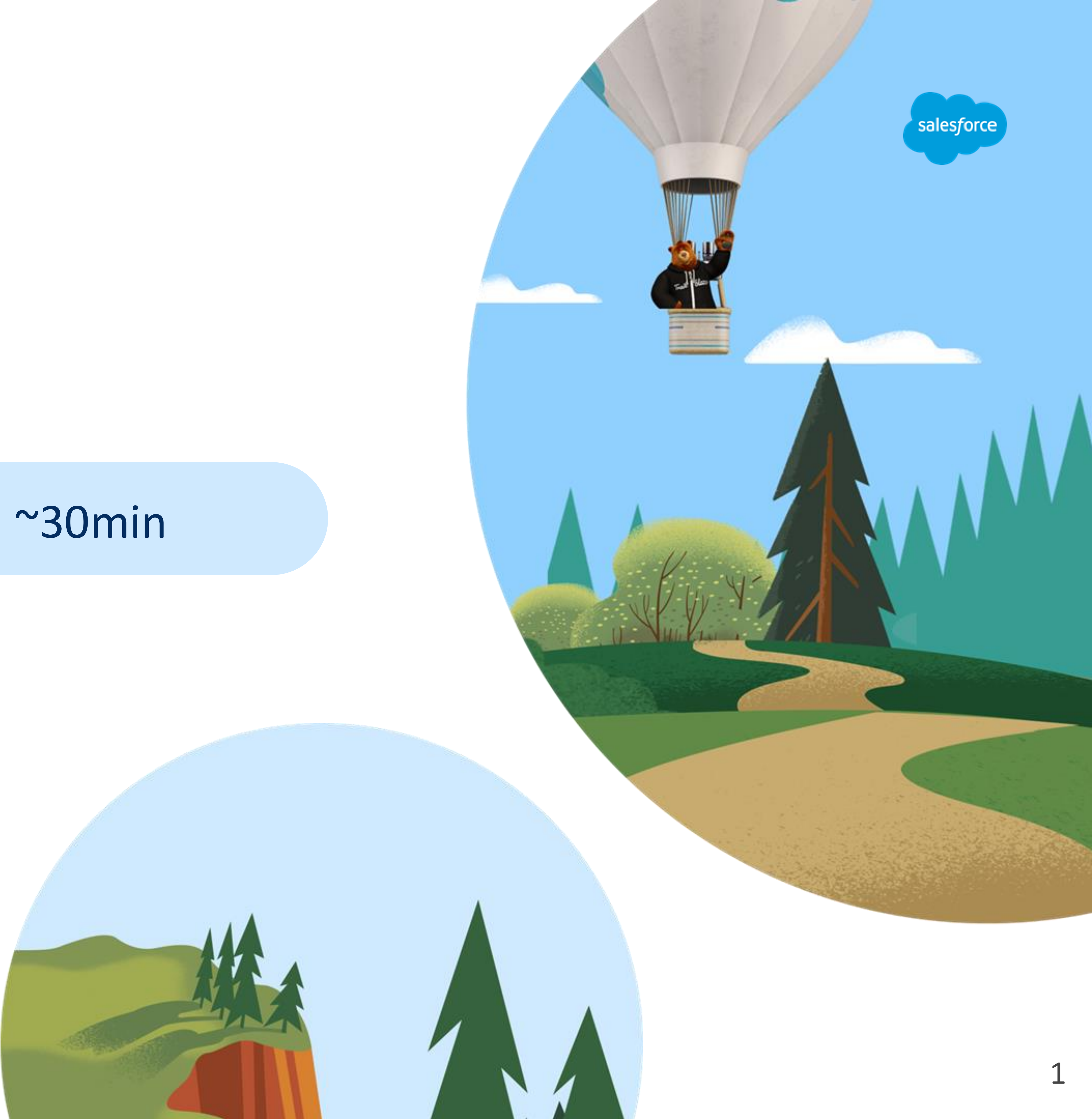
Agenda

Evaluation and Benchmark

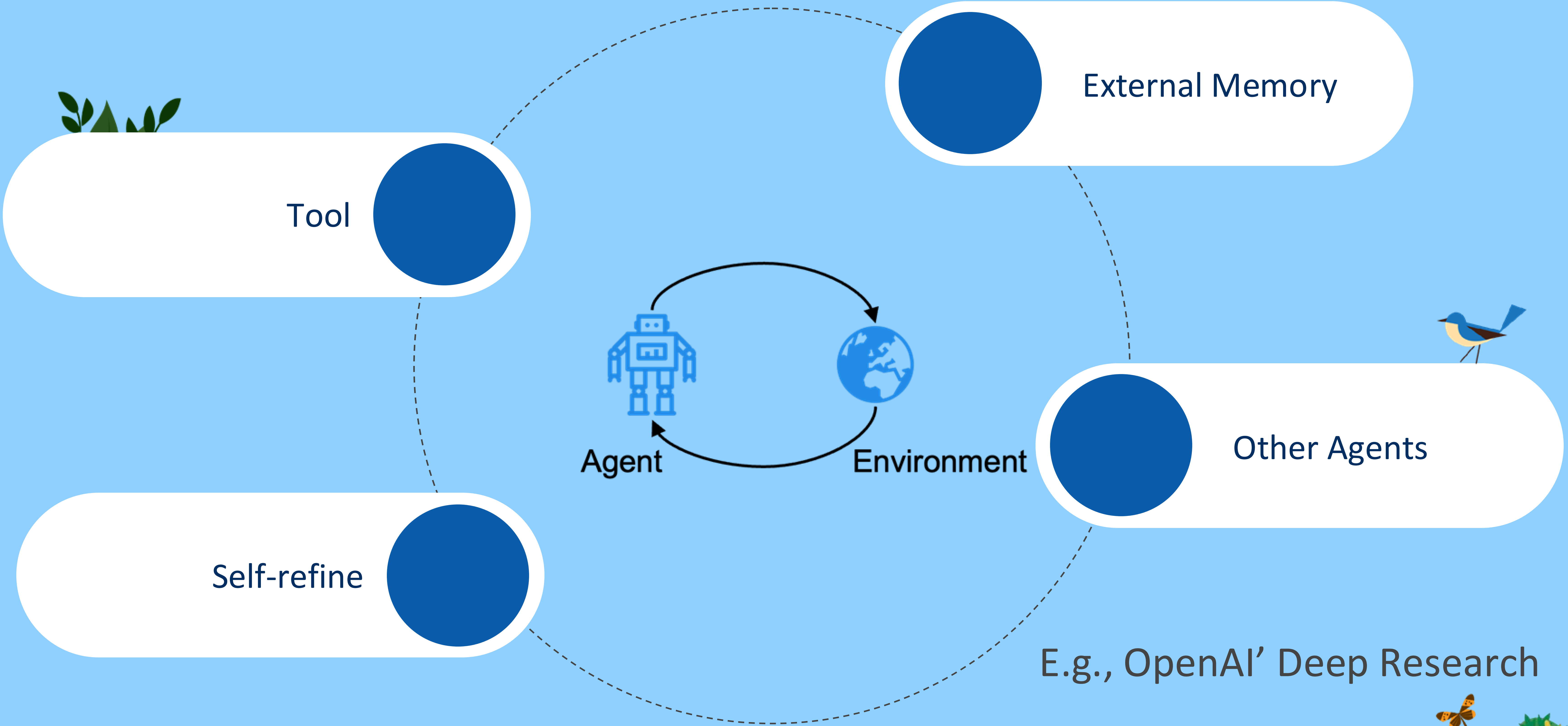
Parametric Knowledge Adaptation

Semi-Parametric Knowledge Adaptation ~30min

Summary, Discussion, QAs



Semi-Parametric Knowledge



External Memory

Tool

Agent

Environment

Other Agents

Self-refine

E.g., OpenAI’ Deep Research

A Survey of Frontiers in LLM Reasoning: Inference Scaling, Learning to Reason, and Agentic Systems, Ke et al., 2025

