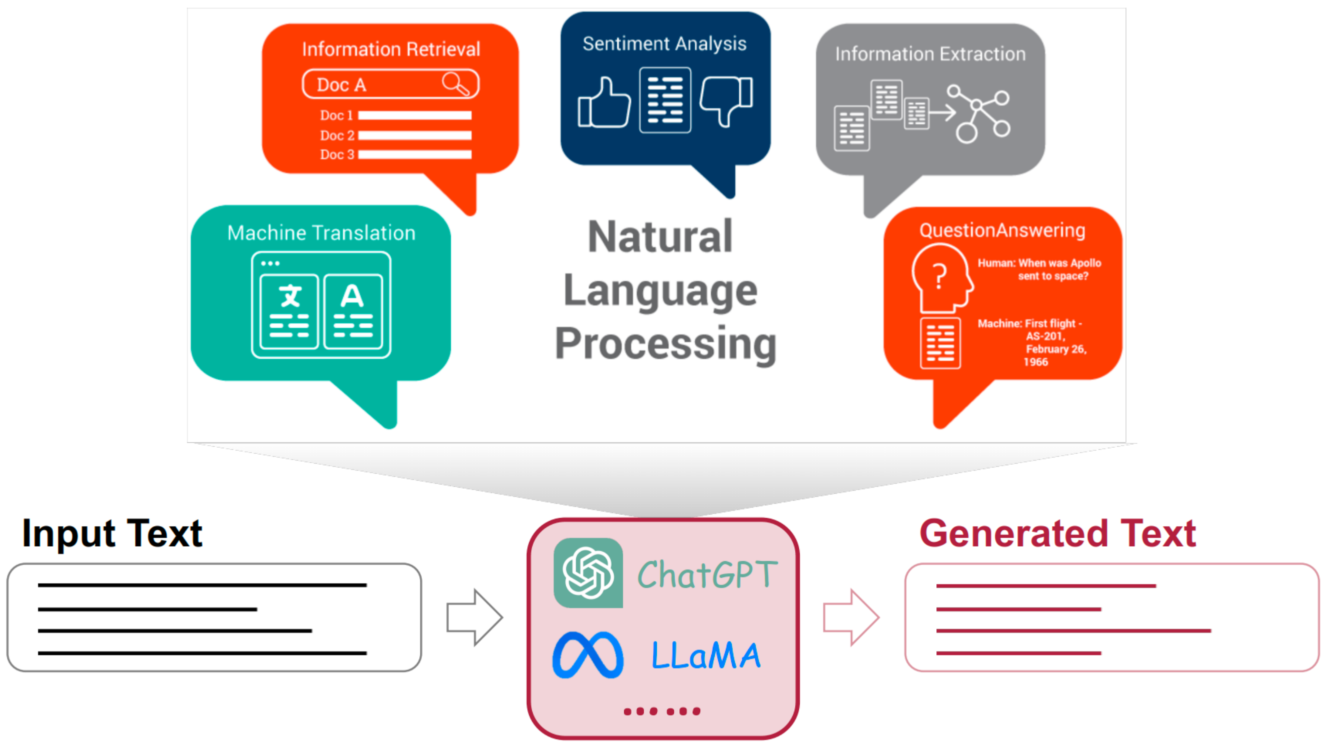
CS 2770: Retrieval Augmented Generation (RAG) for Large Language Models

PhD. Nils Murrugarra-Llerena
nem177@pitt.edu
University of Pittsburgh

Plan for this lecture

1. Introduction of Retrieval Augmented Large Language Models (RA LLMs)
2. Architecture of RA-LLMs and Main Modules
3. Learning Approach of RA-LLMs
4. Challenges and Future Directions of RA-LLMs

Large Language Models (LLMs)

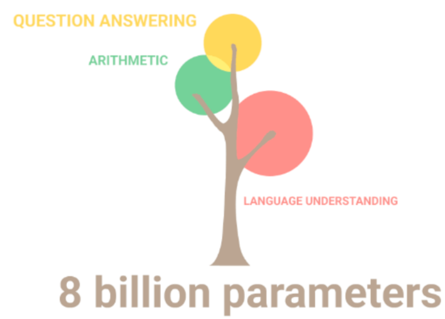


Large Language Models (LLMs)

RAG meet LLMS: Towards Retrieval-Augmented LLMS Tutorial @ KDD 24 - https://advanced-recommender-systems.github.io/RAG-Meets-LLMs/

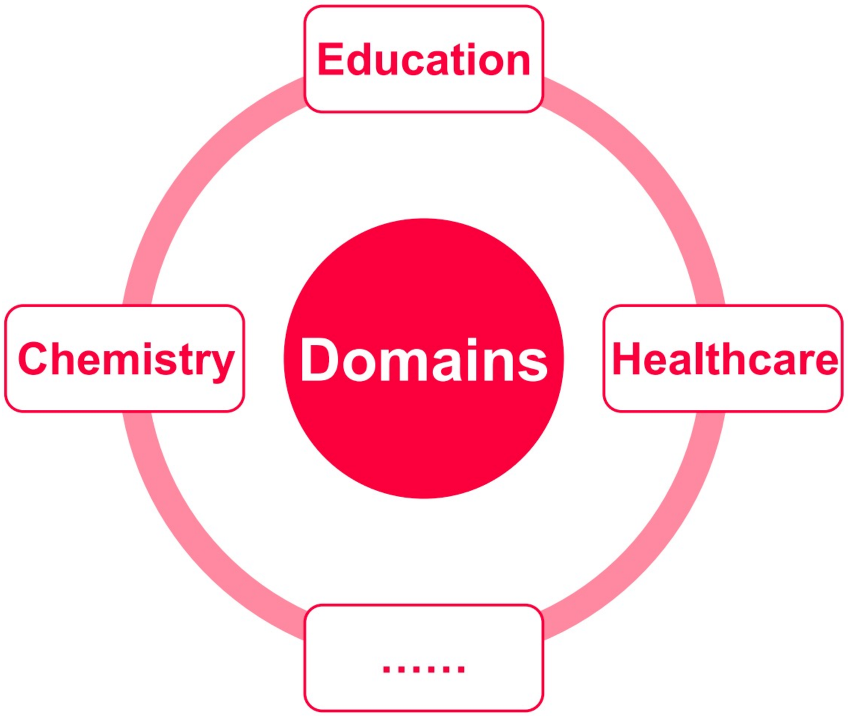
Large Language Models (LLMs)

OpenAI | Meta | Google AI | T5 | ......

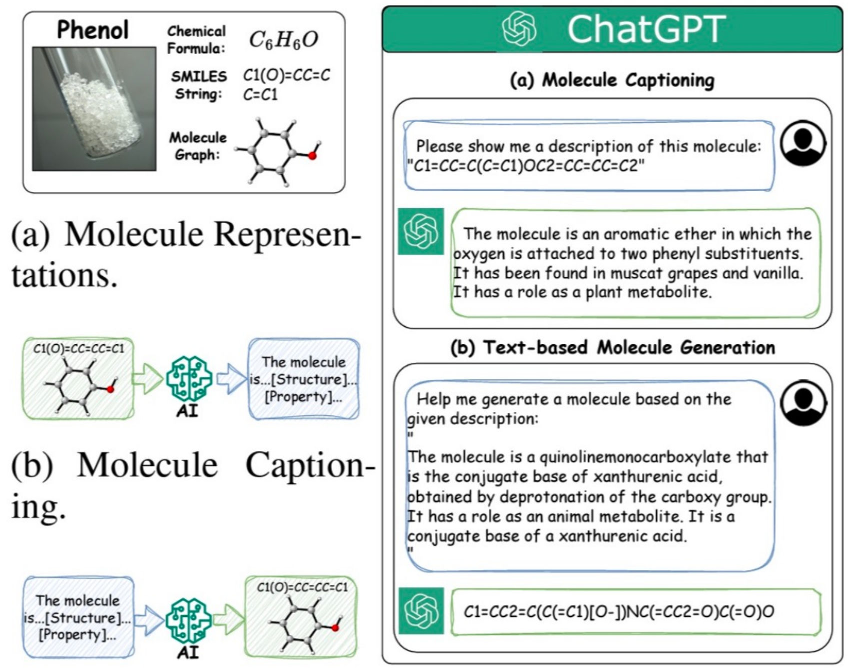


https://github.com/Hannibal046/Awesome-LLM/

LLMs in Downstream Domains

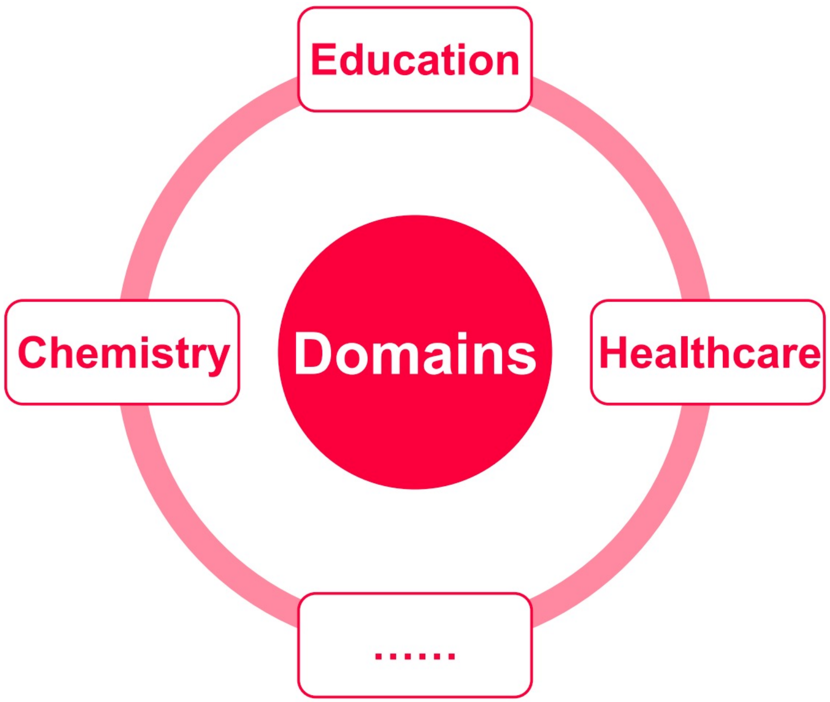


❑ **Molecule discovery**, etc.

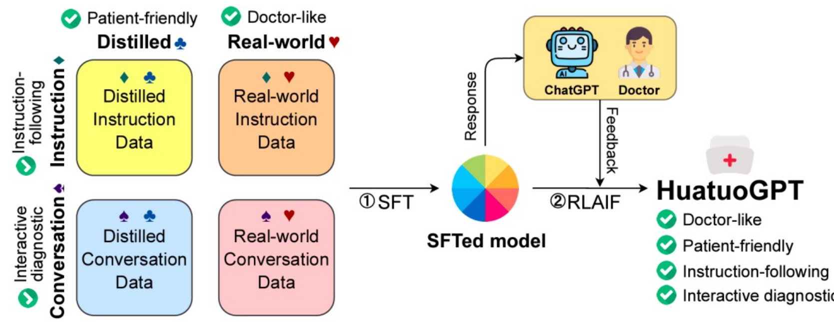


Li et al, 2024, Empowering Molecule Discovery for Molecule-Caption Translation with Large Language Models: A ChatGPT Perspective,
Liu et al., 2024, MolecularGPT: Open Large Language Model (LLM) for Few-Shot Molecular Property Prediction,

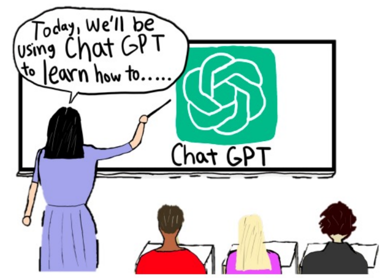
LLMs in Downstream Domains



❑ **Medical consultation**, etc.



❑ **Curriculum & Teaching**, etc.



Zhang et al., 2023, HuatuoGPT, towards Taming Language Model to Be a Doctor

Challenges and Risks of LLMs

Hallucination
The generation of inaccurate, nonsensical, or detached text, posing potential risks and challenges for organizations utilizing these models.

Domain-specific knowledge & expertise
LLMs might not perform well in many domain-specific fields like medicine, law, finance, and more, because of the lack of domain-specific knowledge and expertise.

Privacy
Various risks to data privacy and security exist at different stages of LLMs, which becomes particularly acute in light of incidents where sensitive internal data was exposed to LLMs.

Inconsistency
Sometimes they nail the answer to questions, other times they regurgitate random facts from their training data.

9

LLMs' Challenges in Vertical Domains

- Domain of Law

OXFORD

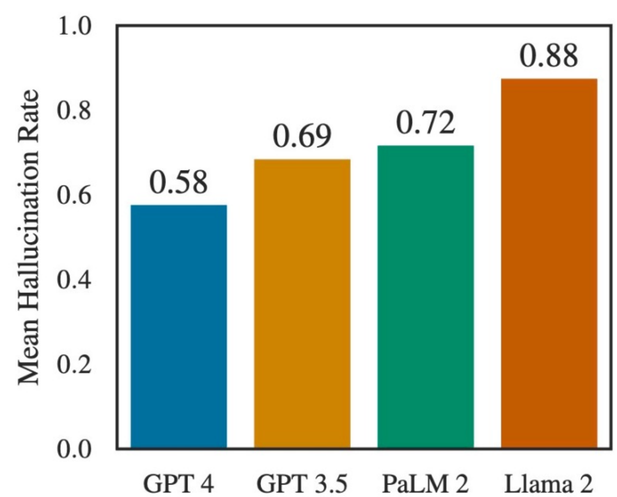
Journal of Legal Analysis, 2024, **16**, 64–93
https://doi.org/10.1093/jla/laae003
Advance access publication 26 June 2024
Article

Large Legal Fictions: Profiling Legal Hallucinations in Large Language Models

Matthew Dahl[*], Varun Magesh[†], Mirac Suzgun[‡], and Daniel E. Ho[§]

In a new study by ***Stanford RegLab*** *and* ***Institute for Human-Centered AI*** *researchers, it is demonstrated that legal hallucinations are pervasive and disturbing:* ***hallucination rates range from 69% to 88% in response to specific legal queries*** *for state-of-the-art language models.*

Hallucinations are common across all LLMs when they are asked a direct, verifiable question about a federal court case

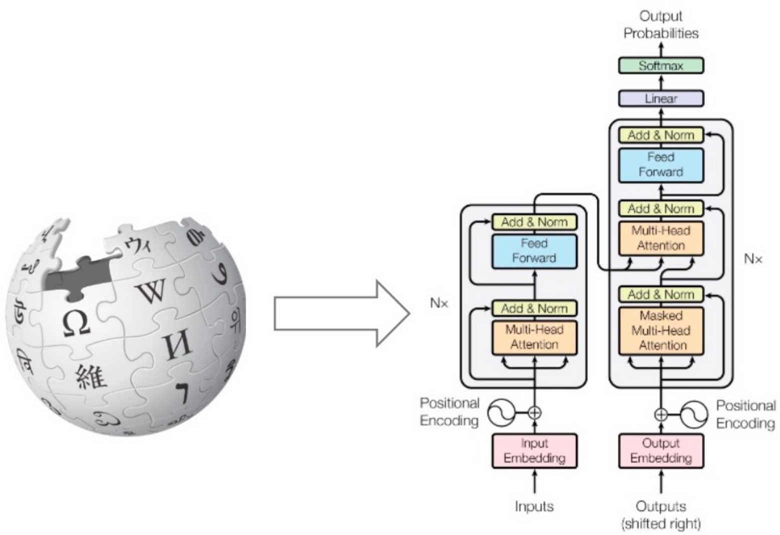


Dahl M, et al. 2024, Large legal fictions: Profiling legal hallucinations in large language models.

Why Large Language Models Work Well?

- ❑ Big Model + Big Training Data

Storing knowledge in the parametric model !



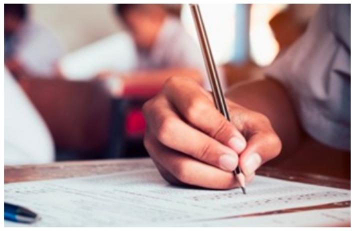
Retrieval-Augmented Large Language Models (RA-LLMs)

- LLMs **cannot memorize all** (particularly long-tail) knowledge in their parameters
- Lack of **domain-specific knowledge**, **updated information**, etc

Hallucination & Unable to answer → Re-training / Finetuning ?

Retrieval-Augmented Large Language Models (RA-LLMs)

Data for Training LLMs
- Low quality
- General
- Fixed
- Hard to update



Content generation

Close-book exam
(Hard mode, have to ***remember everything****)*

Plan for this lecture

1. Introduction of Retrieval Augmented Large Language Models (RA LLMs)

2. Architecture of RA-LLMs and Main Modules

3. Learning Approach of RA-LLMs

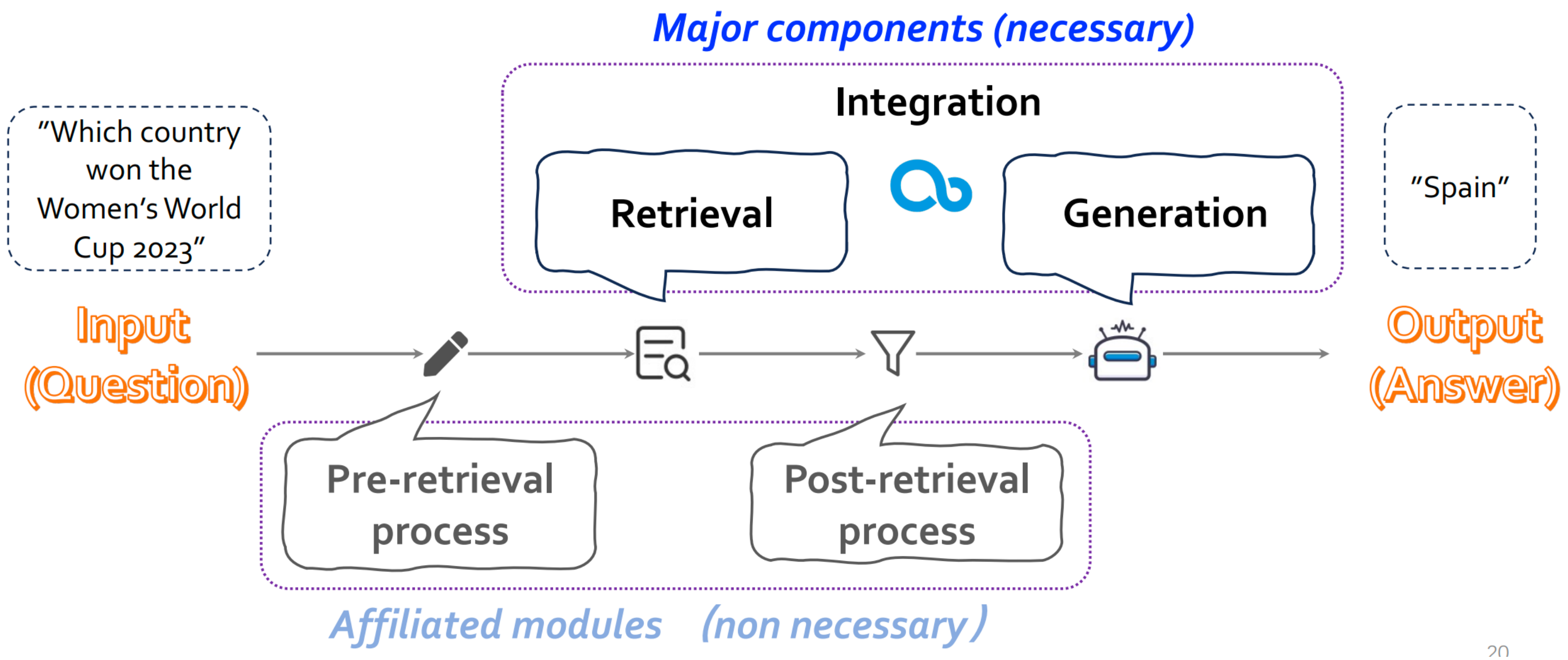
4. Challenges and Future Directions of RA-LLMs