RAG – Role

Bridge Gap

Off-the-shelf LLMs may not have been optimized for leveraging external information in its context

Additional adaptation is required for better performance

Autonomous Decision Making

A RAG system needs to decide whether it needs external information or it can respond directly

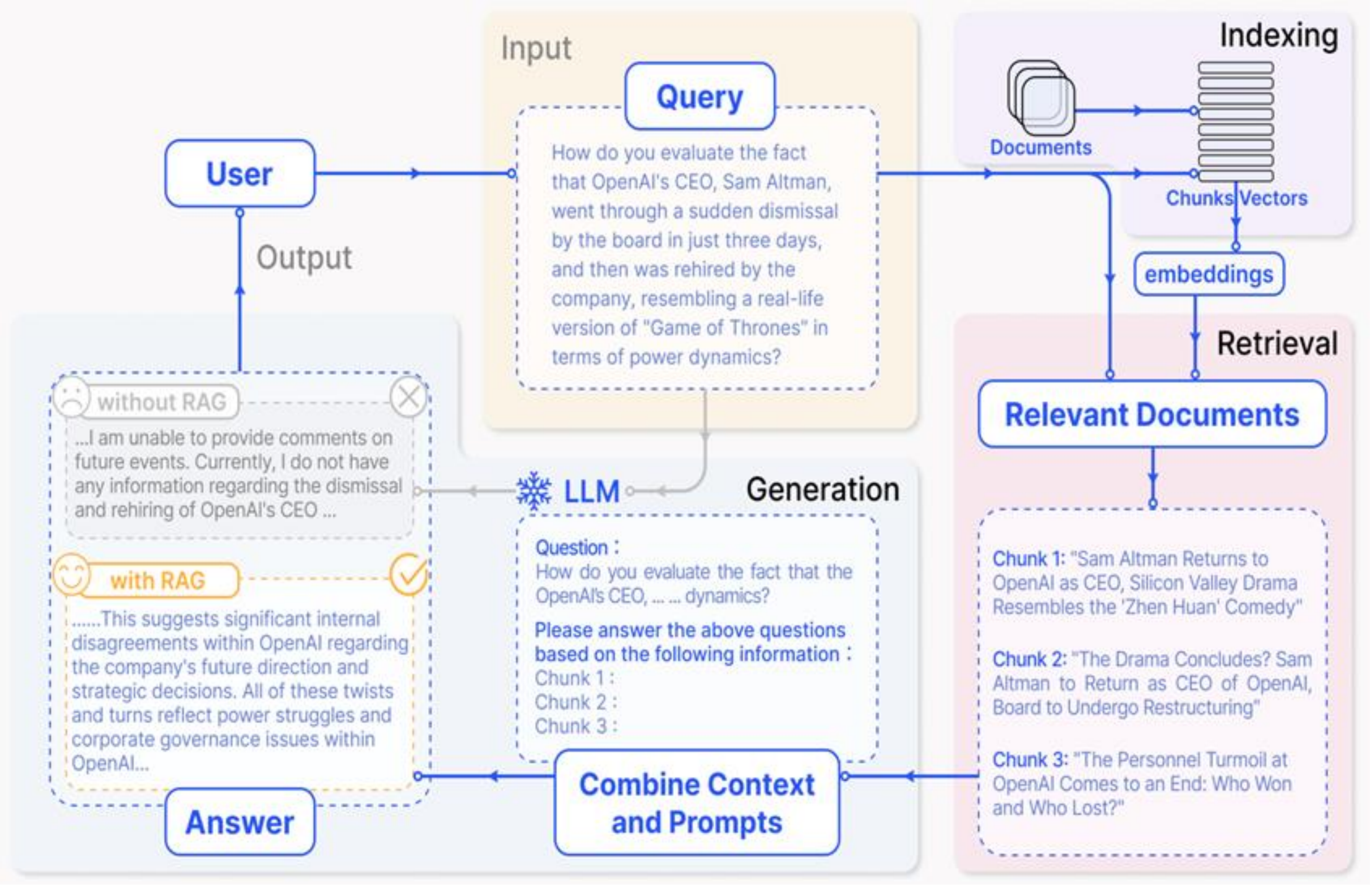
It may need to ask for clarification to the user, do multiple searches via retrieval and aggregate results across documents

RAG - Key Ideas

Example Workflow

Three Main Components

- LLM
- Retriever
- LLM-Retriever Interaction



How do you evaluate the fact that OpenAI's CEO, Sam Altman, went through a sudden dismissal by the board in just three days, and then was rehired by the company, resembling a real-life version of "Game of Thrones" in terms of power dynamics?

...I am unable to provide comments on future events. Currently, I do not have any information regarding the dismissal and rehiring of OpenAI's CEO ...

......This suggests significant internal disagreements within OpenAI regarding the company's future direction and strategic decisions. All of these twists and turns reflect power struggles and corporate governance issues within OpenAI...

Question :
How do you evaluate the fact that the OpenAI's CEO, ... dynamics?
Please answer the above questions based on the following information :
Chunk 1 :
Chunk 2 :
Chunk 3 :

Chunk 1: "Sam Altman Returns to OpenAI as CEO, Silicon Valley Drama Resembles the 'Zhen Huan' Comedy"

Chunk 2: "The Drama Concludes? Sam Altman to Return as CEO of OpenAI, Board to Undergo Restructuring"

Chunk 3: "The Personnel Turmoil at OpenAI Comes to an End: Who Won and Who Lost?"

Minimalist RAG System

RAG – Key Considerations

Training Recipe

Data Recipe:

- Hard to obtain ground truth decision-making trajectory data.
- Model should be robust to potentially noisy context.

Model Recipe:

Algorithm: How to optimize the LLM for search-based interactions?

Training Workflow: What kind of workflow we should use?

Seed Data

Data Source: Where to get the data?

Data Mixture: What should be included in the RAG data?

Data Budget: How much data we need?

RAG – Key Ideas

LLM and Decision Making

Post-train LLMs for contextual usage

Deal with:

- Noisy context (passages from same document and different documents)
- Conflicting evidence
- Counterfactual evidence
- Absence of knowledge

E.g., SFR-RAG (Salesforce), RAG 2.0 (Contextual AI)

LLMs with agentic workflow

- Predefined or autonomous workflow.
- Single agent vs. multi-agent system
- Planner and worker agents

E.g., Infogent, Manus Agent, Deep Research (OpenAI)

INFOGENT: An Agent-Based Framework for Web Information Aggregation, Reddy, et al., 2024

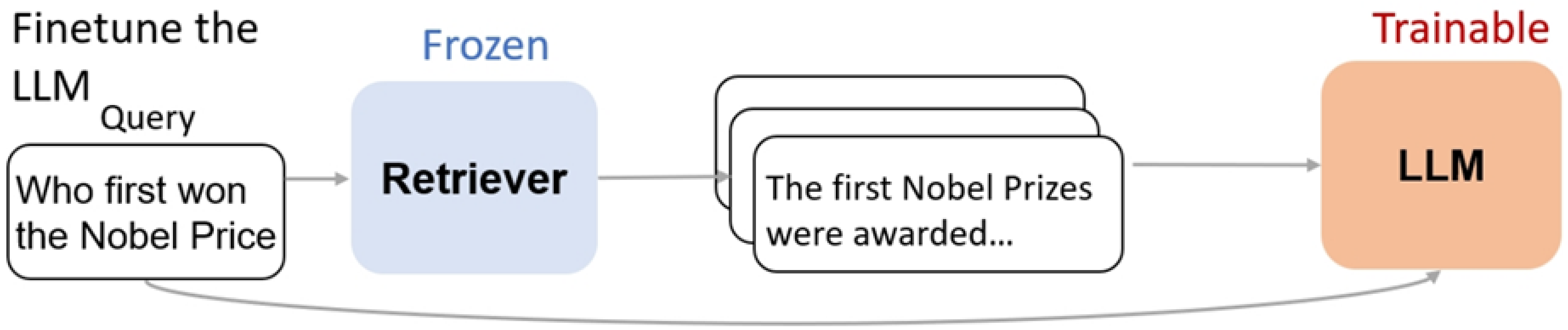
RAG – Key Ideas

Train LLMs for Contextual Use

Post-train LLMs for RAG scenarios:

Create contextual fine-tuning data to deal with noisy contexts, counterfactual contexts, no-answer contexts and conflicting

Examples: SFR-RAG, RAG 2.0


Finetune the LLM
Query
Who first won the Nobel Price
Frozen
Retriever
The first Nobel Prizes were awarded...
Trainable
LLM

1. **Fix the retriever**
2. **Train the LLM for contextual usage**

SFR-RAG: Towards Contextually Faithful LLMs, Nguyen et al., 2024
RAG2.0: https://contextual.ai/introducing-rag2/