Plan for this lecture

1. RA-LLM architecture overview
2. Retriever in RA-LLMs
3. Retrieval results integration
4. Pre/Post-retrieval techniques

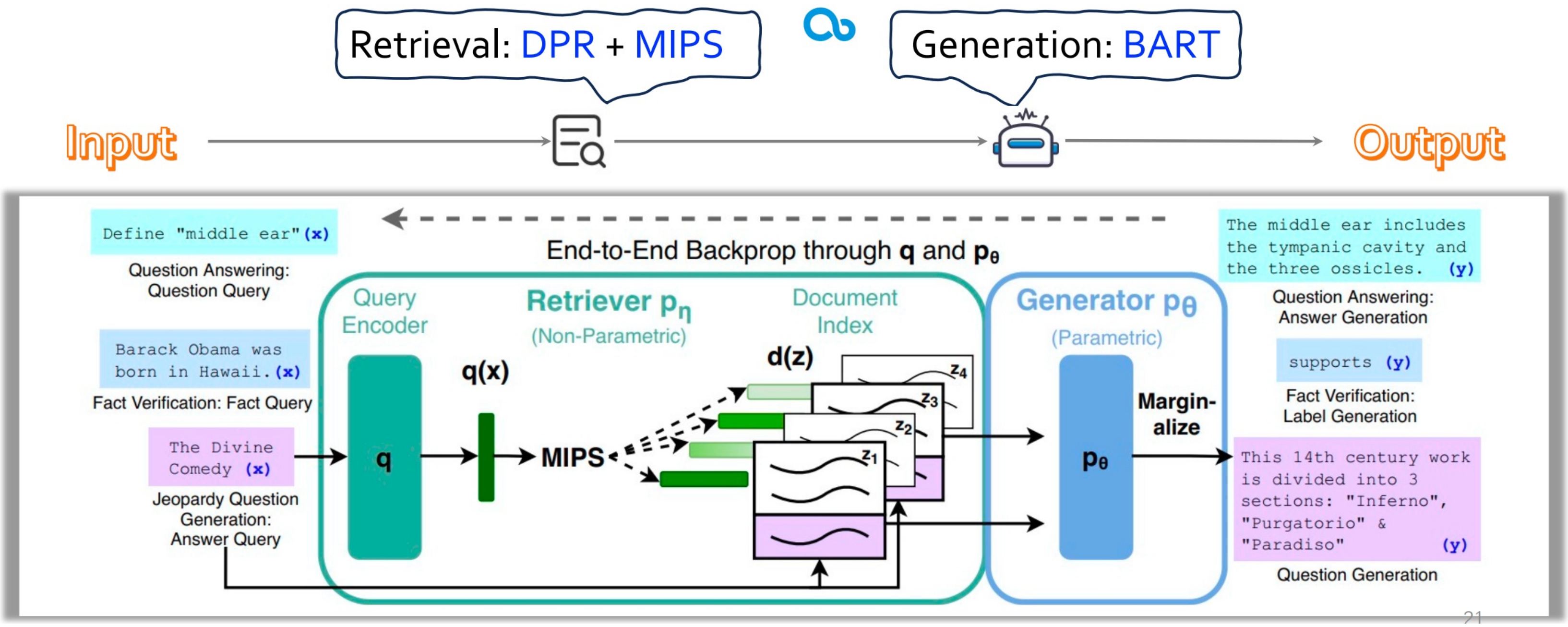
RA-LLM Architecture: Standard Pipeline

- Technical component illustration in a RA-LLM for the Q&A task



A Simple Retrieval-Augmented Generation Model

- RAG Integration: concatenating each retrieved passage with the question

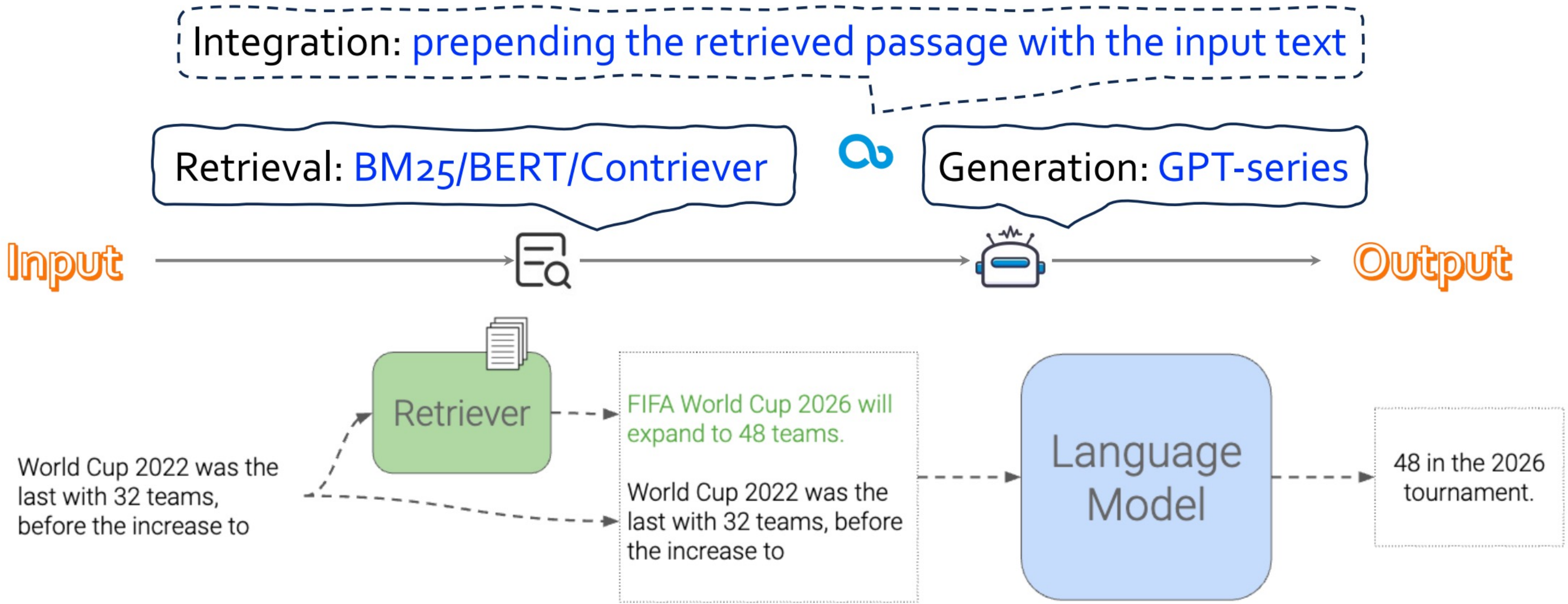


Lewis et al. 2020. "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks"

RAG meet LLMS: Towards Retrieval-Augmented LLMS Tutorial @ KDD 24 - https://advanced-recommender-systems.github.io/RAG-Meets-LLMs/

A Simple Retrieval-Augmented Generation Model

- In-Context RALM



Ram et al. 2023, In-Context Retrieval-Augmented Language Models

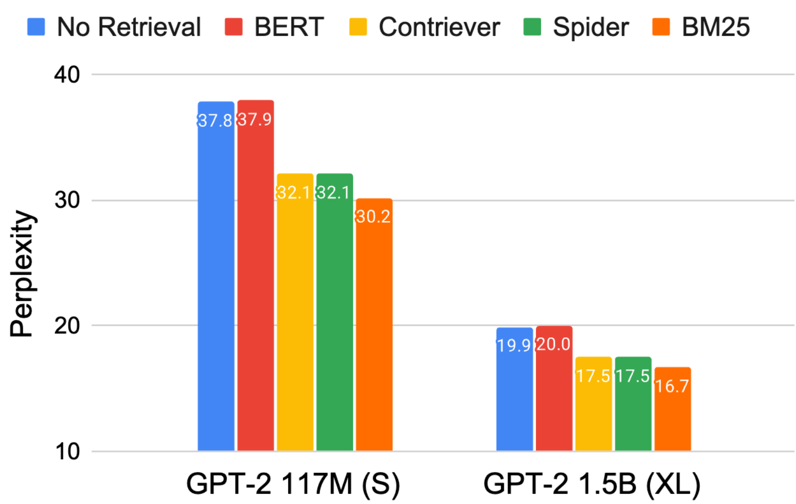
RAG meet LLMS: Towards Retrieval-Augmented LLMS Tutorial @ KDD 24 - https://advanced-recommender-systems.github.io/RAG-Meets-LLMs/

Plan for this lecture

1. RA-LLM architecture overview
2. Retriever in RA-LLMs
3. Retrieval results integration
4. Pre/Post-retrieval techniques

RA-LLM Architecture: Retriever Types

- Different types of retriever deliver different generation performance



Relevance measurement	Retriever learning
Sparse	Task-specific pre-trained
Dense	General-purpose pre-trained

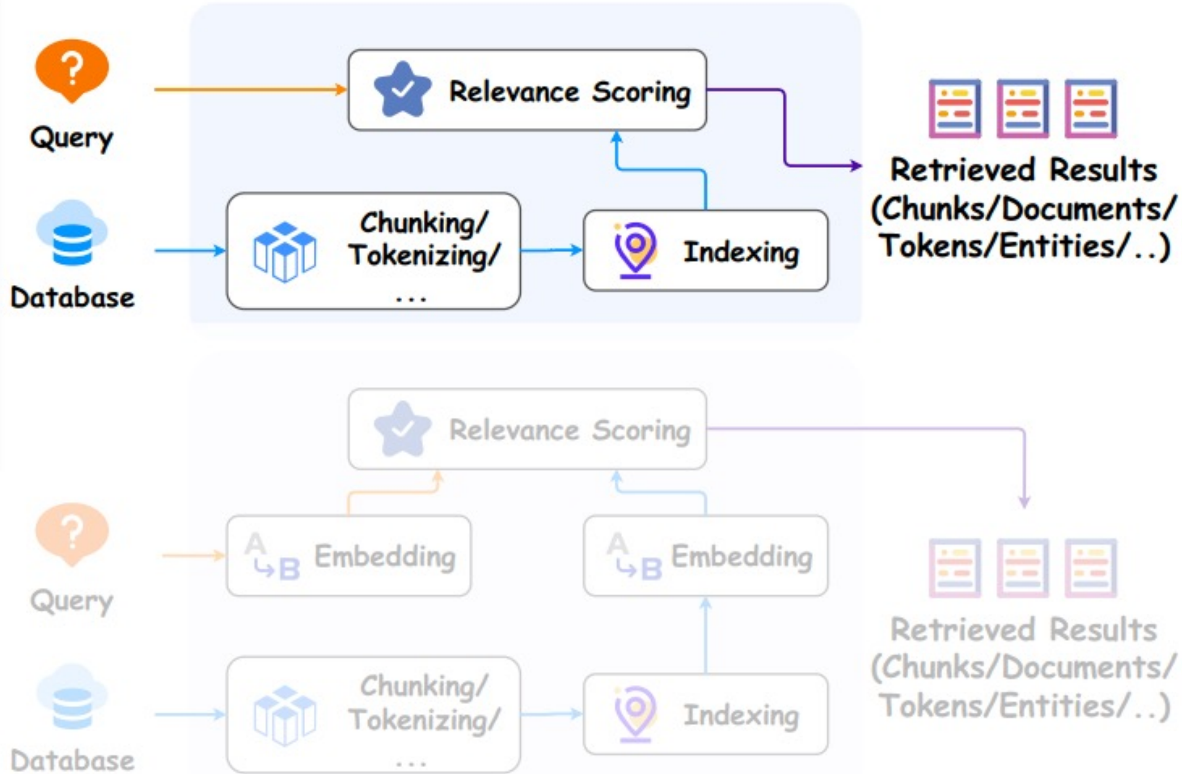
Ram et al. 2023, In-Context Retrieval-Augmented Language Models

RAG meet LLMS: Towards Retrieval-Augmented LLMS Tutorial @ KDD 24 - https://advanced-recommender-systems.github.io/RAG-Meets-LLMs/

Dense v.s. Sparse Retrievers

Sparse Retrievers (SR)

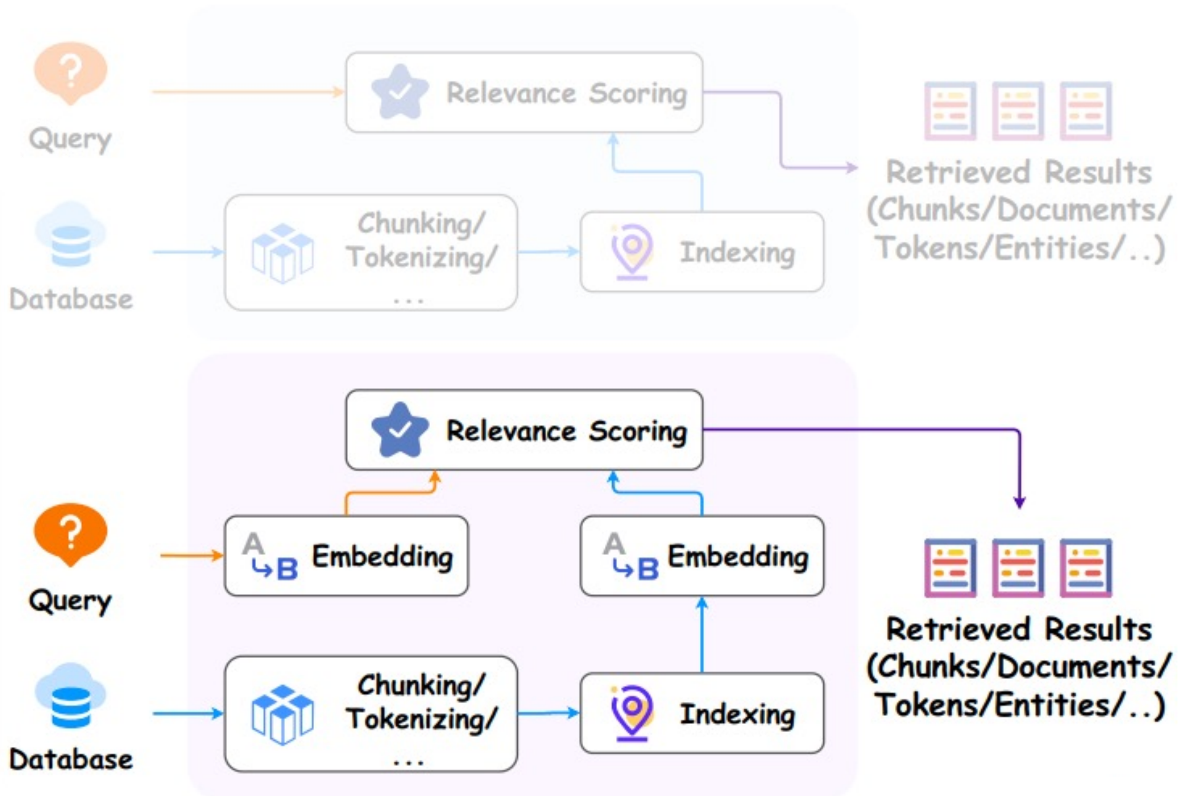
- Feasible to apply
- High efficiency
- Fine performance
- Example: TF-IDF, BM25



Dense v.s. Sparse Retrievers

Dense Retrievers (DR)

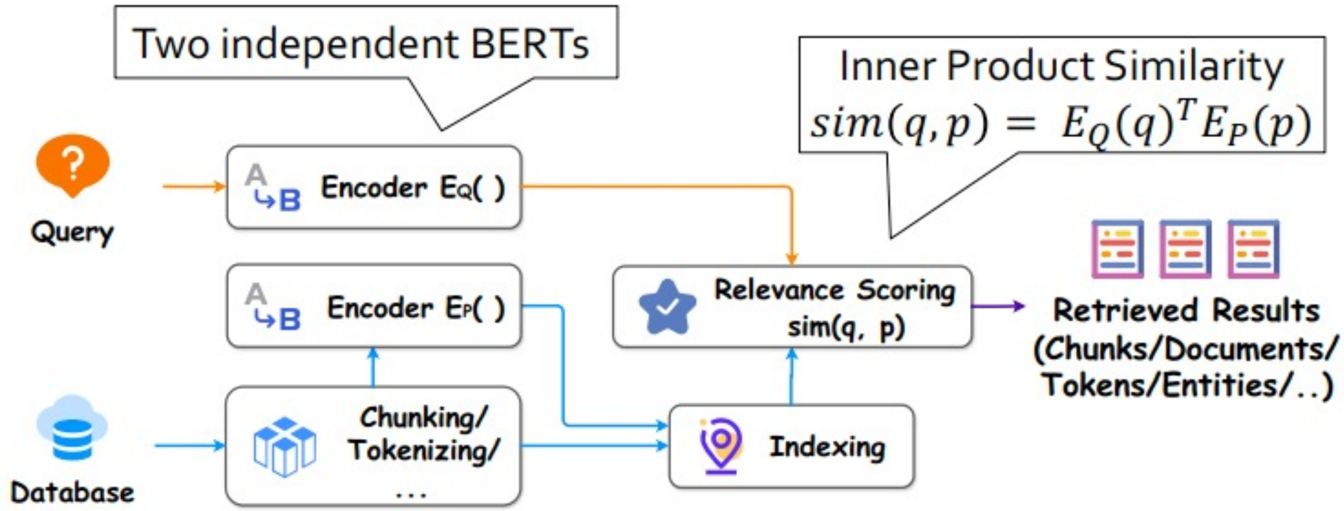
- Allowing fine-tuning
- Better adaptation
- Customizable for more retrieval goals
- Example: DPR, Contriever



RAG meet LLMS: Towards Retrieval-Augmented LLMS Tutorial @ KDD 24 - https://advanced-recommender-systems.github.io/RAG-Meets-LLMs/

Task-Specific Pre-trained Retriever (Supervised)

- **Dense Passage Retriever (DPR):** Pretrained for Question Answering (QA)



Karpukhin et al. 2020. "Dense Passage Retrieval for Open-Domain Question Answering"

RAG meet LLMS: Towards Retrieval-Augmented LLMS Tutorial @ KDD 24 - https://advanced-recommender-systems.github.io/RAG-Meets-LLMs/

Task-Specific Pre-trained Retriever (Supervised)

- **Dense Passage Retriever (DPR):** Pretrained for Question Answering (QA)

  - Learning Objective

$$L(q_i, p_i^+, p_{i,1}^-, \cdots, p_{i,n}^-)$$
$$= \quad -\log \frac{e^{\mathrm{sim}(q_i, p_i^+)}}{e^{\mathrm{sim}(q_i, p_i^+)} + \sum_{j=1}^{n} e^{\mathrm{sim}(q_i, p_{i,j}^-)}}$$

  - Training data: Question-Passage Sets

$$\mathcal{D} = \{\langle \boxed{q_i}, \boxed{p_i^+}, \boxed{p_{i,1}^-, \cdots, p_{i,n}^-} \rangle\}_{i=1}^{m}$$

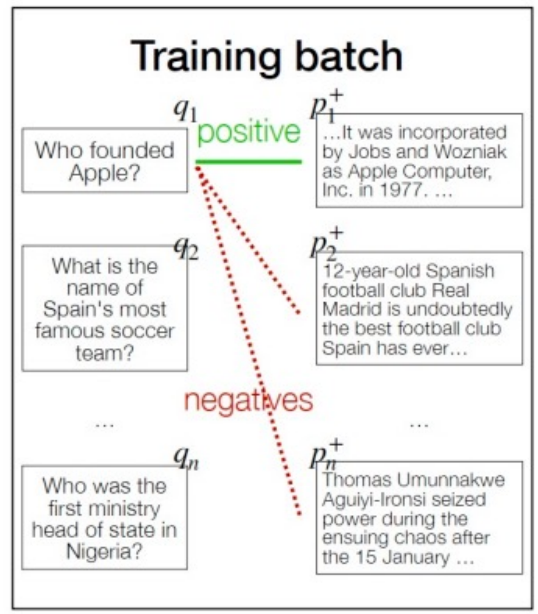
Question

Irrelevant passages

Relevant passage

Negative sample selection?