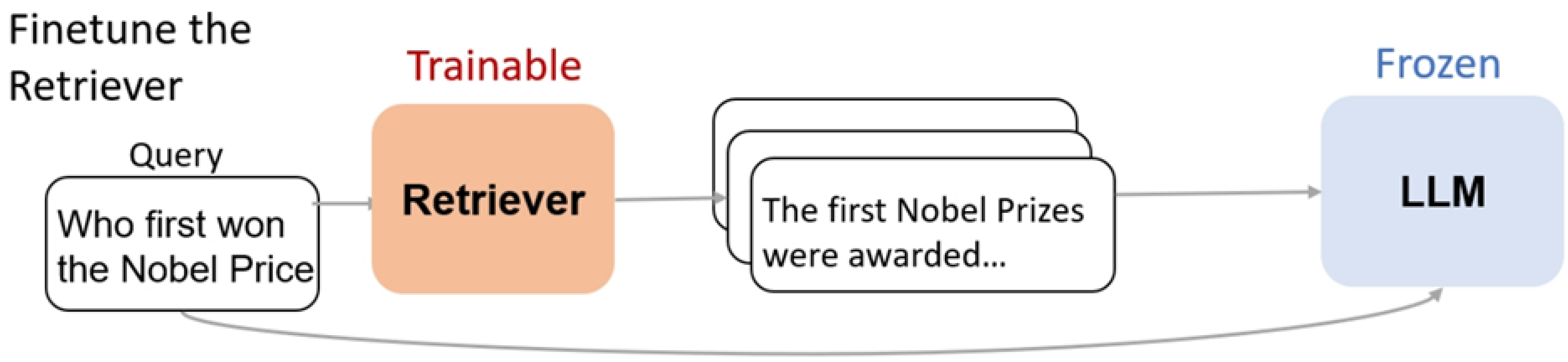
RAG – Key Ideas

Align Retriever to LLM

The output of a frozen LLM is used as supervision signals to train the retriver

Examples: REPLUG, Atlas

1. **Fix the LLM**
2. **Align the retriever to LLM**



Finetune the Retriever
Trainable
Frozen
Query
Who first won the Nobel Price
Retriever
The first Nobel Prizes were awarded...
LLM

REPLUG: Retrieval-Augmented Black-Box Language Models, Shi et al., 2023
Atlas: Few-shot Learning with Retrieval Augmented Language Models, Izacard, 2022

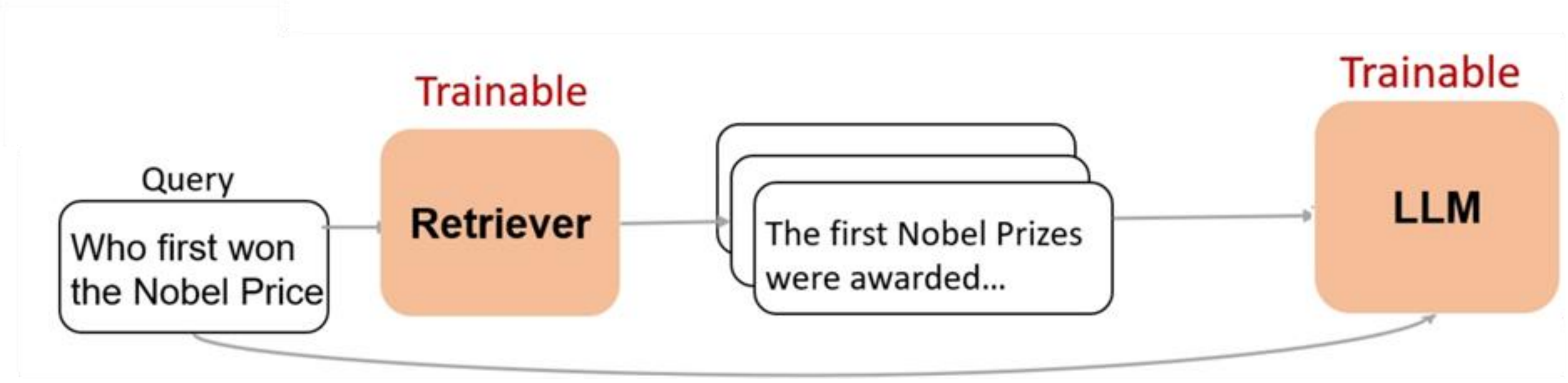
RAG – Key Ideas

Train both the LLM and Retriver

1. Train both the LLM and the retriever

Jointly or sequentially train the retriever and LLMs so that they are aligned

Examples: RA-DIT


Trainable
Trainable
Query
Who first won the Nobel Price
Retriever
The first Nobel Prizes were awarded...
LLM

RA-DIT: Retrieval-Augmented Dual Instruction Tuning, Lin et al, 2024

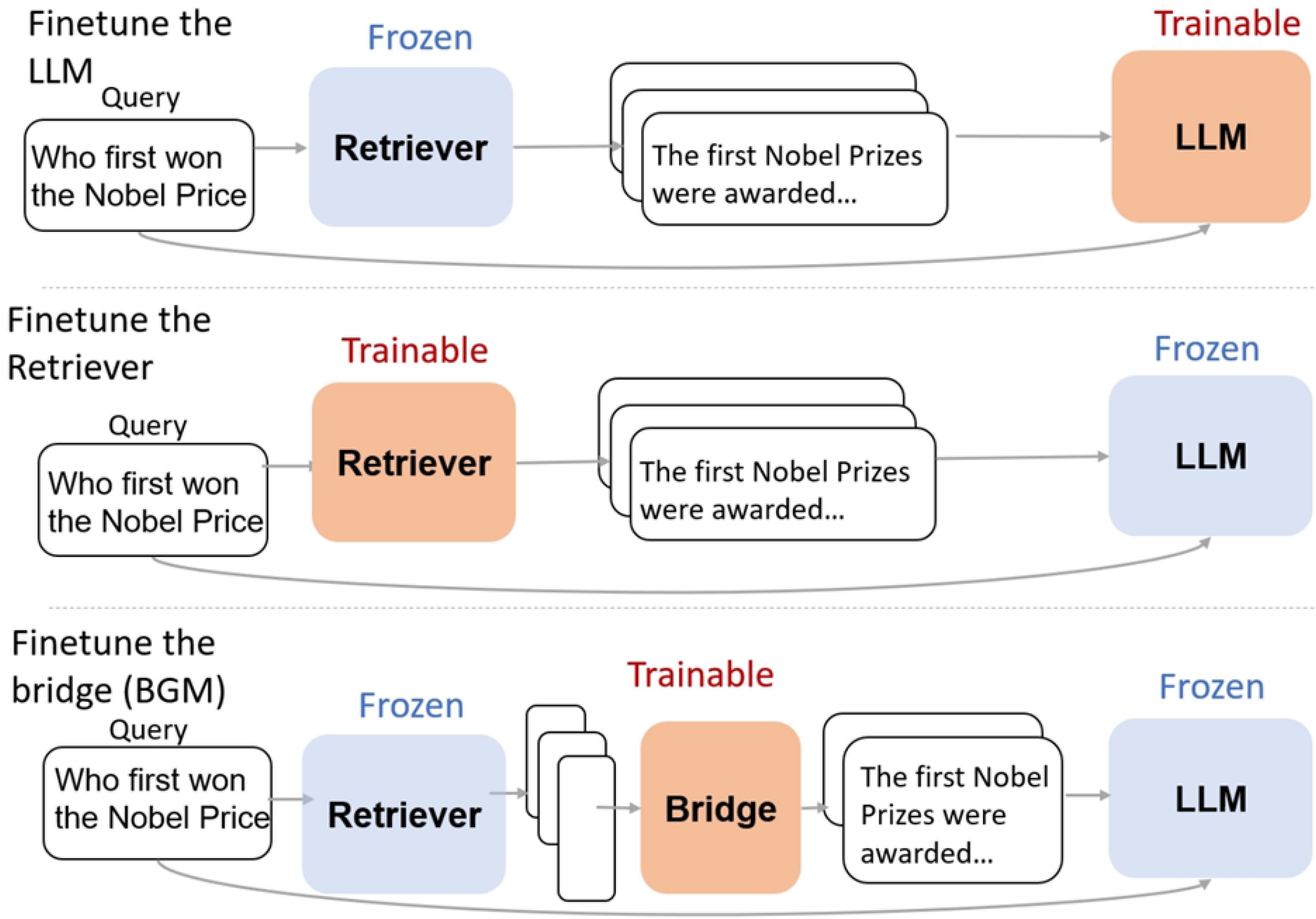
RAG – Key Ideas

LLM-Retriever Interaction

Fix the LLM and Retriver

Train a “bridge” (a LLM) to connect their preference

Main innovation: There is preference gap between **retriever** (built for human) and **LLM** (can prefer different order, selection..). One alternative way besides training LLM or retriever is to train an intermediate bridge


Finetune the LLM
Query
Who first won the Nobel Price
Frozen
Retriever
The first Nobel Prizes were awarded...
Trainable
LLM
Finetune the Retriever
Query
Who first won the Nobel Price
Trainable
Retriever
The first Nobel Prizes were awarded...
Frozen
LLM
Finetune the bridge (BGM)
Query
Who first won the Nobel Price
Frozen
Retriever
Trainable
Bridge
The first Nobel Prizes were awarded...
Frozen
LLM