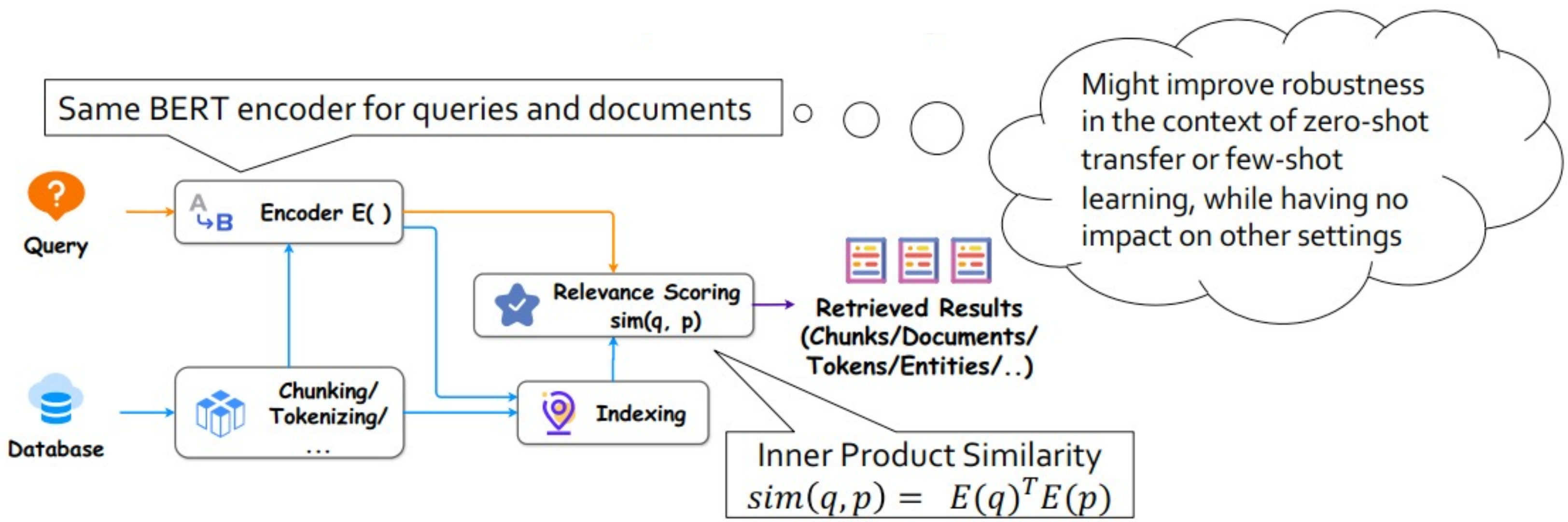
  - Training with in-batch negatives



Karpukhin et al. 2020. "Dense Passage Retrieval for Open-Domain Question Answering"

General-Purpose Pre-trained Retriever (Unsupervised)

- **Contriever**: Pre-trained with unsupervised learning



Contrastive learning with unaligned documents

$$\mathcal{L}(q, k_+) = -\frac{\exp(s(q, k_+)/\tau)}{\exp(s(q, k_+)/\tau) + \sum_{i=1}^{K} \exp(s(q, k_i)/\tau)}$$

Izacard et al. 2022. "Unsupervised Dense Information Retrieval with Contrastive Learning

DPR & Contriever Performance on OpenQA Tasks

End-to-end QA (Exact Match) Accuracy

Training	Model	NQ	TriviaQA	WQ	TREC	SQuAD
Single	BM25+BERT (Lee et al., 2019)	26.5	47.1	17.7	21.3	33.2
Single	ORQA (Lee et al., 2019)	33.3	45.0	36.4	30.1	20.2
Single	HardEM (Min et al., 2019a)	28.1	50.9	-	-	-
Single	GraphRetriever (Min et al., 2019b)	34.5	56.0	36.4	-	-
Single	PathRetriever (Asai et al., 2020)	32.6	-	-	-	**56.5**
Single	$\text{REALM}_{\text{Wiki}}$ (Guu et al., 2020)	39.2	-	40.2	46.8	-
Single	$\text{REALM}_{\text{News}}$ (Guu et al., 2020)	40.4	-	40.7	42.9	-
	BM25	32.6	52.4	29.9	24.9	38.1
Single	DPR	**41.5**	56.8	34.6	25.9	29.8
	BM25+DPR	39.0	57.0	35.2	28.0	36.7
Multi	DPR	**41.5**	56.8	**42.4**	49.4	24.1
	BM25+DPR	38.8	**57.9**	41.1	**50.6**	35.8

Both widely applied in RAG and RA-LLMs

DPR in

RAG, FiD, RETRO, EPR, UDR, …

Contriever in

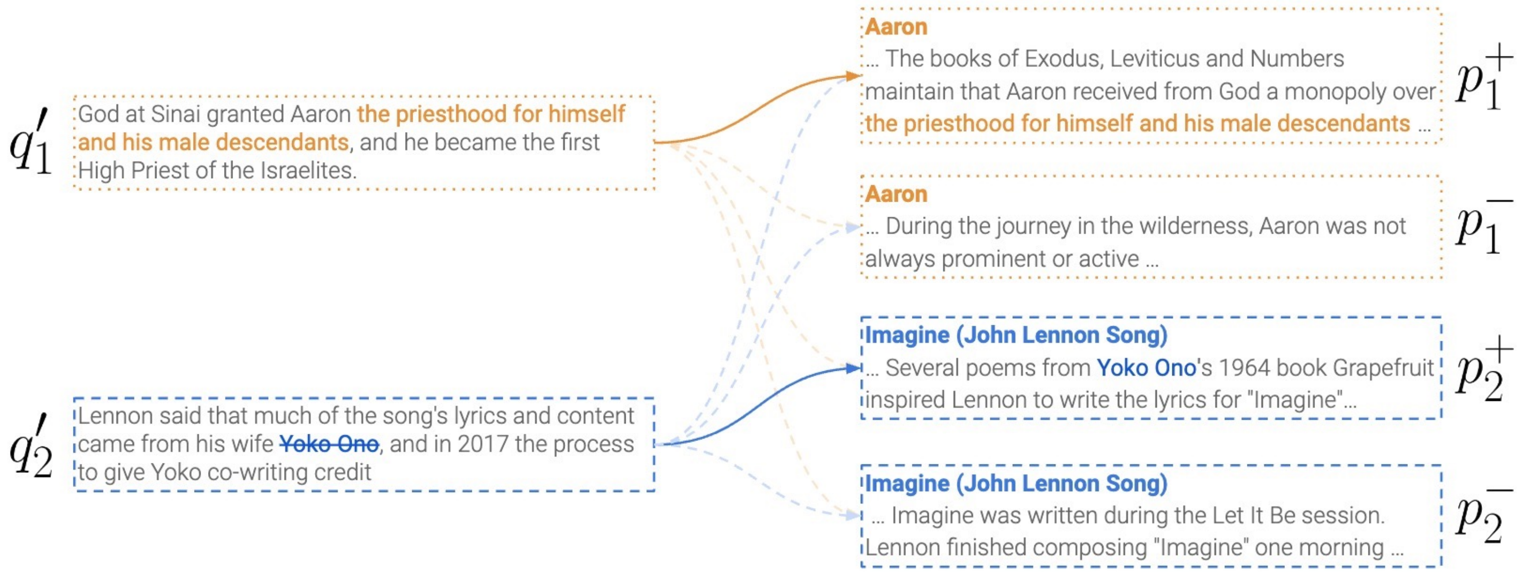
Self-RAG, Atlas, RAVEN, …

Both better than the sparse retriever!

	NaturalQuestions			TriviaQA		
	R@5	R@20	R@100	R@5	R@20	R@100
Inverse Cloze Task (Sachan et al., 2021)	32.3	50.9	66.8	40.2	57.5	73.6
Masked salient spans (Sachan et al., 2021)	41.7	59.8	74.9	53.3	68.2	79.4
BM25 (Ma et al., 2021)	-	62.9	78.3	-	**76.4**	**83.2**
Contriever	**47.8**	**67.8**	**82.1**	**59.4**	74.2	**83.2**
supervised model: DPR (Karpukhin et al., 2020)	-	78.4	85.4	-	79.4	85.0

Task-Specific Pre-trained Retriever (Unsupervised)

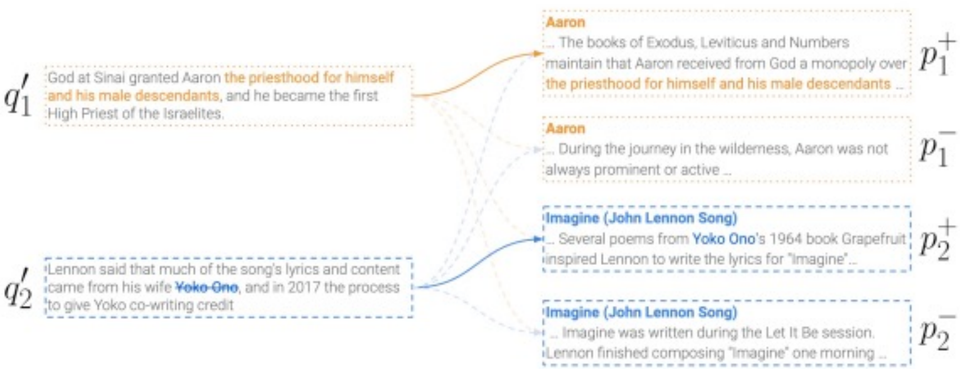
- **Spider** (Span-based unsupervised dense retriever)
  Recurring Span Retrieval: It is based on the notion of recurring spans within a document: given two paragraphs with the same recurring span, we construct a query from one of the paragraphs, while the other is taken as the target for retrieval



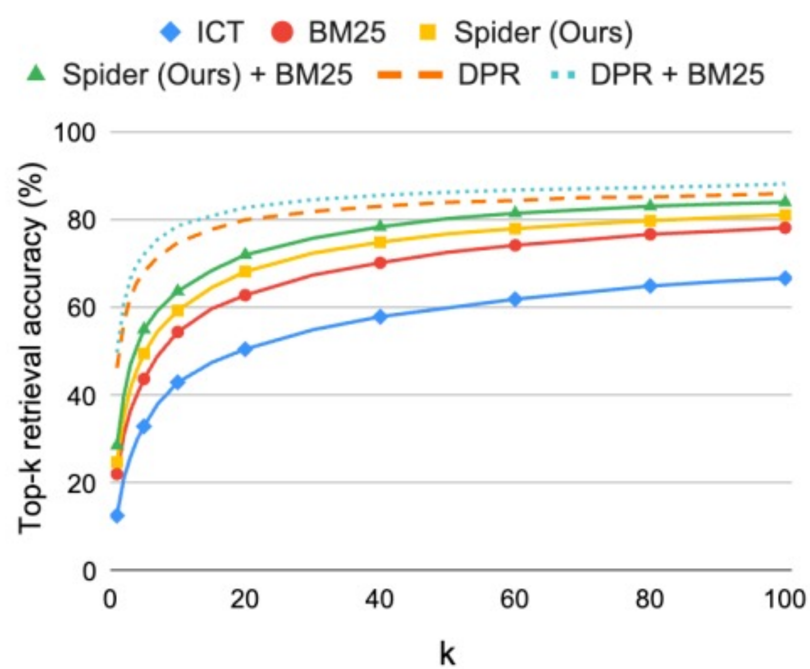
Ram et al., 2022, Learning to Retrieve Passages without Supervision

Task-Specific Pre-trained Retriever (Unsupervised)

- Learning and results of Spider



$$-\log \frac{\exp\left(s(q_i', p_i^+)\right)}{\sum_{j=1}^{m}\left(\exp\left(s(q_i', p_j^+)\right) + \exp\left(s(q_i', p_j^-)\right)\right)}$$



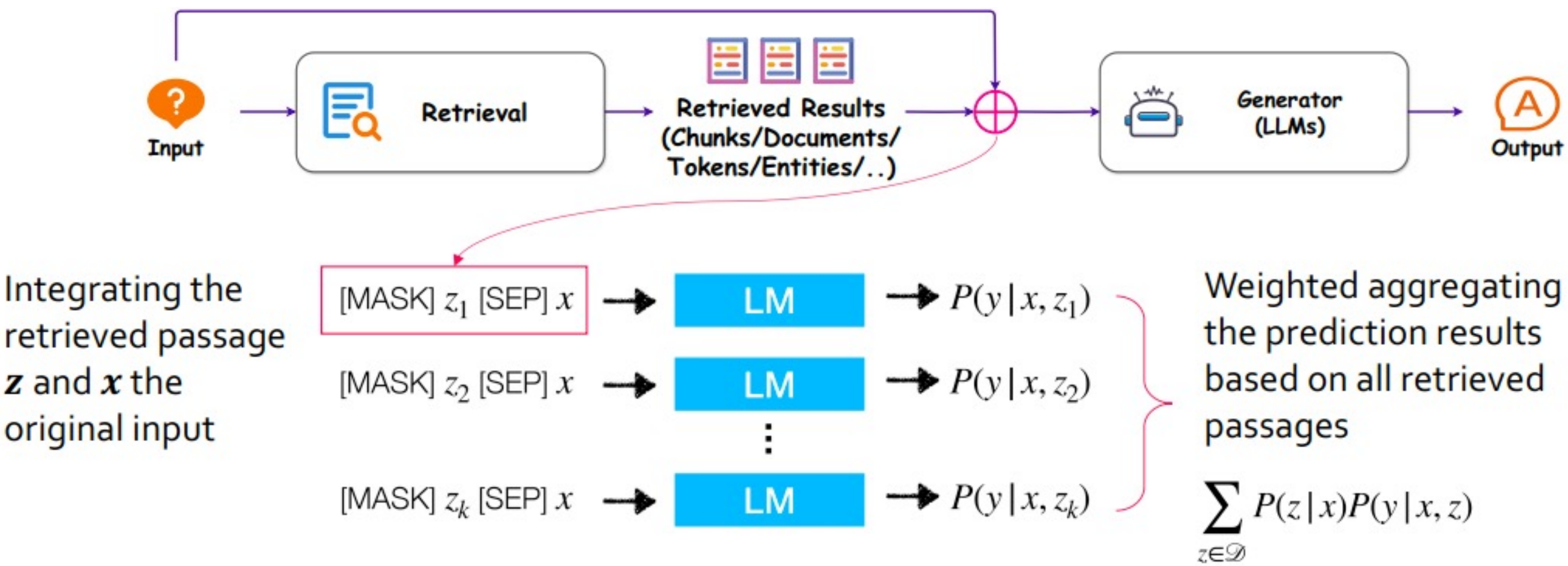
Ram et al., 2022, Learning to Retrieve Passages without Supervision

Plan for this lecture

1. RA-LLM architecture overview
2. Retriever in RA-LLMs
3. Retrieval results integration
4. Pre/Post-retrieval techniques

Retrieved Results Integration: Input-layer Integration

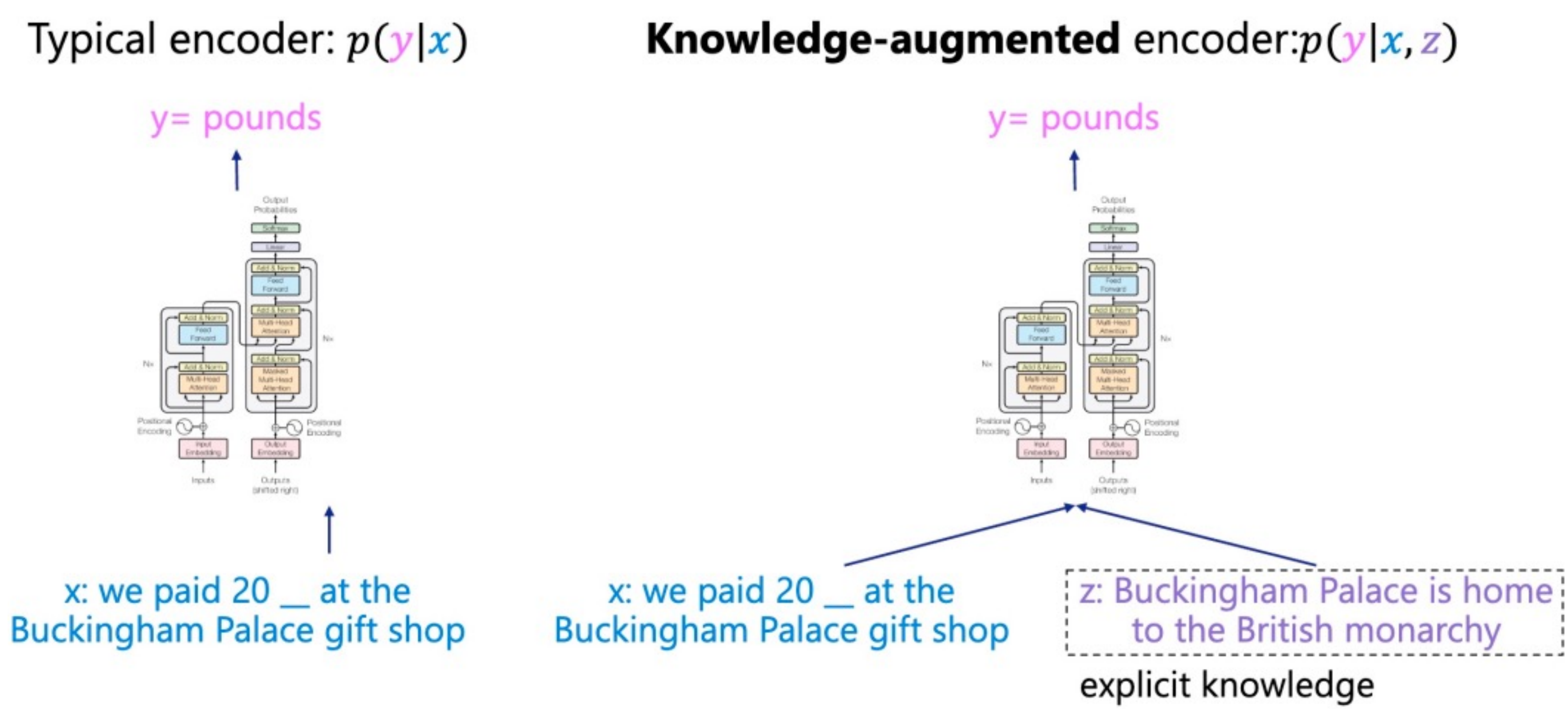
- REALM



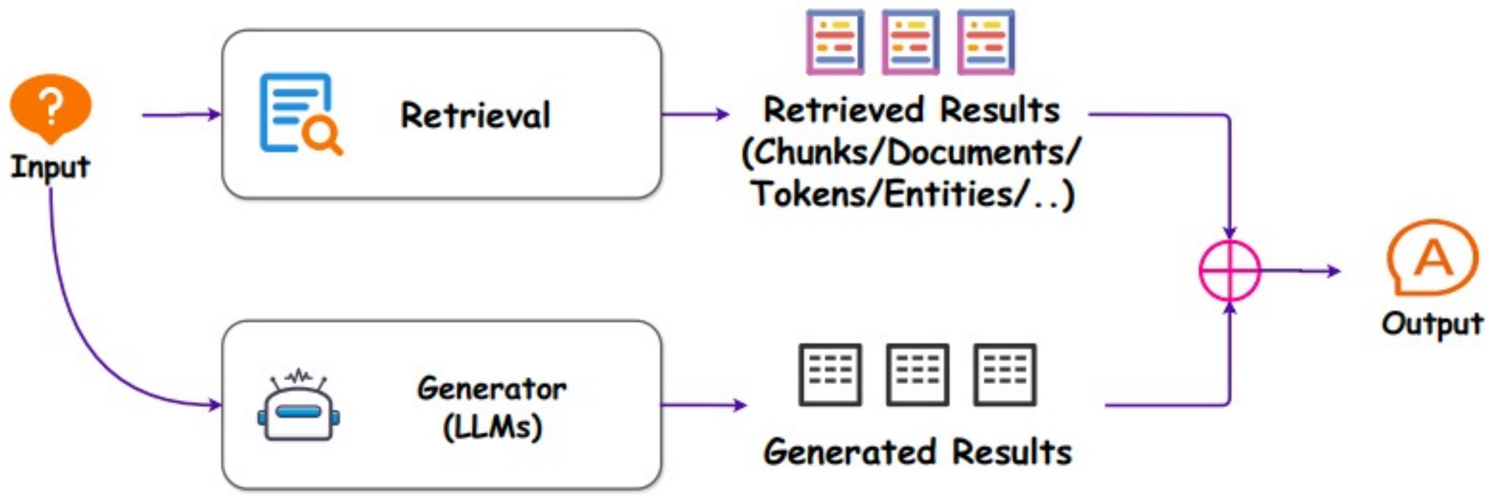
Guu et al. 2020. "REALM: Retrieval-Augmented Language Model Pre-Training"