Bridging the Preference Gap between Retrievers and LLMs, Ke et al., 2024

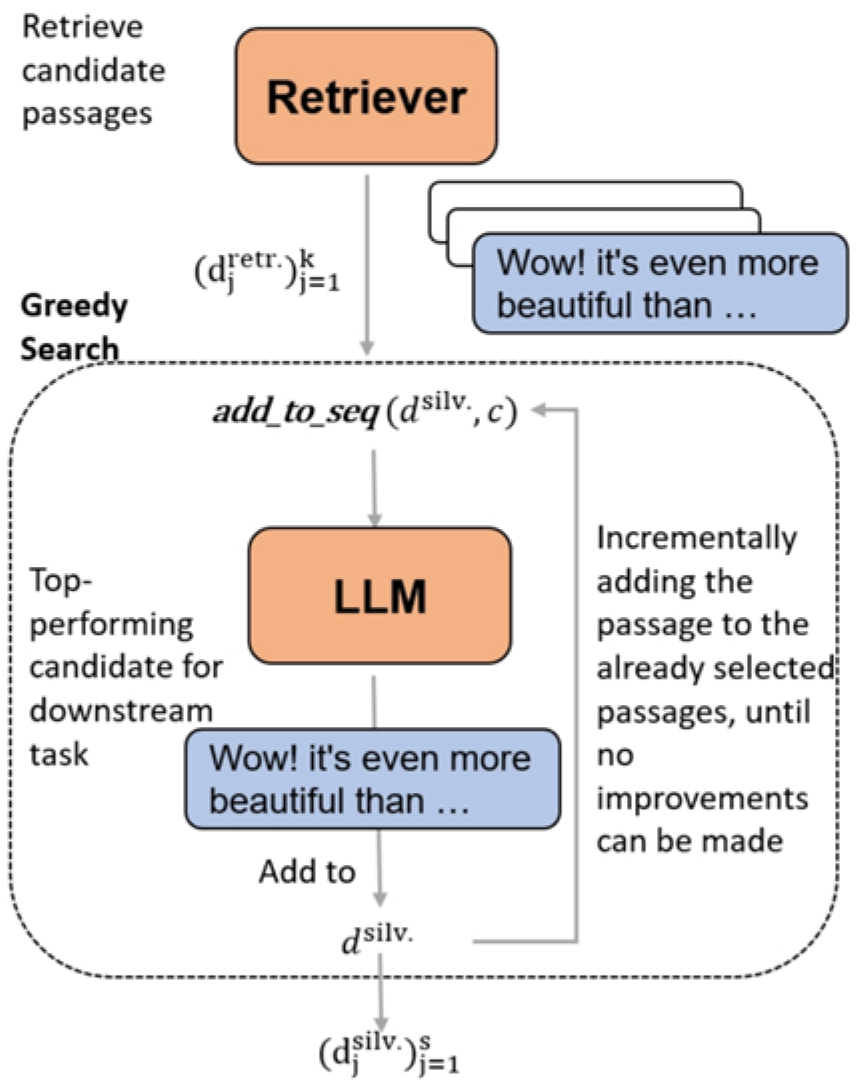
RAG – Key Ideas

LLM-Retriever Interaction

Ground Truth Data: Use greedy search to find the silver passage

Step 1

Collect silver passage sequence



Retrieve candidate passages
Retriever
$(d_j^{retr.})_{j=1}^{k}$
Wow! it's even more beautiful than …
Greedy Search
$add\_to\_seq\,(d^{silv.}, c)$
LLM
Top-performing candidate for downstream task
Wow! it's even more beautiful than …
Incrementally adding the passage to the already selected passages, until no improvements can be made
Add to
$d^{silv.}$
$(d_j^{silv.})_{j=1}^{s}$

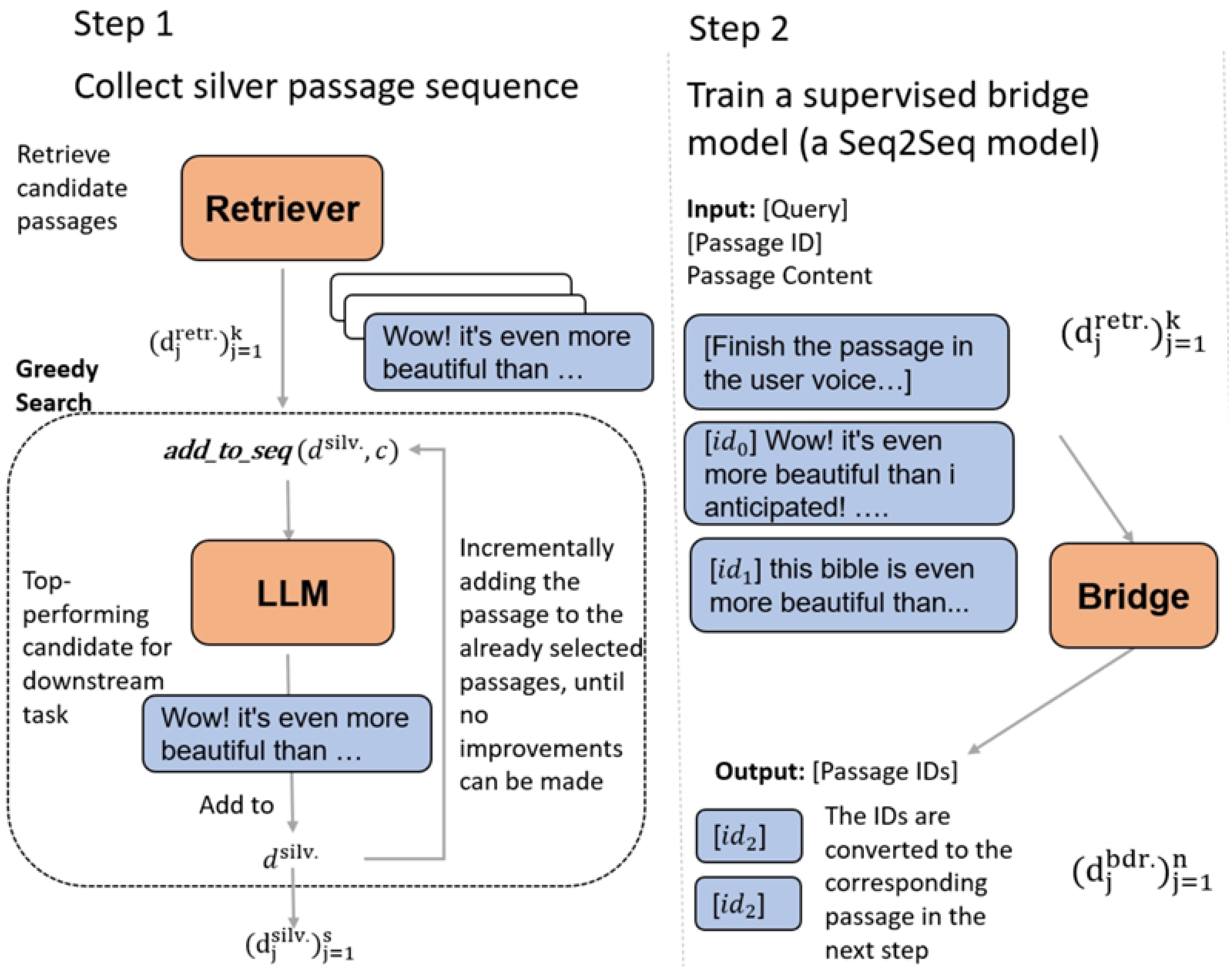
Bridging the Preference Gap between Retrievers and LLMs, Ke et al., 2024

RAG – Key Ideas

LLM-Retriever Interaction

Ground Truth Data: Use greedy search to find the silver passage

Workflow: IT → RL



Step 1

Collect silver passage sequence

Retrieve candidate passages

Retriever

$(d_j^{retr.})_{j=1}^{k}$

Wow! it's even more beautiful than …

Greedy Search

$add\_to\_seq\,(d^{silv.}, c)$

LLM

Top-performing candidate for downstream task

Wow! it's even more beautiful than …

Add to

$d^{silv.}$

Incrementally adding the passage to the already selected passages, until no improvements can be made

$(d_j^{silv.})_{j=1}^{s}$

Step 2

Train a supervised bridge model (a Seq2Seq model)

Input: [Query]
[Passage ID]
Passage Content

[Finish the passage in the user voice…]

$[id_0]$ Wow! it's even more beautiful than i anticipated! ….