RAG meet LLMS: Towards Retrieval-Augmented LLMS Tutorial @ KDD 24 - https://advanced-recommender-systems.github.io/RAG-Meets-LLMs/

Retrieval-Augmented Generator

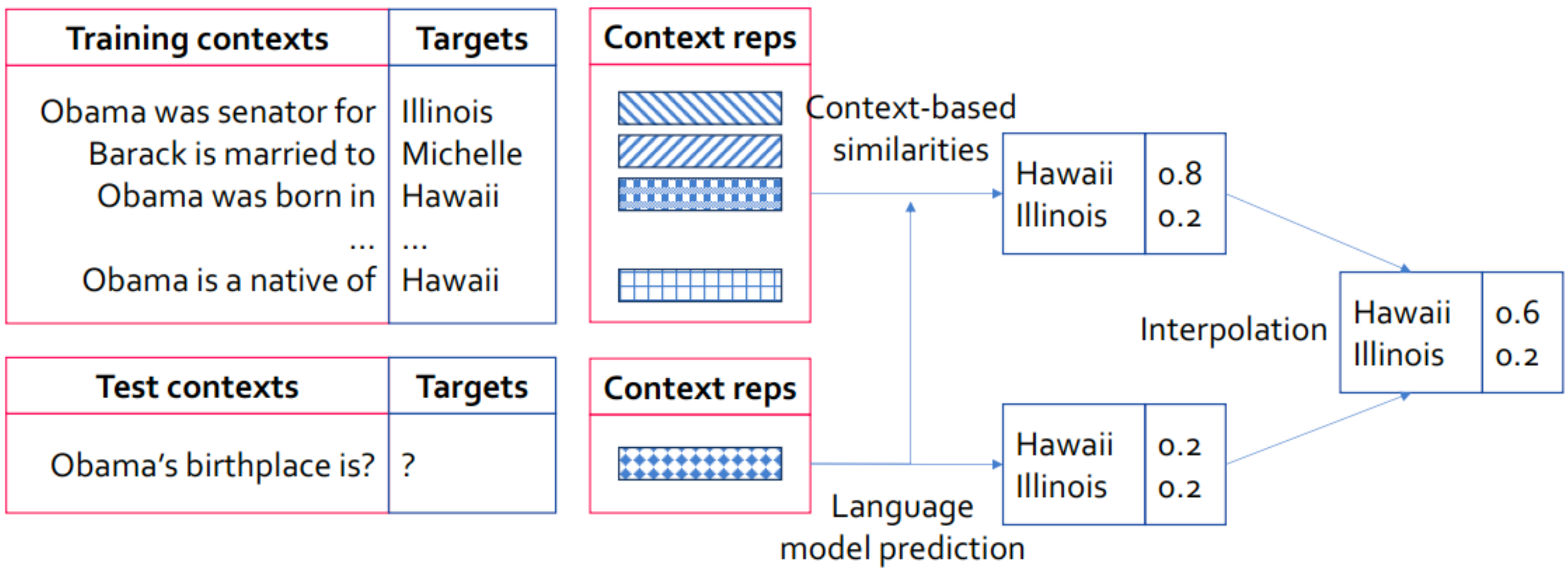


Retrieved Results Integration: Output-layer integration



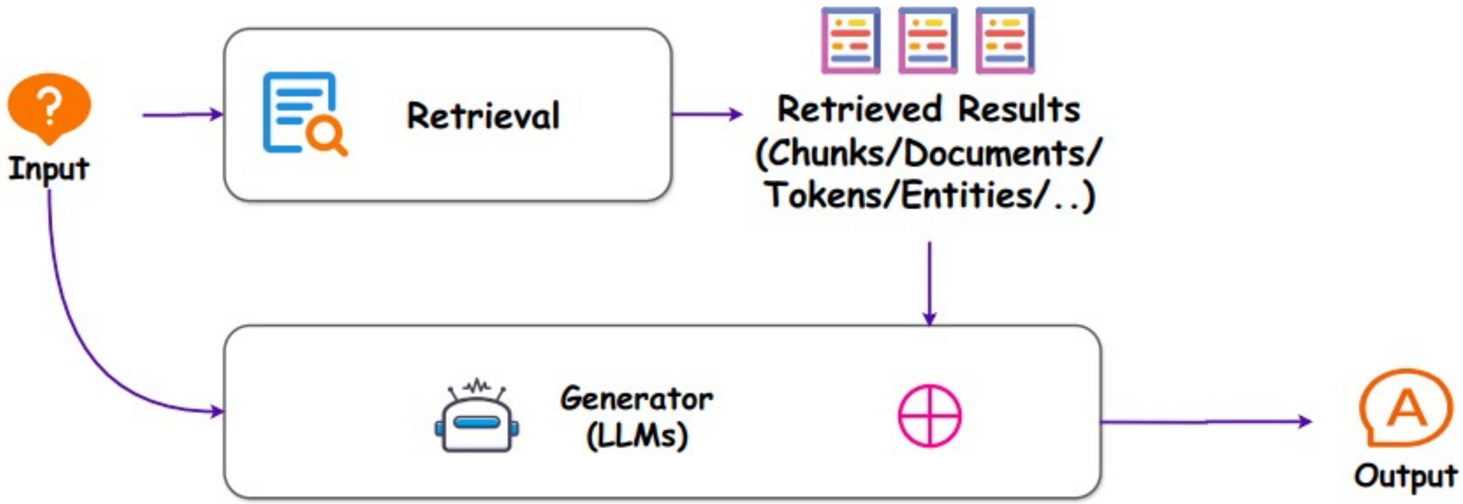
RA-LLM Architecture: Output-layer Integration

- **kNN-LM**: Combining retrieved probabilities and predicted ones in generation



Khandelwal el al. 2019. "Generalization through Memorization: Nearest Neighbor Language Models"

Retrieved Results Integration: Intermediate-layer Integration

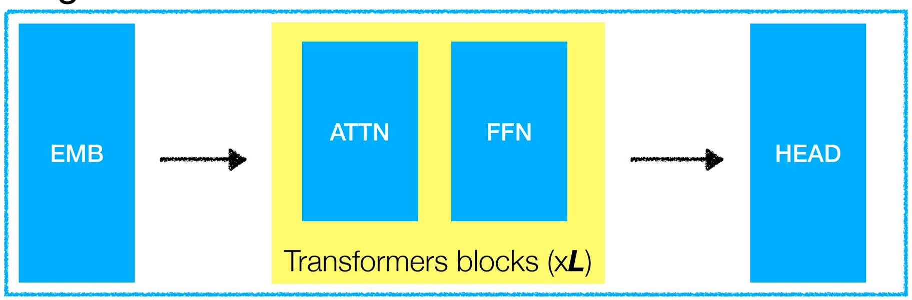


Borgeaud et al. 2022. Improving Language Models by Retrieving from Trillions of Tokens

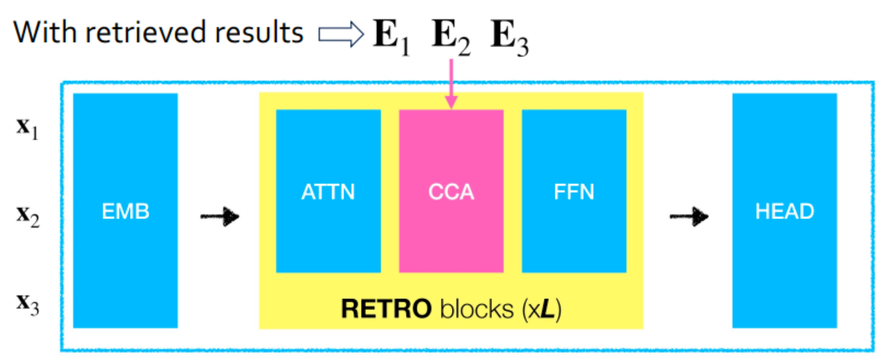
RAG meet LLMS: Towards Retrieval-Augmented LLMS Tutorial @ KDD 24 - https://advanced-recommender-systems.github.io/RAG-Meets-LLMs/

Retrieved Results Integration: Intermediate-layer Integration

Regular Decoder



Decoder to incorporate retrieved results (RETRO)

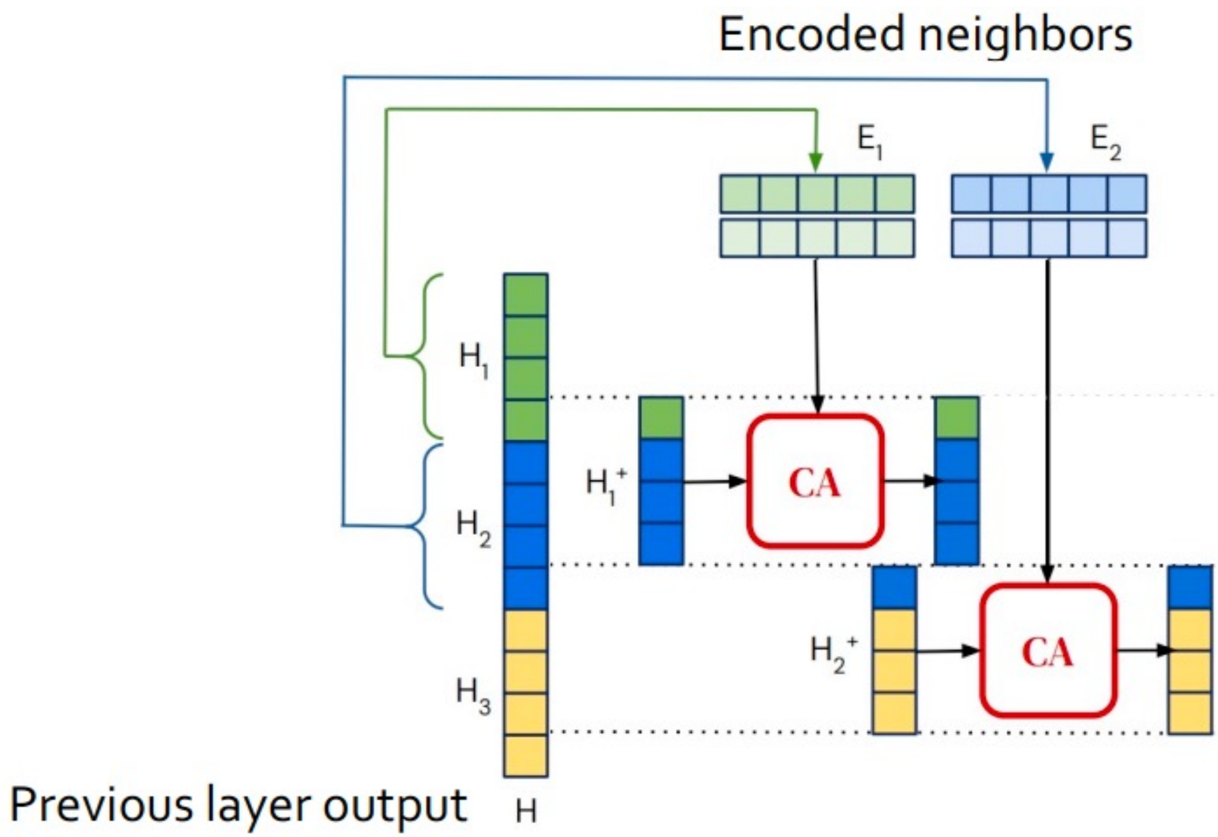


Chunked Cross Attention (CCA)

Borgeaud et al. 2022. Improving Language Models by Retrieving from Trillions of Tokens

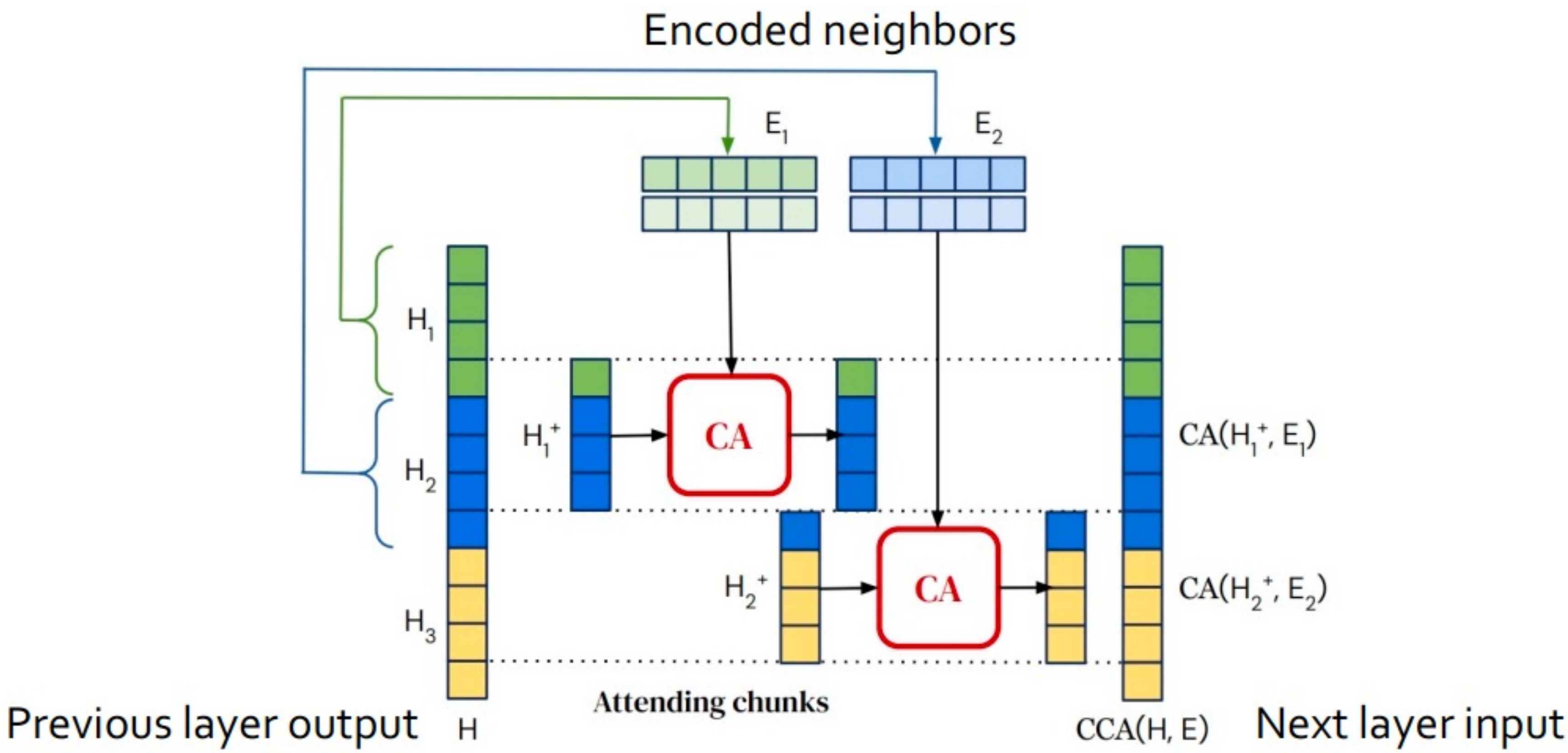
RAG meet LLMS: Towards Retrieval-Augmented LLMS Tutorial @ KDD 24 - https://advanced-recommender-systems.github.io/RAG-Meets-LLMs/

Retrieved Results Integration: Intermediate-layer Integration



RAG meet LLMS: Towards Retrieval-Augmented LLMS Tutorial @ KDD 24 - https://advanced-recommender-systems.github.io/RAG-Meets-LLMs/

Retrieved Results Integration: Intermediate-layer Integration

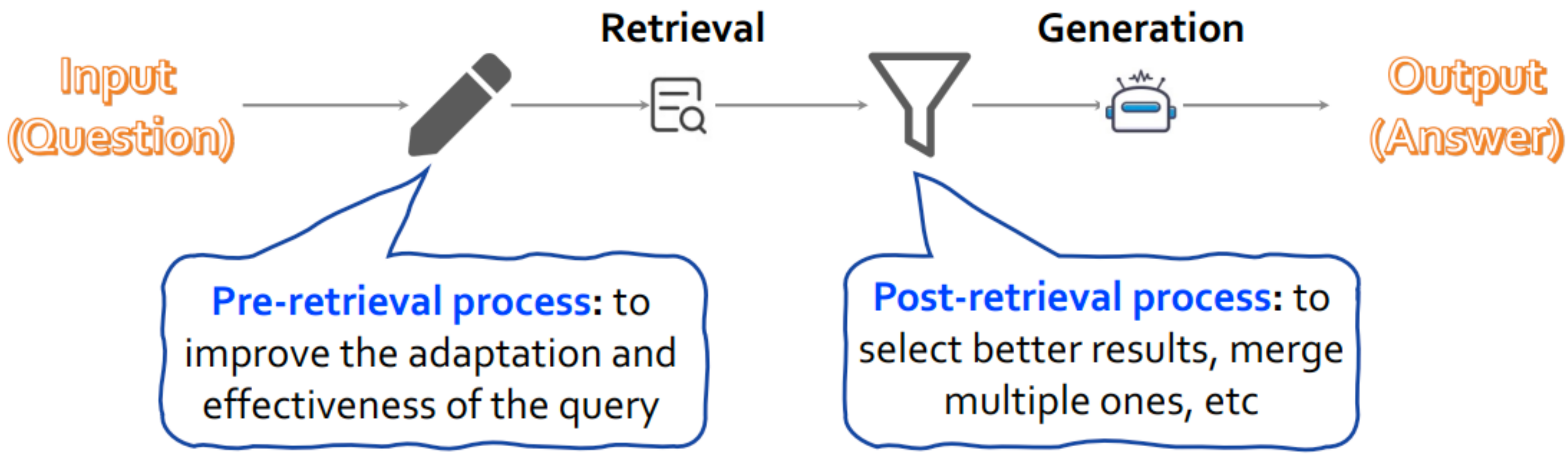


RAG meet LLMS: Towards Retrieval-Augmented LLMS Tutorial @ KDD 24 - https://advanced-recommender-systems.github.io/RAG-Meets-LLMs/