$[id_1]$ this bible is even more beautiful than...

$(d_j^{retr.})_{j=1}^{k}$

Bridge

Output: [Passage IDs]

$[id_2]$

$[id_2]$

The IDs are converted to the corresponding passage in the next step

$(d_j^{bdr.})_{j=1}^{n}$

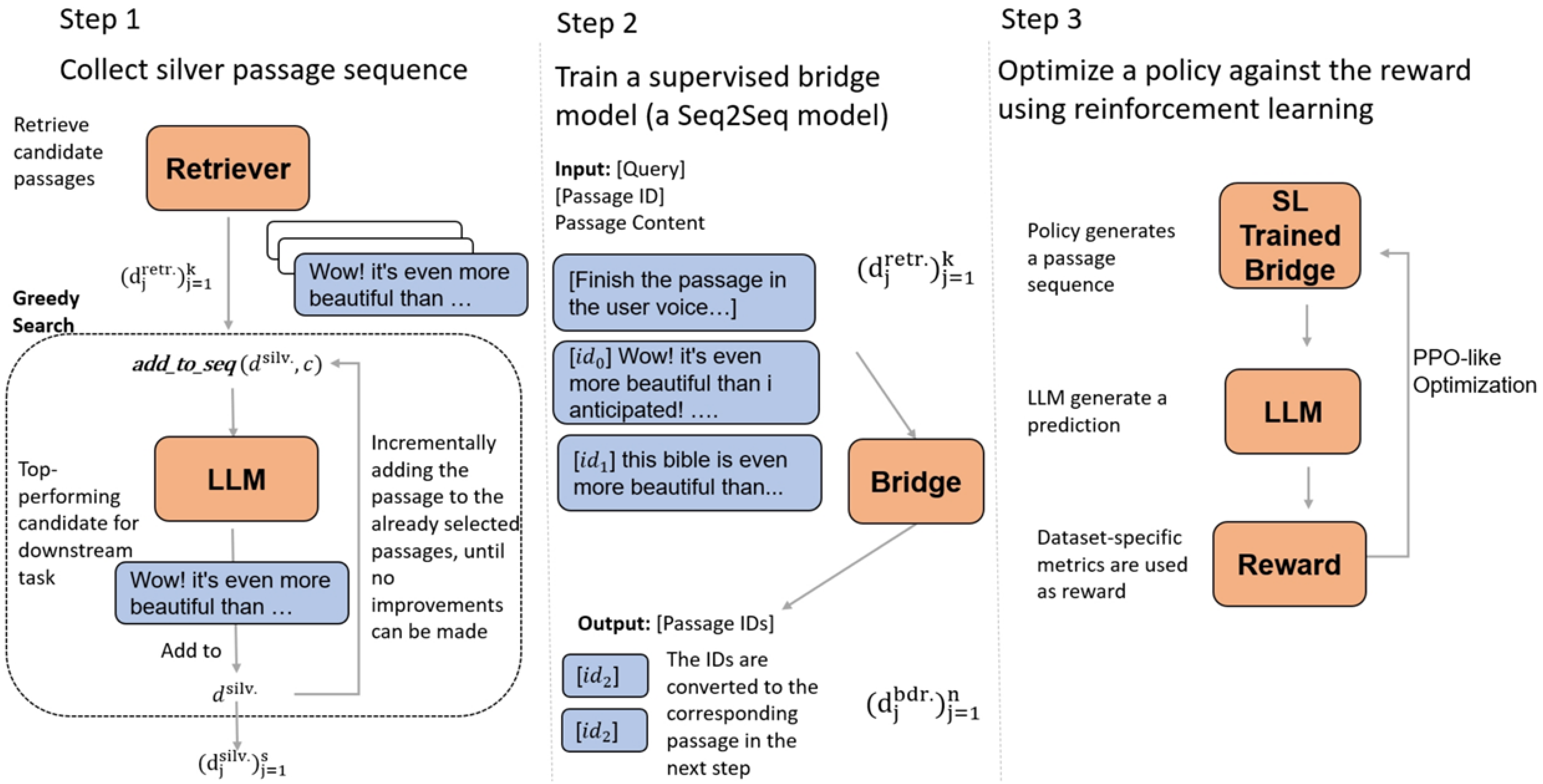
Bridging the Preference Gap between Retrievers and LLMs, Ke et al., 2024

RAG – Key Ideas

LLM-Retriever Interaction

Ground Truth Data: Use greedy search to find the silver passage

Workflow: IT → RL



Step 1

Collect silver passage sequence

Retrieve candidate passages
Retriever
$(d_j^{retr.})_{j=1}^{k}$
Wow! it's even more beautiful than …
Greedy Search
$add\_to\_seq(d^{silv.}, c)$
LLM
Top-performing candidate for downstream task
Wow! it's even more beautiful than …
Incrementally adding the passage to the already selected passages, until no improvements can be made
Add to
$d^{silv.}$
$(d_j^{silv.})_{j=1}^{s}$

Step 2

Train a supervised bridge model (a Seq2Seq model)