Plan for this lecture

1. RA-LLM architecture overview
2. Retriever in RA-LLMs
3. Retrieval results integration
4. Pre/Post-retrieval techniques

Pre/Post-Retrieval Techniques



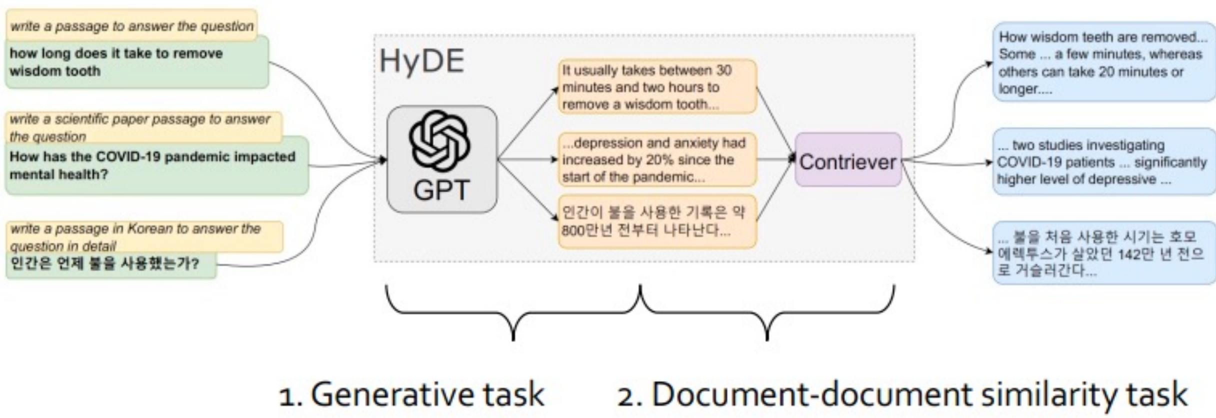
- Query rewriting
- Query decomposition
- Query expansion

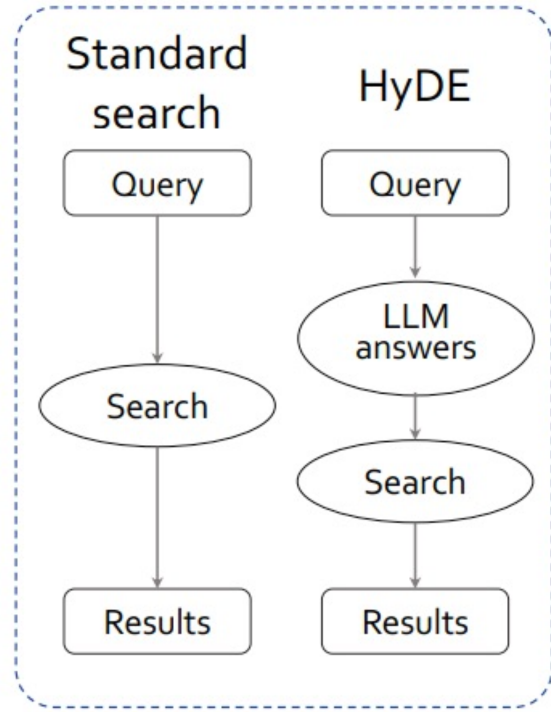
- Re-ranking
- Compression
- Correction

RAG meet LLMS: Towards Retrieval-Augmented LLMS Tutorial @ KDD 24 - https://advanced-recommender-systems.github.io/RAG-Meets-LLMs/

Pre-Retrieval Techniques

- **Query Rewriting**: to improve the adaptation of the query



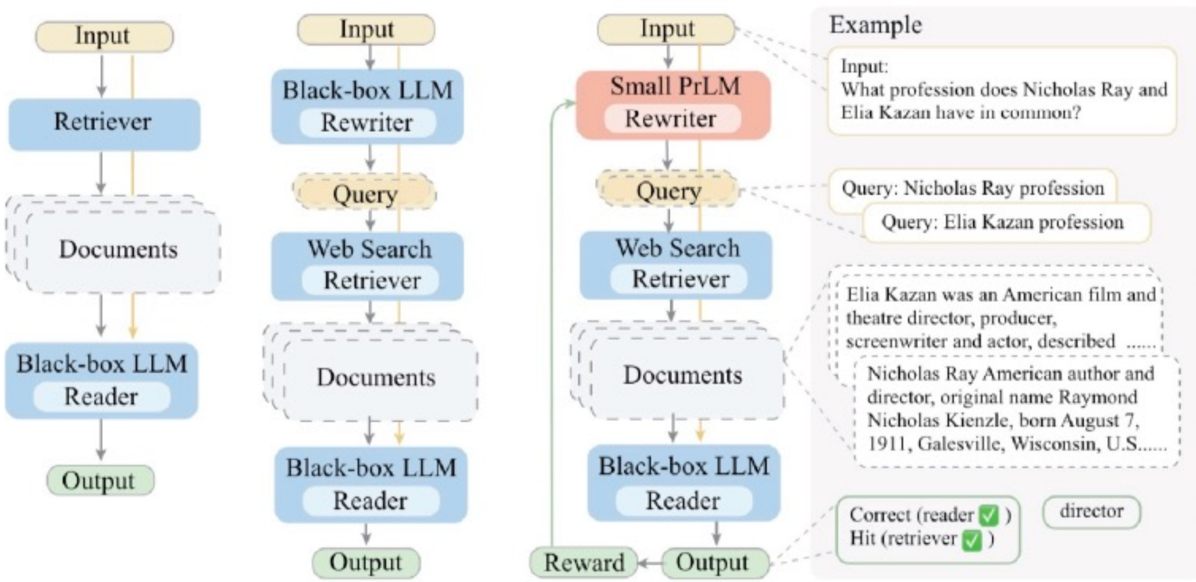


Gao et al. 2022. “Precise zero-shot dense retrieval without relevance labels”

RAG meet LLMS: Towards Retrieval-Augmented LLMS Tutorial @ KDD 24 - https://advanced-recommender-systems.github.io/RAG-Meets-LLMs/

Pre-Retrieval Techniques

- **HyDE**: Hypothetical Document Embeddings



Model	EM	F_1
HotpotQA		
Direct	32.36	43.05
Retrieve-then-read	30.47	41.34
LLM rewriter	32.80	43.85
Trainable rewriter	34.38	45.97
AmbigNQ		
Direct	42.10	53.05
Retrieve-then-read	45.80	58.50
LLM rewriter	46.40	58.74
Trainable rewriter	47.80	60.71
PopQA		
Direct	41.94	44.61
Retrieve-then-read	43.20	47.53
LLM rewriter	46.00	49.74
Trainable rewriter	45.72	49.51

Works on different QA settings

Wang et al. 2023. “Query rewriting for retrieval-augmented large language models”

RAG meet LLMS: Towards Retrieval-Augmented LLMS Tutorial @ KDD 24 - https://advanced-recommender-systems.github.io/RAG-Meets-LLMs/

Pre-Retrieval Techniques

- **Query Expansion**

LLM Prompts

Write a passage that answers the given query:

Query: what state is this zip code 85282
Passage: Welcome to TEMPE, AZ 85282. 85282 is a rural zip code in Tempe, Arizona. The population is primarily white...
...
Query: when was pokemon green released
Passage:

Method	Fine-tuning	MS MARCO dev MRR@10	MS MARCO dev R@50	MS MARCO dev R@1k	TREC DL 19 nDCG@10
Sparse retrieval					
BM25	✗	18.4	58.5	85.7	51.2*
+ query2doc	✗	$21.4^{+3.0}$	$65.3^{+6.8}$	$91.8^{+6.1}$	$\mathbf{66.2}^{+15.0}$
BM25 + RM3	✗	15.8	56.7	86.4	52.2
docT5query (Nogueira and Lin)	✓	**27.7**	**75.6**	**94.7**	64.2
Dense retrieval w/o distillation					
ANCE (Xiong et al., 2021)	✓	33.0	-	95.9	64.5
HyDE (Gao et al., 2022)	✗	-	-	-	61.3
$\text{DPR}_{\text{bert-base}}$ (our impl.)	✓	33.7	80.5	95.9	64.7
+ query2doc	✓	$\mathbf{35.1}^{+1.4}$	$\mathbf{82.6}^{+2.1}$	$\mathbf{97.2}^{+1.3}$	$\mathbf{68.7}^{+4.0}$

New query = original query + generated documents

$$q^{+} = \text{concat}(q,\ [\text{SEP}],\ d')$$

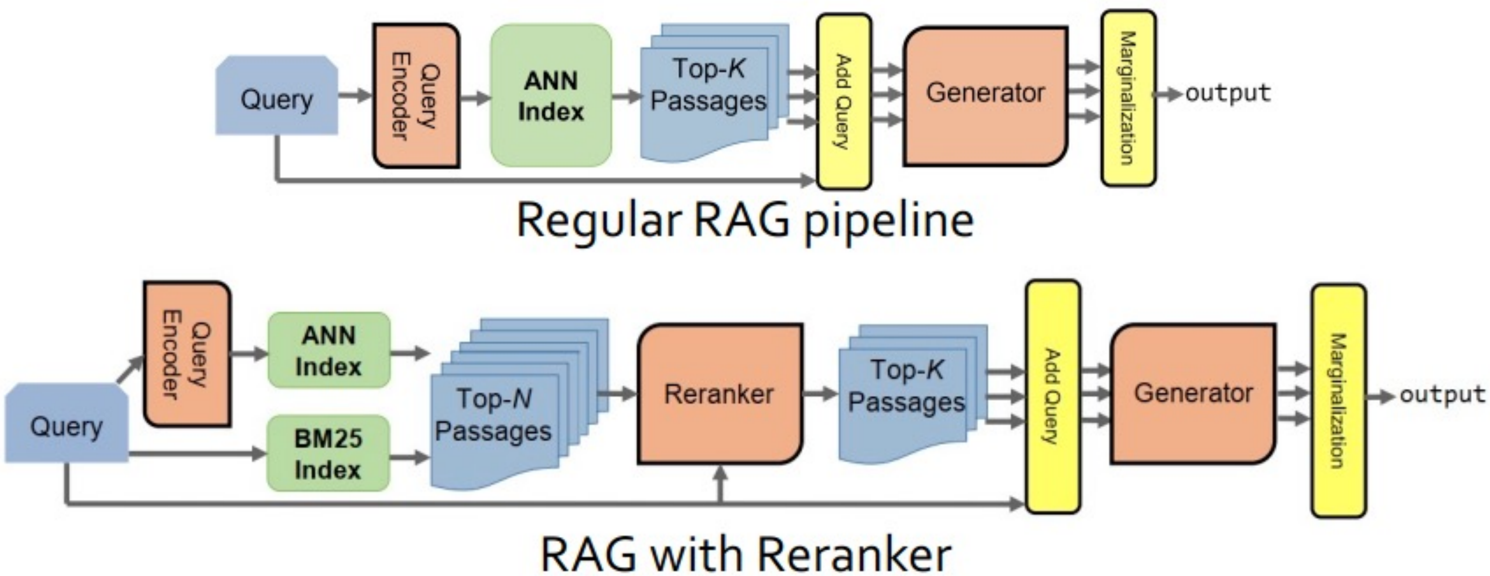
Works for both sparse and dense retrievers

Wang et al. 2023. "Query2doc: Query Expansion with Large Language Models"

RAG meet LLMS: Towards Retrieval-Augmented LLMS Tutorial @ KDD 24 - https://advanced-recommender-systems.github.io/RAG-Meets-LLMs/

Post-Retrieval Techniques

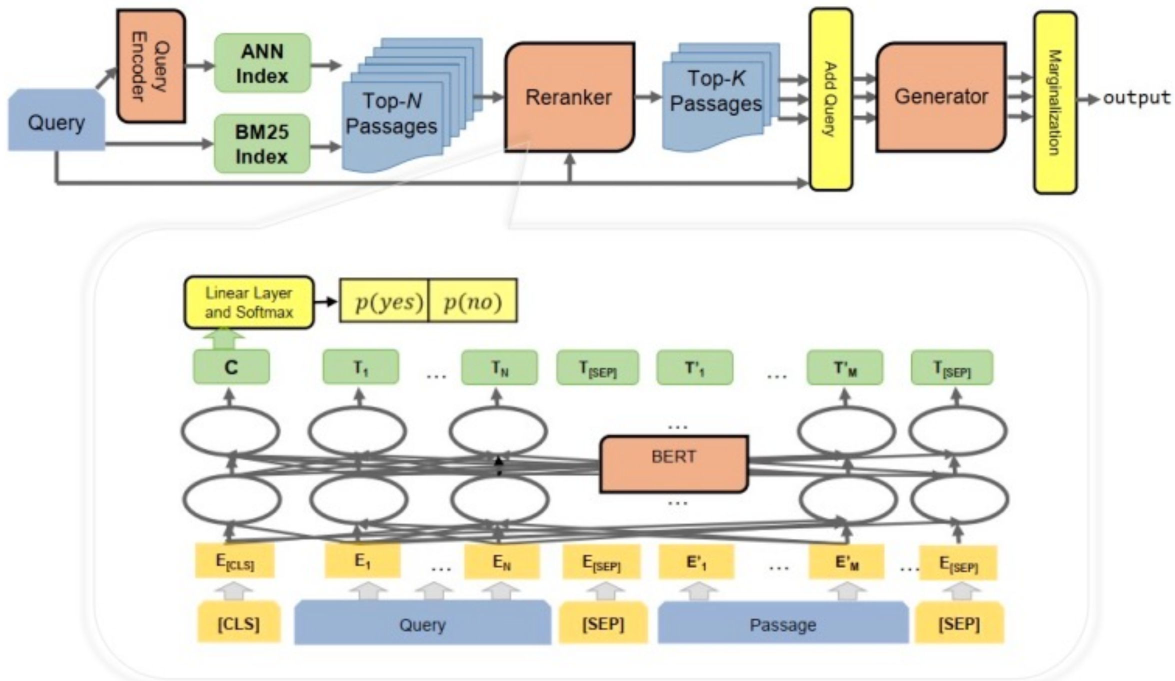
- **Retrieved Result Rerank (Re2G)**
  Results from initial retrieval can be greatly improved through the use of a reranker
  Reranker allows merging retrieval results from sources with incomparable scores, e.g., BM25 and neural initial retrieval



Glass et al. 2022. "Re2G: Retrieve, Rerank, Generate"

Retrieved Result Rerank (Re2G) Model

- **Reranker**: interaction model based on the sequence-pair classification



Nogueira and Cho, 2019, Passage Re-ranking with BERT

Retrieved Result Rerank (Re2G) Performance

	T-REx		NQ		TriviaQA		FEVER		WoW	
	R-Prec	**R@5**	**R-Prec**	**R@5**	**R-Prec**	**R@5**	**R-Prec**	**R@5**	**R-Prec**	**R@5**
BM25	46.88	69.59	24.99	42.57	26.48	45.57	42.73	70.48	27.44	45.74
DPR Stage 1	49.02	63.34	56.64	64.38	60.12	64.04	75.49	84.66	34.74	60.22
KGI_0 DPR	65.02	75.52	64.65	69.60	60.55	63.65	80.34	86.53	**48.04**	**71.02**
Re^2G DPR	**67.16**	**76.42**	**65.88**	**70.90**	**62.33**	**65.72**	**84.13**	**87.90**	47.09	69.88
KGI_0 DPR+BM25	60.48	80.06	36.91	66.94	40.81	64.79	65.95	90.34	35.63	68.47
Reranker Stage 1	81.22	87.00	70.78	73.05	**71.80**	**71.98**	87.71	92.43	55.50	**74.98**
Re^2G Reranker	**81.24**	**88.58**	**70.92**	**74.79**	60.37	70.61	**90.06**	**92.91**	**57.89**	74.62

Significantly outperforms pipelines without the *Rerank* stage

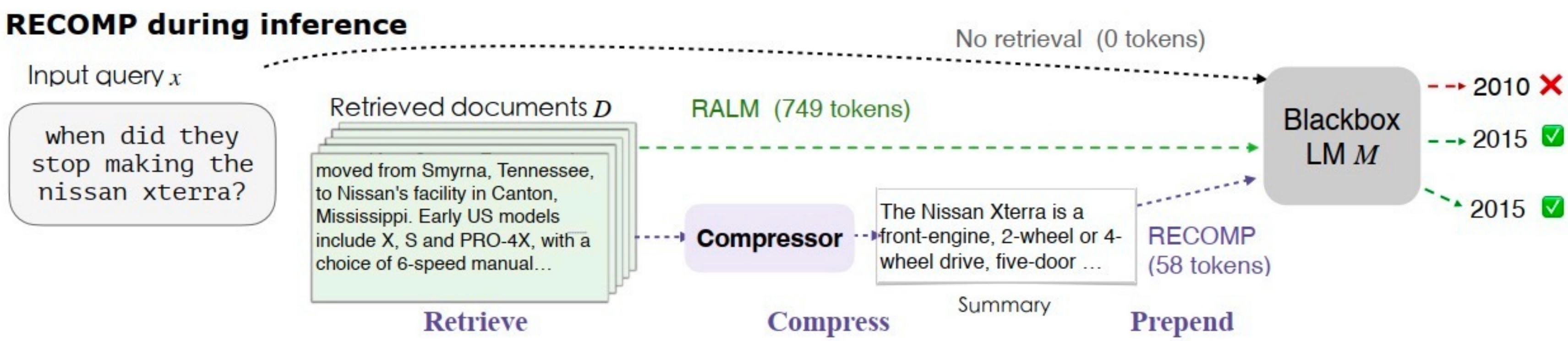
Glass et al. 2022. “Re2G: Retrieve, Rerank, Generate”

RAG meet LLMS: Towards Retrieval-Augmented LLMS Tutorial @ KDD 24 - https://advanced-recommender-systems.github.io/RAG-Meets-LLMs/

Post-Retrieval Techniques

Retrieved Result Compression

- To reduce the computational costs and also relieve the burden of LMs to identify relevant information in long retrieved documents.



Compressor Learning Objectives

- Concise
- Effective
- Faithful

Xu et al. 2023. "RECOMP: Improving retrieval- augmented LMs with context compression and selective augmentation"

Post-Retrieval Techniques

QA tasks

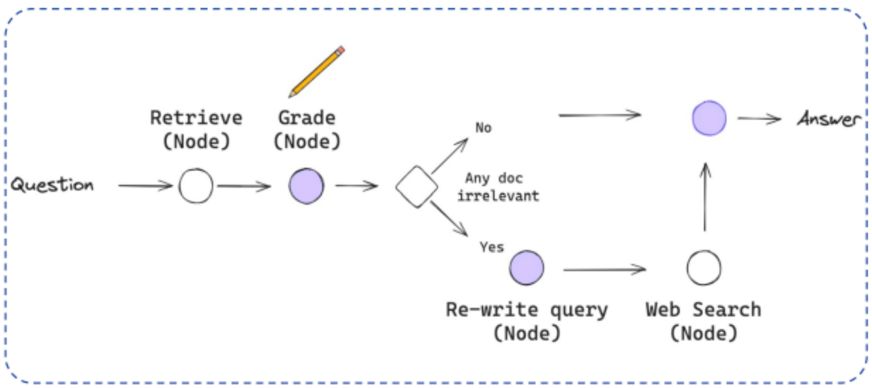
In-Context evidence	# tok	NQ EM	F1	# tok	TQA EM	F1	# tok	HotpotQA EM	F1
-	0	21.99	29.38	0	49.33	54.85	0	17.80	26.10
RALM without compression									
Top 1 documents	132	33.07	41.45	136	57.84	64.94	138	28.80	40.58
Top 5 documents	660	**39.39**	**48.28**	677	**62.37**	**70.09**	684	**32.80**	**43.90**
Phrase/token level compression									
Top 5 documents (NE)	338	23.60	31.02	128	54.96	61.19	157	22.20	31.89
Top 5 documents (BoW)	450	28.48	36.84	259	58.16	65.15	255	25.60	36.00
Extractive compression of top 5 documents									
Oracle	34	60.22	64.25	32	79.29	82.06	70	41.80	51.07
Random	32	23.27	31.09	31	50.18	56.24	61	21.00	29.86
BM25	36	25.82	33.63	37	54.67	61.19	74	26.80	38.02
DPR	39	34.32	43.38	41	56.58	62.96	78	27.40	38.15
Contriever	36	30.06	31.92	40	53.67	60.01	78	28.60	39.48
Ours	37	36.57	44.22	38	**58.99**	65.26	75	**30.40**	**40.14**

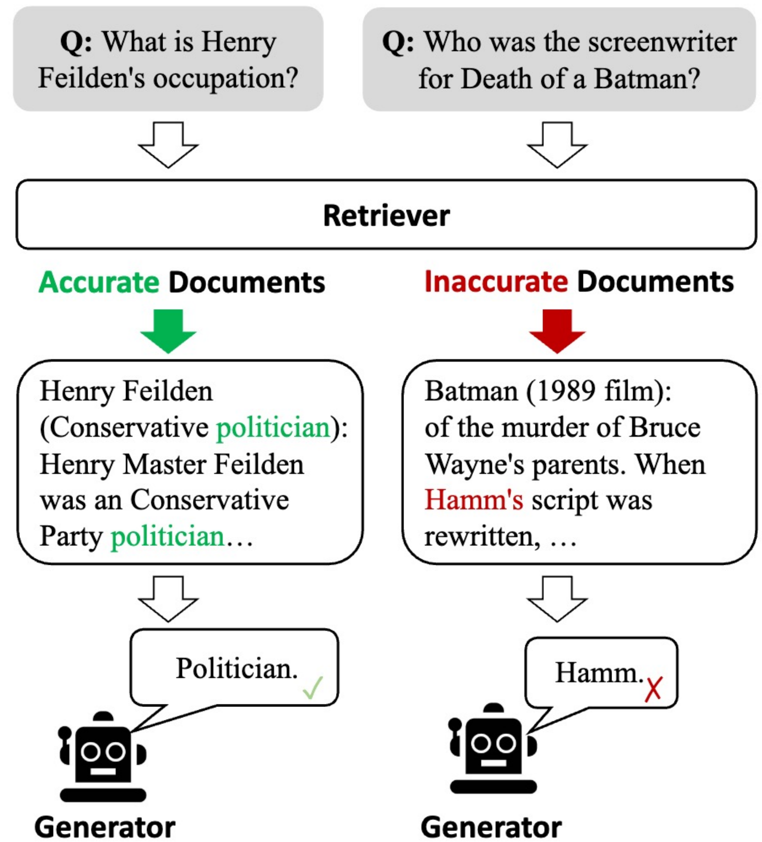
Outperforms representative sparse and dense retrievers

Xu et al. 2023. "RECOMP: Improving retrieval- augmented LMs with context compression and selective augmentation"

Post-Retrieval Techniques: Corrective RAG

Grading and correcting

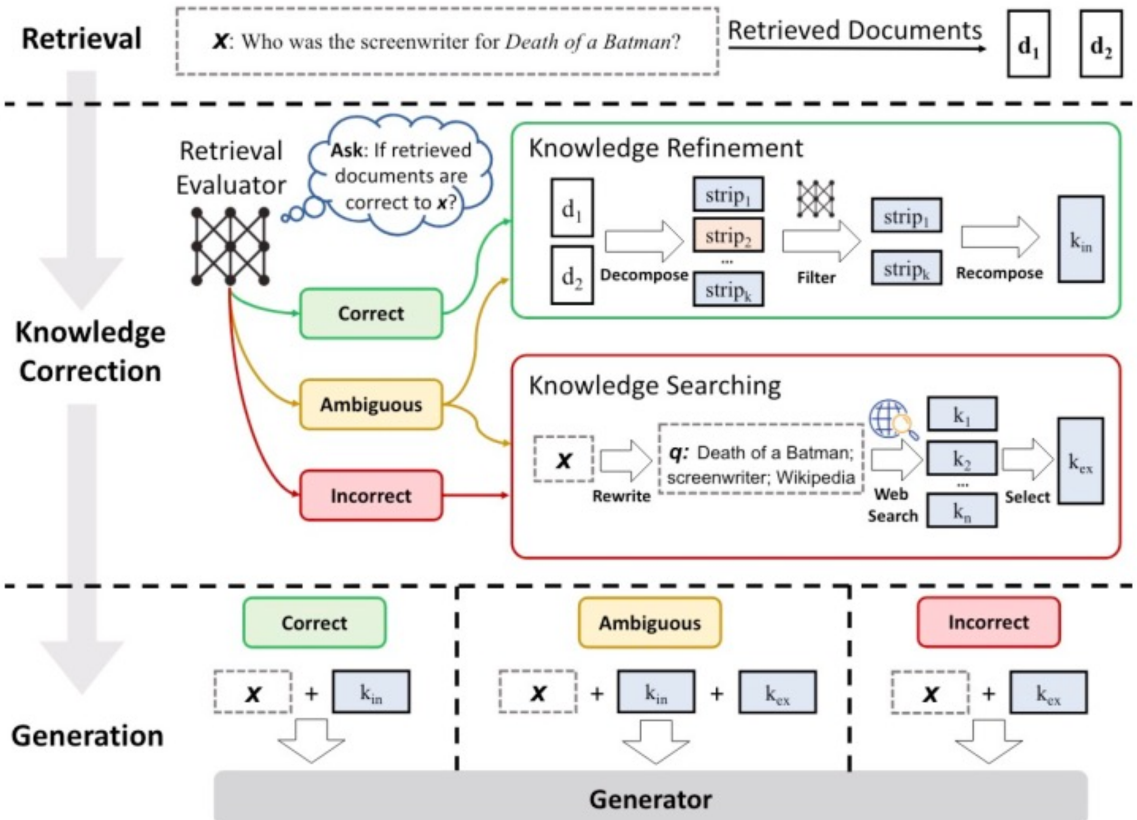




Yan et al., 2024, Corrective Retrieval Augmented Generation

RAG meet LLMS: Towards Retrieval-Augmented LLMS Tutorial @ KDD 24 - https://advanced-recommender-systems.github.io/RAG-Meets-LLMs/

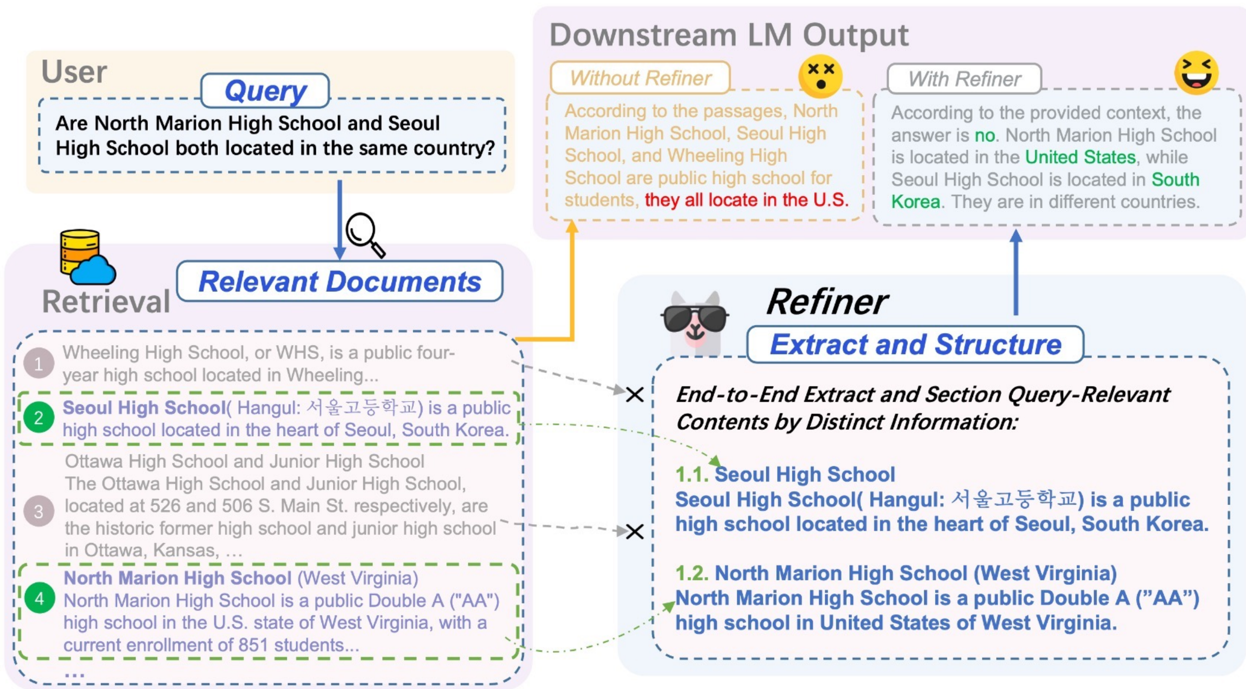
Post-Retrieval Techniques: Corrective RAG



RAG meet LLMS: Towards Retrieval-Augmented LLMS Tutorial @ KDD 24 - https://advanced-recommender-systems.github.io/RAG-Meets-LLMs/

Post-Retrieval Techniques: Refiner

Refiner: leveraging a single decoder-only LLM to adaptively extract query relevant contents verbatim along with the necessarv context



Li et al., 2024, Refiner: Restructure Retrieval Content Efficiently to Advance Question-Answering Capabilities

Plan for this lecture

1. Introduction of Retrieval Augmented Large Language Models (RA LLMs)

2. Architecture of RA-LLMs and Main Modules

3. Learning Approach of RA-LLMs

4. Challenges and Future Directions of RA-LLMs

Learning Approach of RA-LLMs

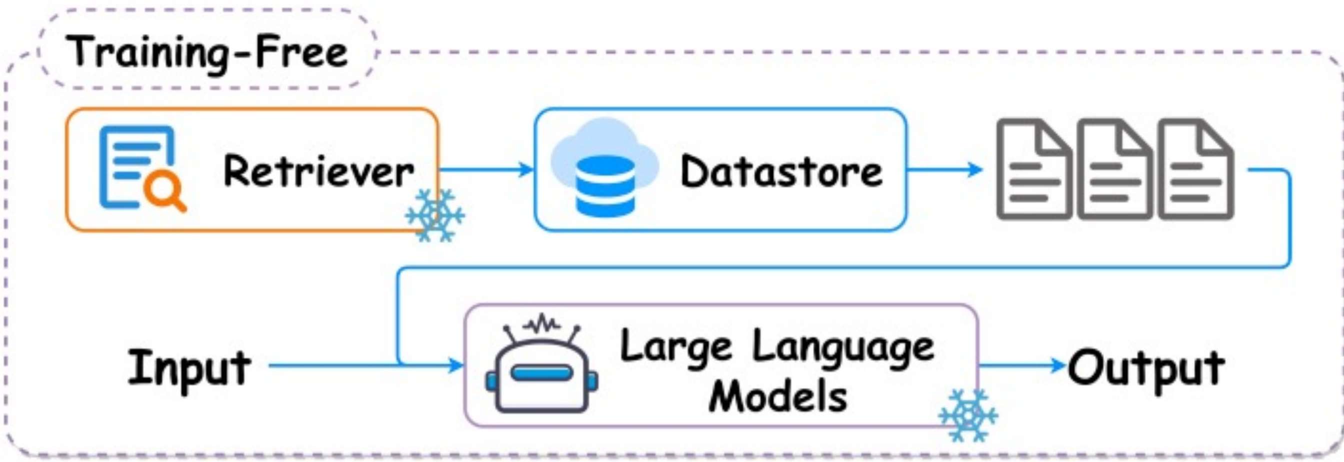
Training-free Methods

Training-based Methods

- Independent Learning
- Sequential Learning
- Joint Learning

RA-LLM Learning: Training-free

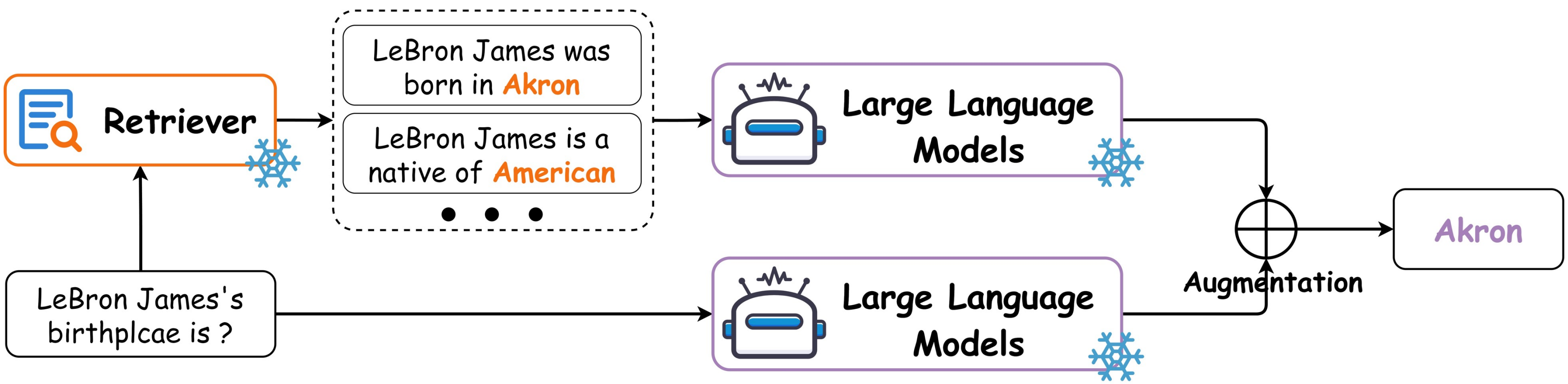
Retrieval models and language models are both frozen.



RA-LLM Learning: Training-free

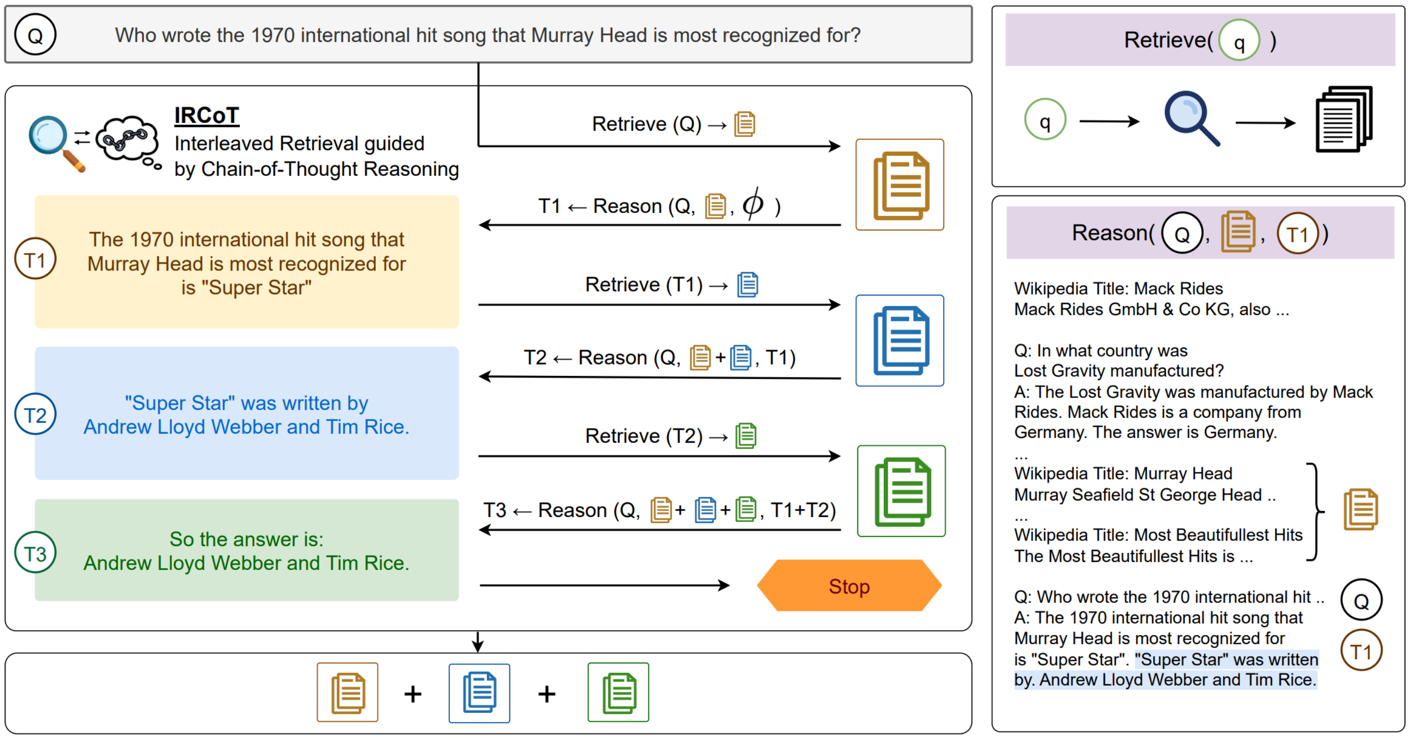
Prompt Engineering-based Methods

Retrieval-Guided Token Generation Methods



RA-LLM Learning: Training-free

IRCoT

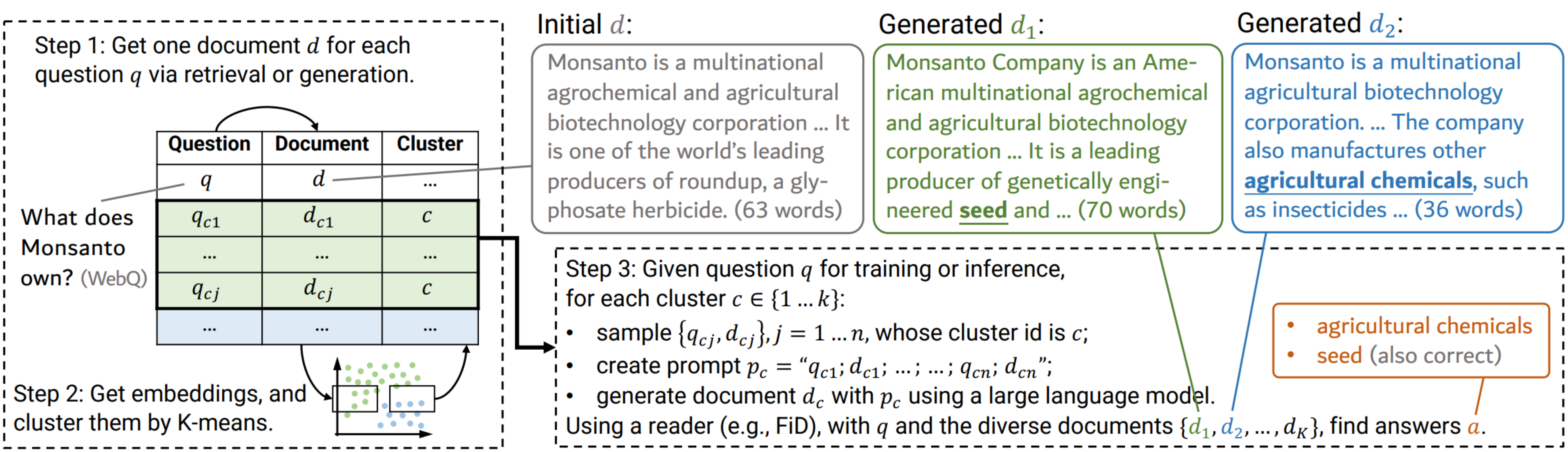


Trivedi, Harsh, et al. "Interleaving Retrieval with Chain-of-Thought Reasoning for Knowledge-Intensive Multi-Step Questions." ACL. 2023