Input: [Query]
[Passage ID]
Passage Content
[Finish the passage in the user voice…]
$[id_0]$ Wow! it's even more beautiful than i anticipated! ….
$[id_1]$ this bible is even more beautiful than...
$(d_j^{retr.})_{j=1}^{k}$
Bridge
Output: [Passage IDs]
$[id_2]$
$[id_2]$
The IDs are converted to the corresponding passage in the next step
$(d_j^{bdr.})_{j=1}^{n}$

Step 3

Optimize a policy against the reward using reinforcement learning

Policy generates a passage sequence
SL Trained Bridge
LLM generate a prediction
LLM
Dataset-specific metrics are used as reward
Reward
PPO-like Optimization

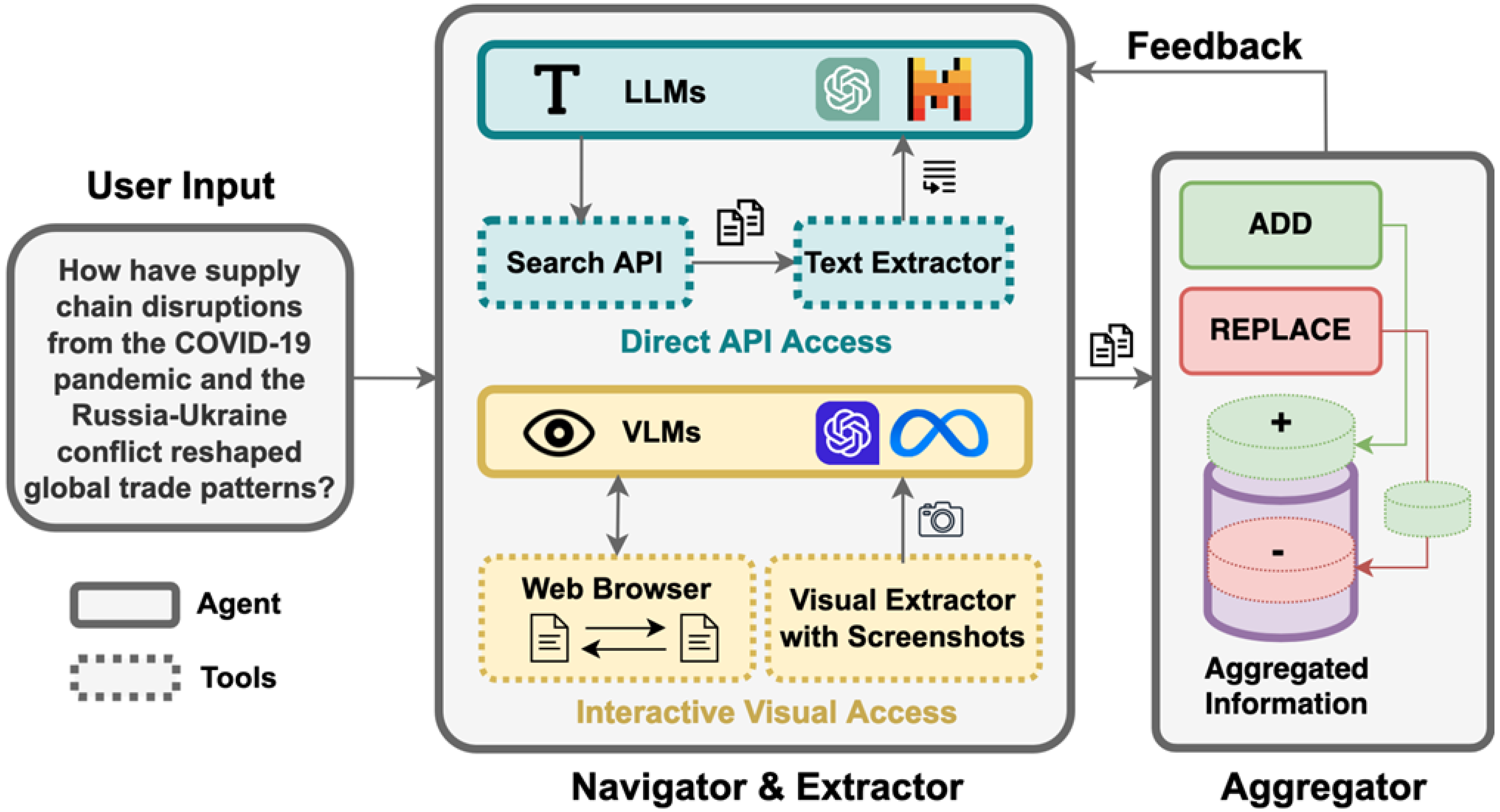
Bridging the Preference Gap between Retrievers and LLMs, Ke et al., 2024

Agentic RAG

RAG with Predefined Workflow

Main innovation: RAG can be performed in multiple predefined steps (workflow) to approach the final goal. Those steps usually involve API call, web browser, planner, etc.

Examples: Infogent, MindSearch


User Input
How have supply chain disruptions from the COVID-19 pandemic and the Russia-Ukraine conflict reshaped global trade patterns?
Agent
Tools
LLMs
Search API
Text Extractor
Direct API Access
VLMs
Web Browser
Visual Extractor with Screenshots
Interactive Visual Access
Navigator & Extractor
Feedback
ADD
REPLACE
+
-
Aggregated Information
Aggregator

INFOGENT: An Agent-Based Framework for Web Information Aggregation, Reddy, et al., 2024
MindSearch: Mimicking Human Minds Elicits Deep AI Searcher, Chen et al., 2024

RAG – Key Ideas Summary

Training Recipe

Data Recipe:

often use heuristic way to construct the ground truth

Model Recipe:

Algorithm and Workflow: so far, it is largely follows the parametric knowledge adaptation

Seed Data

Data Source: Knowledge-extensive tasks

Data Mixture: Can be large scale (e.g., Math, Logic, Code, Science, Reasoning..)

Data Budget: Follow the budget required in the specific method