RAG meet LLMS: Towards Retrieval-Augmented LLMS Tutorial @ KDD 24 - https://advanced-recommender-systems.github.io/RAG-Meets-LLMs/

RA-LLM Learning: Training-free

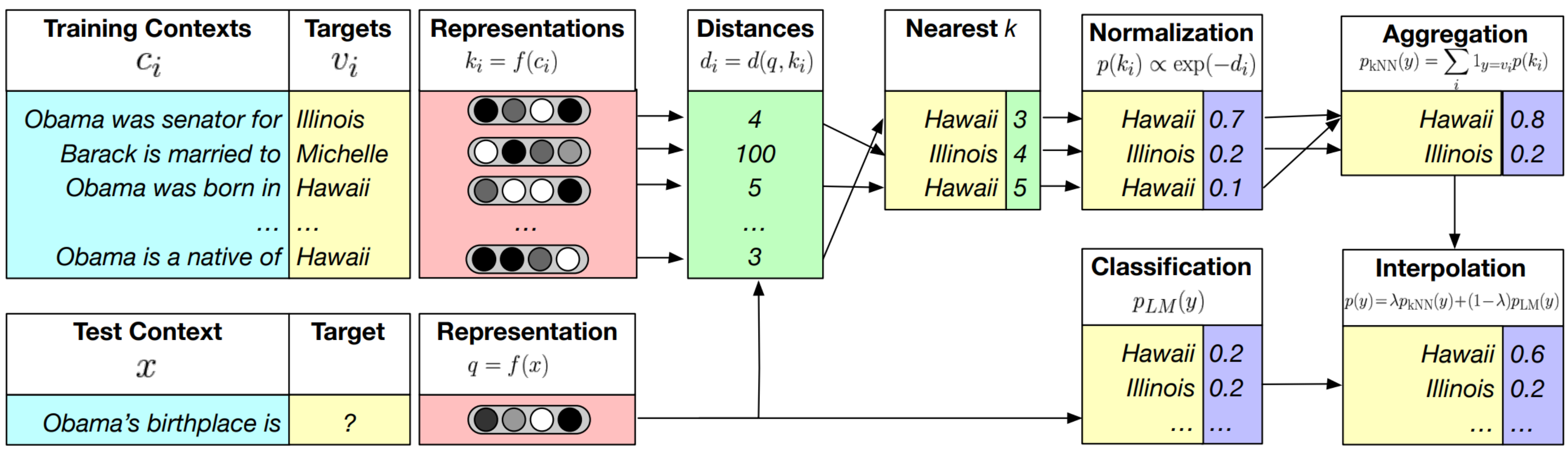
GENREAD



Yu, Wenhao, et al. "Generate rather than Retrieve: Large Language Models are Strong Context Generators." International Conference on Learning Representations. 2023

RA-LLM Learning: Training-free

kNN-LM

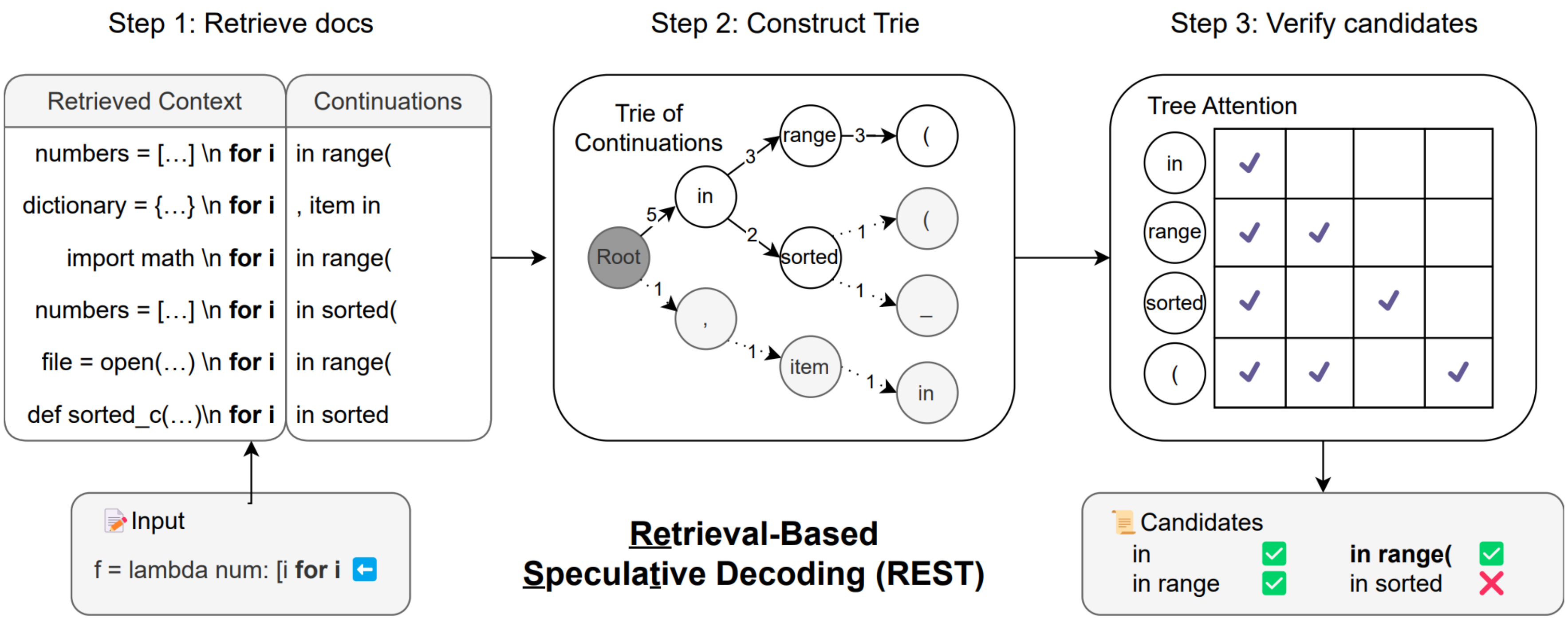


$$p(y|x) = \lambda \, p_{\text{kNN}}(y|x) + (1 - \lambda) \, p_{\text{LM}}(y|x)$$

Khandelwal, Urvashi, et al. "Generalization through Memorization: Nearest Neighbor Language Models." International Conference on Learning Representations. 2019.

RA-LLM Learning: Training-free

REST



He, Zhenyu, et al. "REST: Retrieval-Based Speculative Decoding." NAACL. 2024

RA-LLM Learning: Training-free

✓ Work with off-the-shelf models

x All components are fixed and not trained

x Might not achieve optimal learning result of the whole model

Learning Approach of RA-LLMs

Training-free Methods

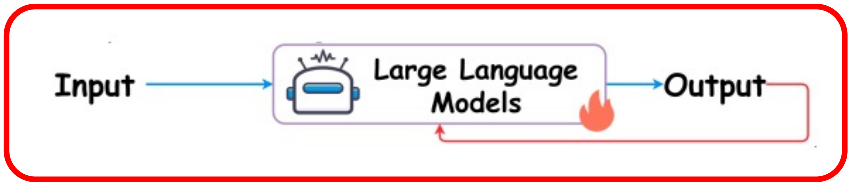
Training-based Methods

- Independent Learning
- Sequential Learning
- Joint Learning

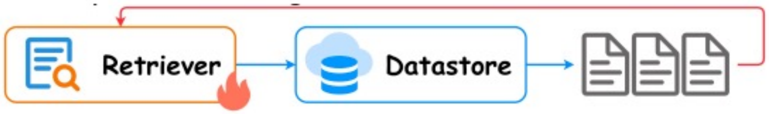
RA-LLM Learning: Independent Training

Retrieval models and language models are trained independently.

- Independent training of Retriever.

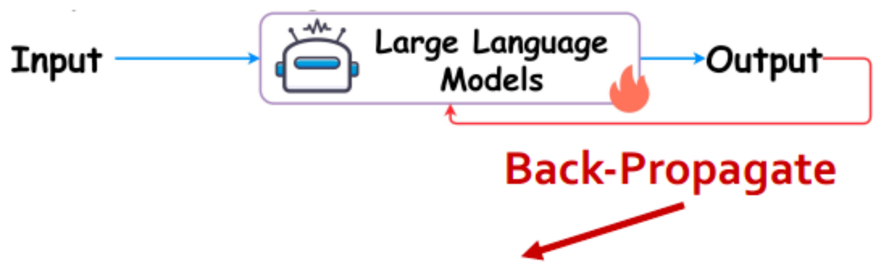


- Independent training of large language models



RA-LLM Learning: Independent Training

Independent training of large language models.



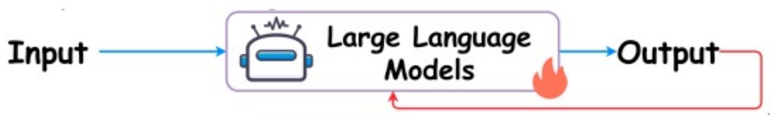
$$Minimize \; -logP_{LM}(y|x)$$

......

RA-LLM Learning: Independent Training

Retrieval models and language models are trained independently.

• Independent training of Retriever.



• Independent training of large language models

RA-LLM Learning: Independent Training

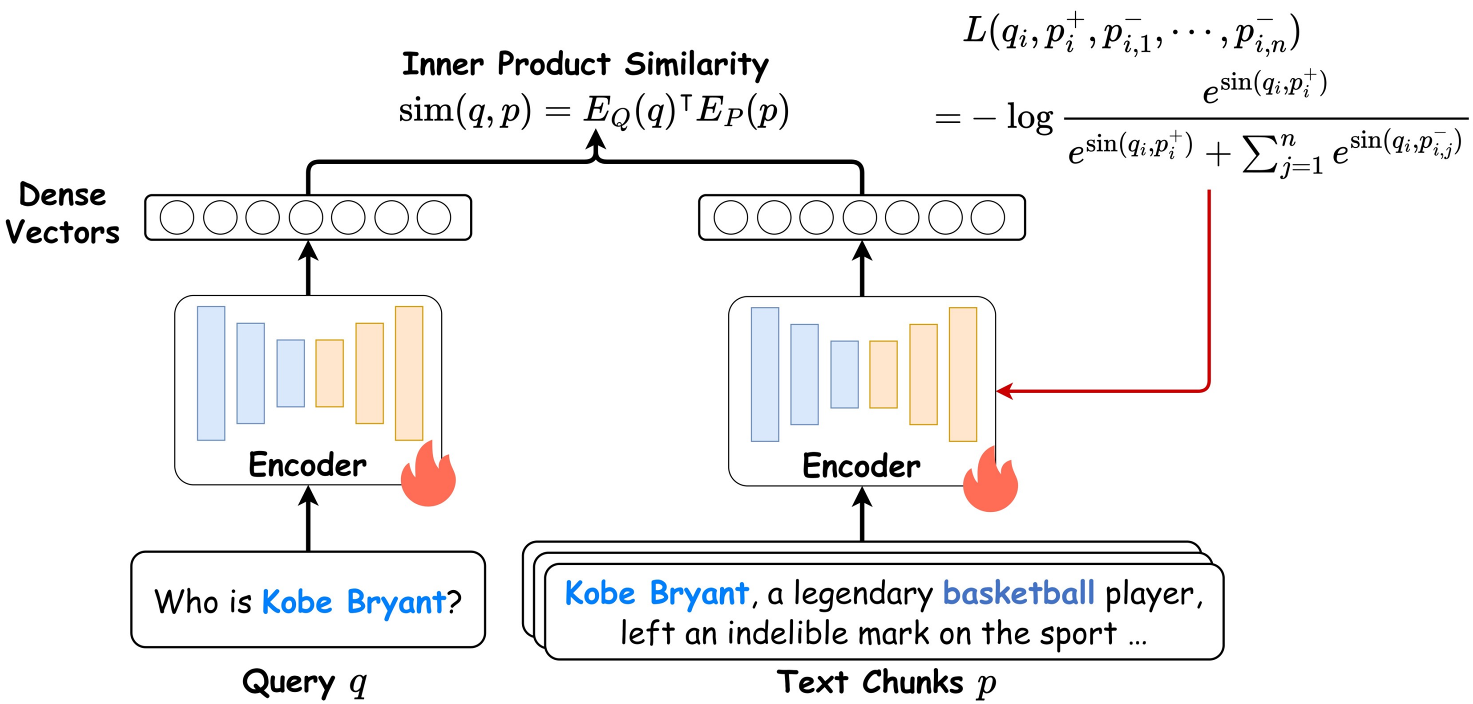
Sparse retrieval models: TF-IDF / BM25

No training is Needed!

Ramos, Juan. "Using TF-IDF to determine word relevance in document queries." Proceedings of the first instructional conference on machine learning. 2003.
Robertson, Stephen, and Hugo Zaragoza. "The probabilistic relevance framework: BM25 and beyond." Foundations and Trends® in Information Retrieval. 2009

RA-LLM Learning: Independent Training

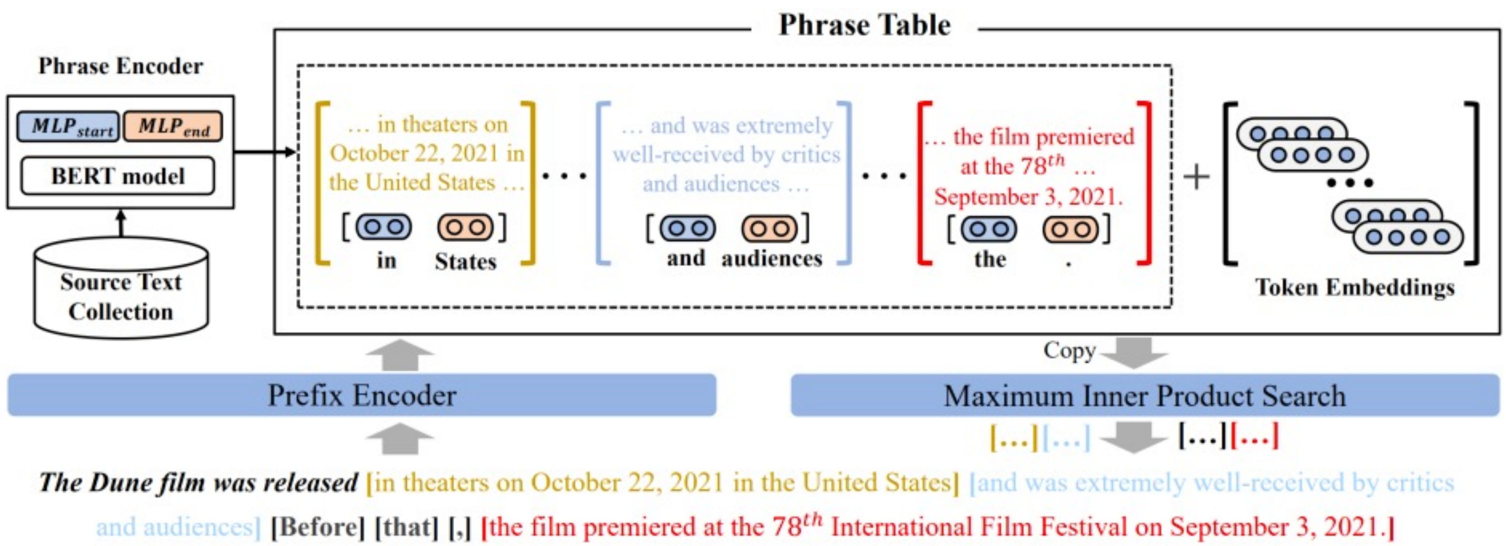
Dense retrieval models: DPR



Karpukhin, Vladimir, et al. "Dense passage retrieval for open-domain question answering." 2020 Conference on Empirical Methods in Natural Language Processing, 2020.

RA-LLM Learning: Independent Training

Dense retrieval models: CoG



$$\mathcal{H}_{i+1} = \mathrm{PrefixEncoder}(x_i, \mathcal{H}_i).$$

$$\mathcal{D}_{\texttt{start}} = \mathrm{MLP}_{\texttt{start}}(\mathcal{D}), \mathcal{D}_{\texttt{end}} = \mathrm{MLP}_{\texttt{end}}(\mathcal{D}).$$

$$\mathrm{PhraseEncoder}(s, e, D) = [\mathcal{D}_{\texttt{start}}[s]; \mathcal{D}_{\texttt{end}}[e]] \in \mathbb{R}^d$$

Tian Lan, et al. "Copy is All You Need." In The Eleventh International Conference on Learning Representations, 2022.

RA-LLM Learning: Independent Training

Model Training:

$$\mathcal{L}_p = -\frac{1}{n}\sum_{k=1}^{n}\log\frac{\exp(q_k \cdot p_k)}{\sum_{p\in\mathcal{P}_k}\exp(q_k \cdot p_p) + \sum_{w\in V}\exp(q_k \cdot v_w)}$$

$$\mathcal{L}_t = -\frac{1}{m}\sum_{i=1}^{m}\log\frac{\exp(q_i, v_{D_i})}{\sum_{w\in V}\exp(q_i, v_w)}$$

Tian Lan, et al. “Copy is All You Need.” In The Eleventh International Conference on Learning Representations, 2022.

RA-LLM Learning: Independent Training

✓ Work with off-the-shelf models, flexible

✓ Each part can be improved independently

x Lack of integrity between Retrieval and Generation

x Retrieval models are not optimized specified for the tasks/ domains/ generators

Learning Approach of RA-LLMs

Training-free Methods

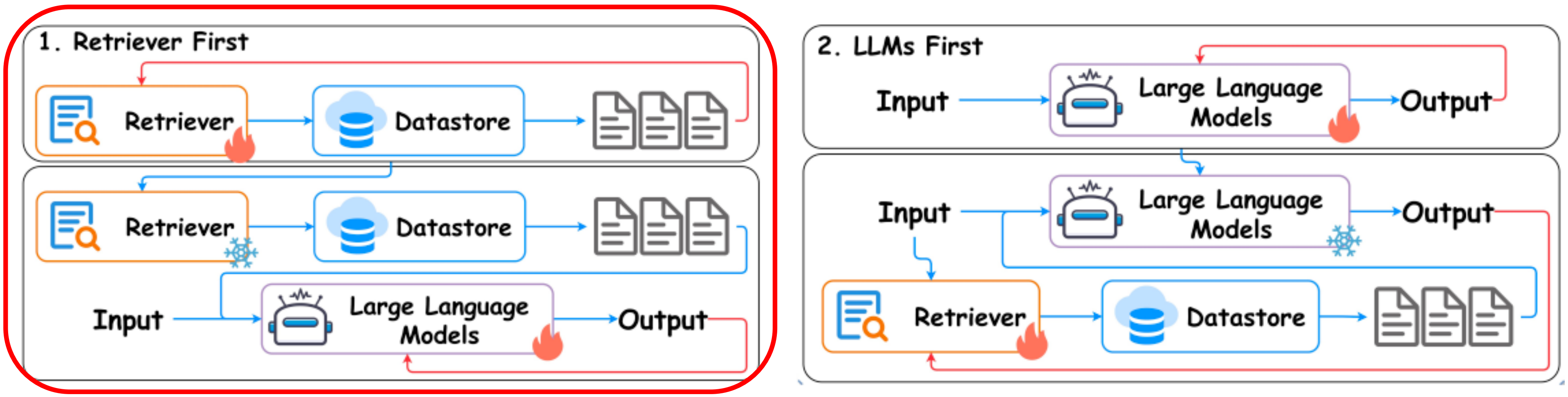
Training-based Methods

- Independent Learning
- Sequential Learning
- Joint Learning

RA-LLM Learning: Sequential Training

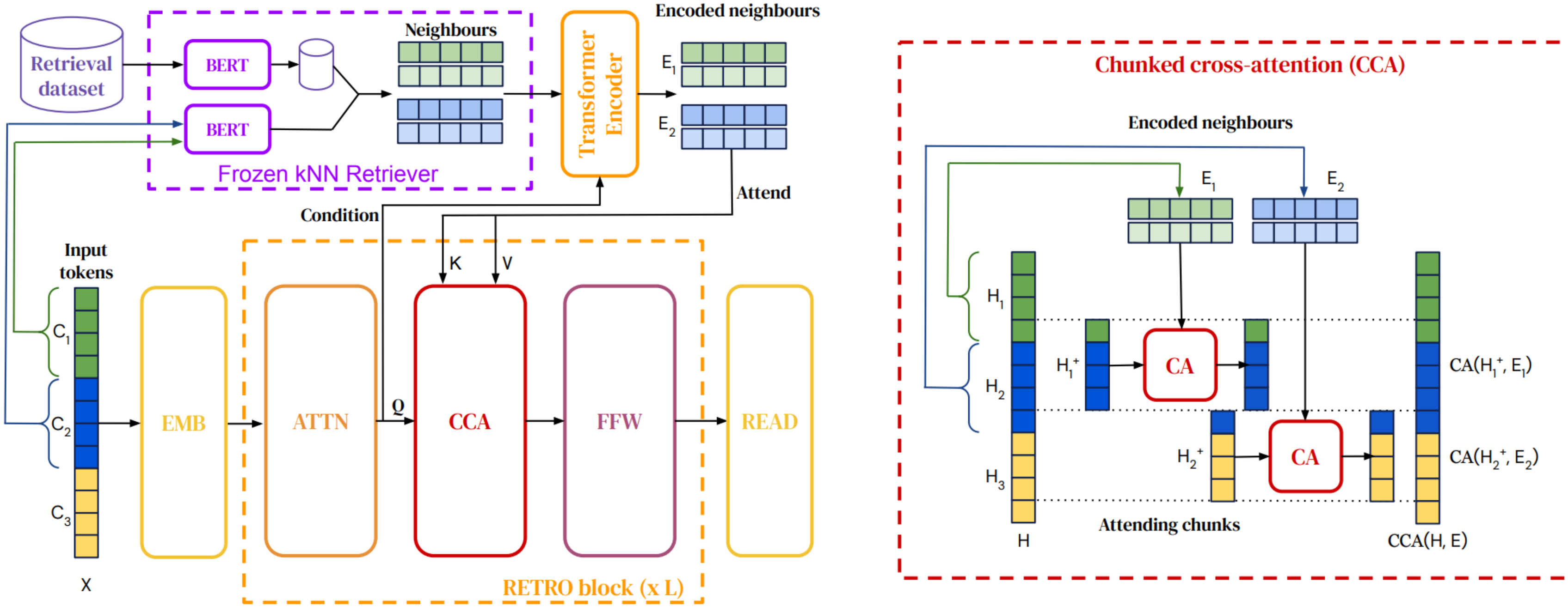
One component is first trained independently and then fixed.

The other component is trained with an objective that depends on the first one



RA-LLM Learning: Sequential Training

RETRO



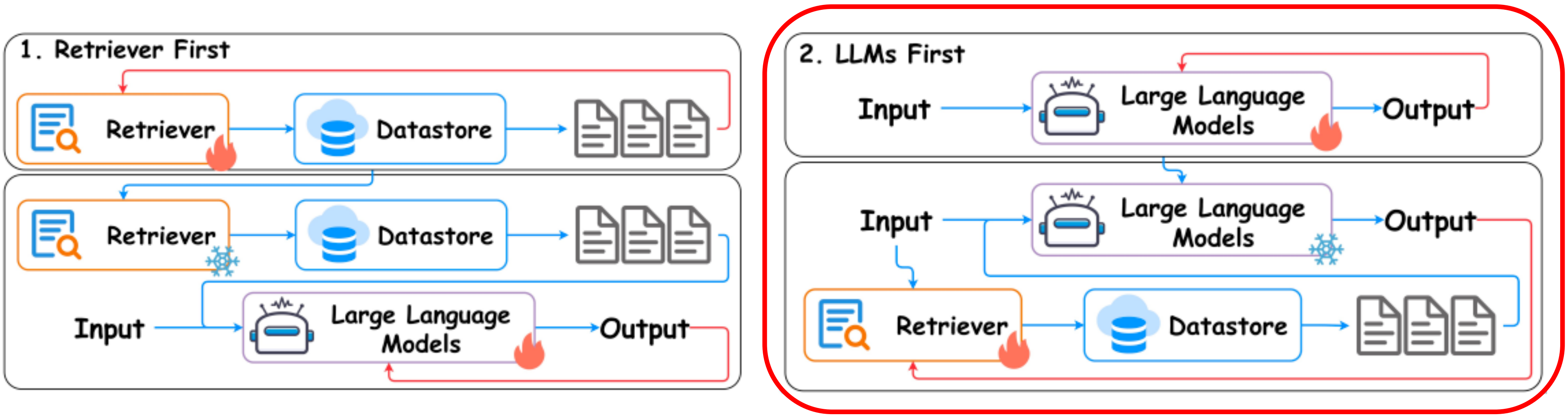
Borgeaud et al. 2022. Improving Language Models by Retrieving from Trillions of Tokens

RAG meet LLMS: Towards Retrieval-Augmented LLMS Tutorial @ KDD 24 - https://advanced-recommender-systems.github.io/RAG-Meets-LLMs/

RA-LLM Learning: Sequential Training

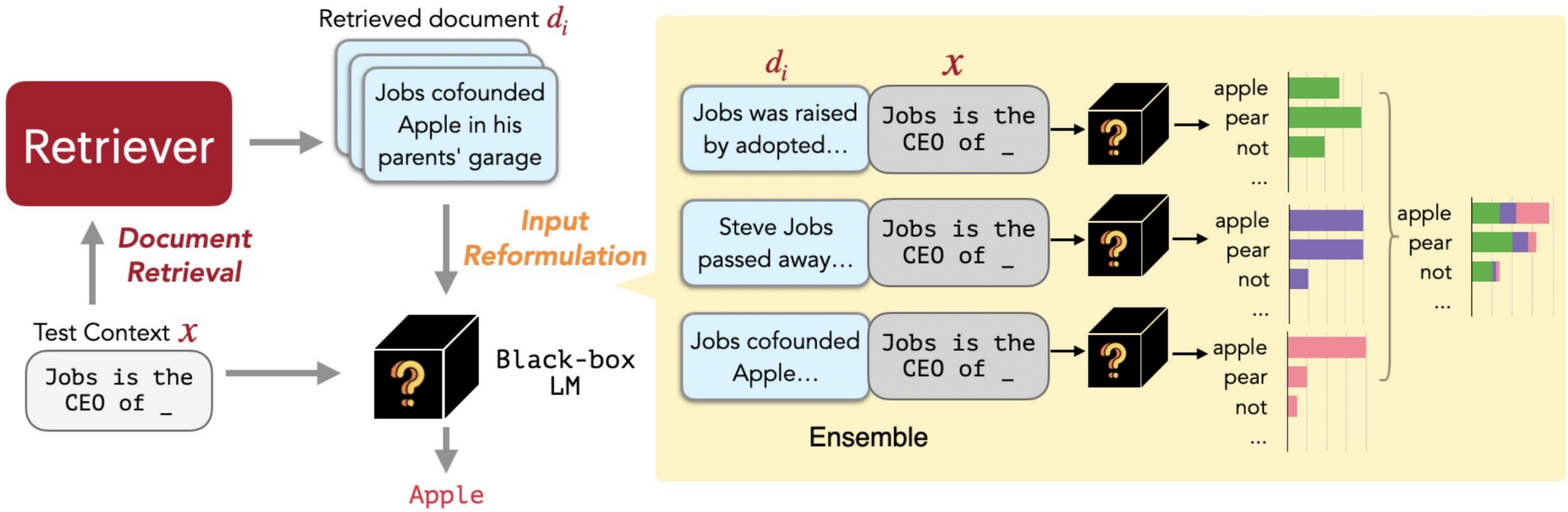
One component is first trained independently and then fixed.

The other component is trained with an objective that depends on the first one



RA-LLM Learning: Sequential Training

REPLUG (Retrieve and Plug)



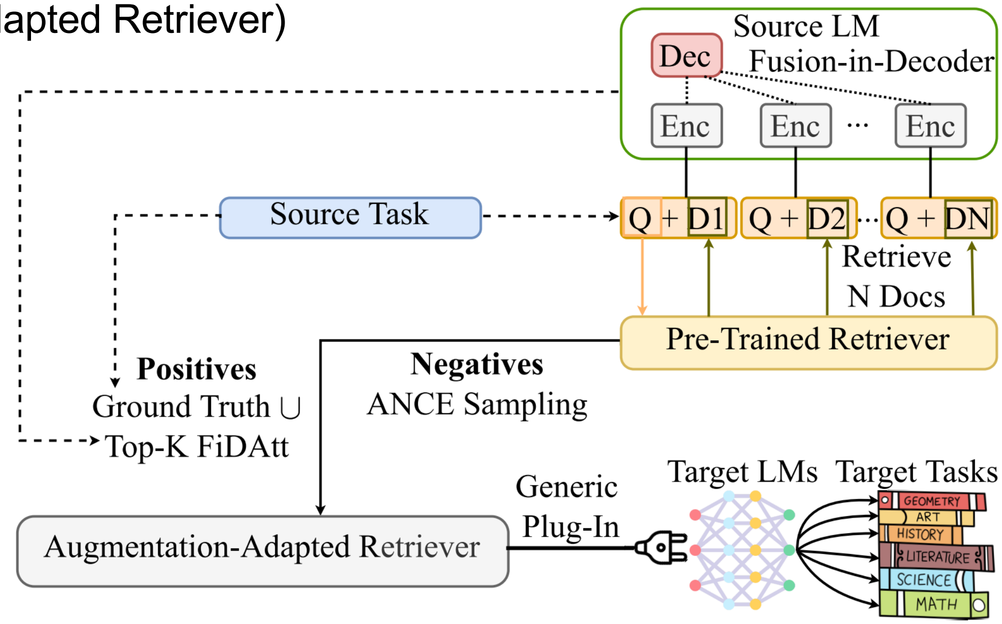
$$\mathcal{L} = \frac{1}{|\mathcal{B}|} \sum_{x \in \mathcal{B}} KL\Big(P_R(d \mid x) \parallel Q_{\mathrm{LM}}(d \mid x, y)\Big) \quad P_R(d \mid x) = \frac{e^{s(d,x)/\gamma}}{\sum_{d \in \mathcal{D}'} e^{s(d,x)/\gamma}} \quad Q(d \mid x, y) = \frac{e^{P_{LM}(y \mid d, x)/\beta}}{\sum_{d \in \mathcal{D}'} e^{P_{LM}(y \mid d, x)/\beta}}$$

Shi, Weijia, et al. "REPLUG: Retrieval-Augmented Black-Box Language Models." NAACL. 2024.

RAG meet LLMS: Towards Retrieval-Augmented LLMS Tutorial @ KDD 24 - https://advanced-recommender-systems.github.io/RAG-Meets-LLMs/

RA-LLM Learning: Sequential Training

AAR (Augmentation-Adapted Retriever)



$$\mathcal{L} = \sum_{q} \sum_{d^+ \in D^{a+}} \sum_{d^- \in D^-} l(f(\boldsymbol{q}, \boldsymbol{d}^+), f(\boldsymbol{q}, \boldsymbol{d}^-)),$$

Yu, Zichun, et al. "Augmentation-Adapted Retriever Improves Generalization of Language Models as Generic Plug-In." ACL. 2023.

RA-LLM Learning: Sequential Training

- ✓ Work with off-the-shelf models
- ✓ Generators can be trained effectively based on the retrieved results
- ✓ Retrievers can be trained to provide useful information to help the generators
- x One component is still fixed and not trained
- x Might not achieve optimal learning result of the whole modell

Learning Approach of RA-LLMs

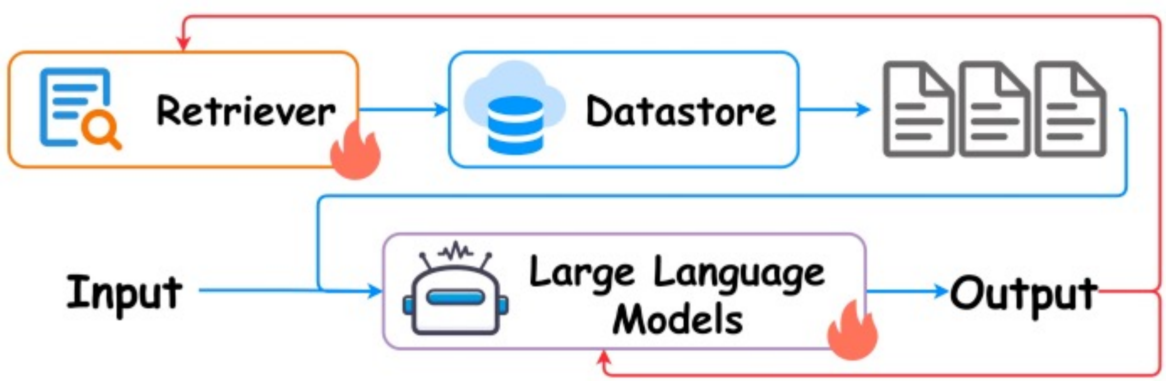
Training-free Methods

Training-based Methods

- Independent Learning
- Sequential Learning
- Joint Learning

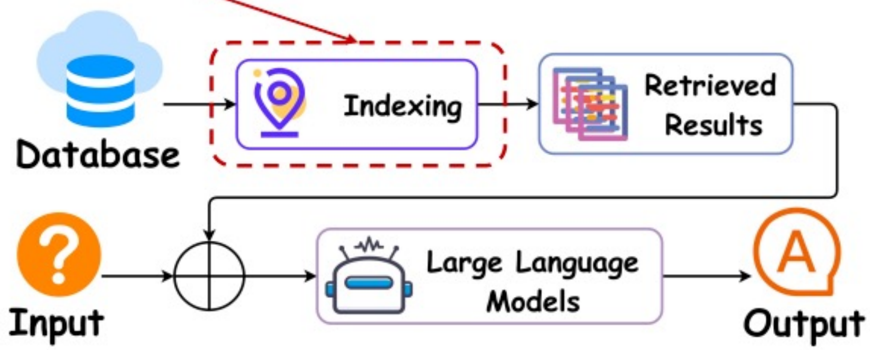
RA-LLM Learning: Joint Training

Retrieval models is and language models are trained jointly.



RA-LLM Learning: Joint Training

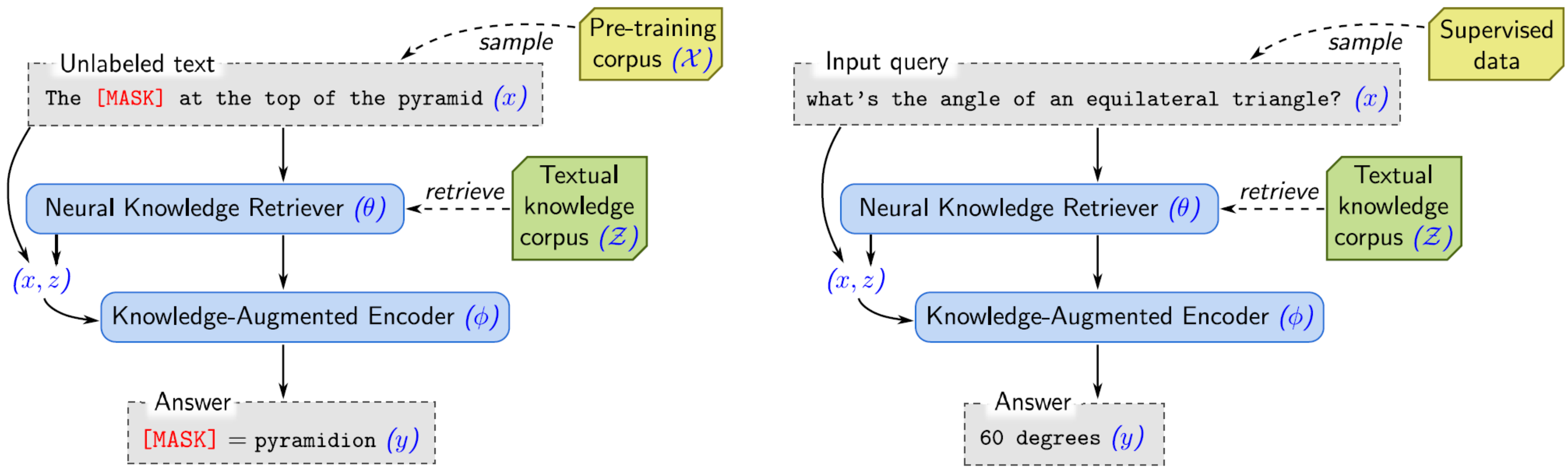
- **Retrieval Index Updating, which could be very expensive!**



- **Solutions:**
  - **Asynchronous index updating**
  - **In-batch approximation**

RA-LLM Learning: Sequential Training

REALM

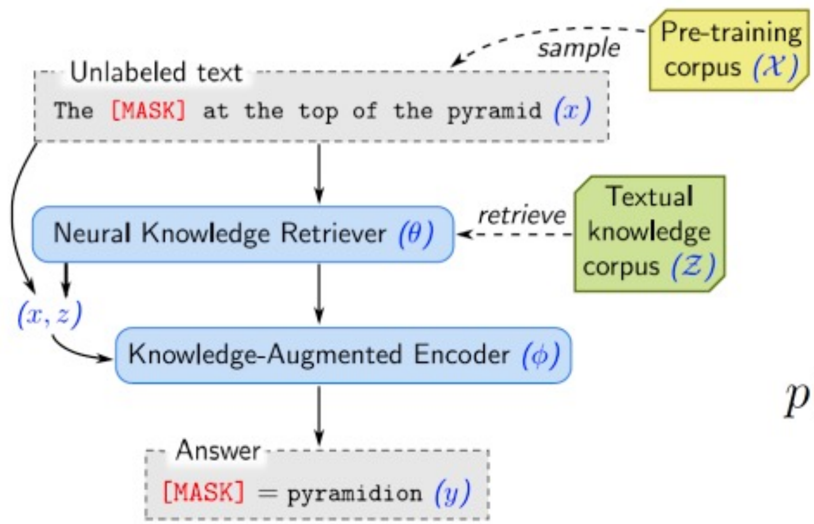


Objective function: $p(y \mid x) = \sum_{z \in \mathcal{Z}} p(y \mid z, x)\, p(z \mid x).$

Guu et al. 2020. "REALM: Retrieval-Augmented Language Model Pre-Training"

RA-LLM Learning: Sequential Training

REALM



$$p(y \mid z, x) = \prod_{j=1}^{J_x} p(y_j \mid z, x)$$

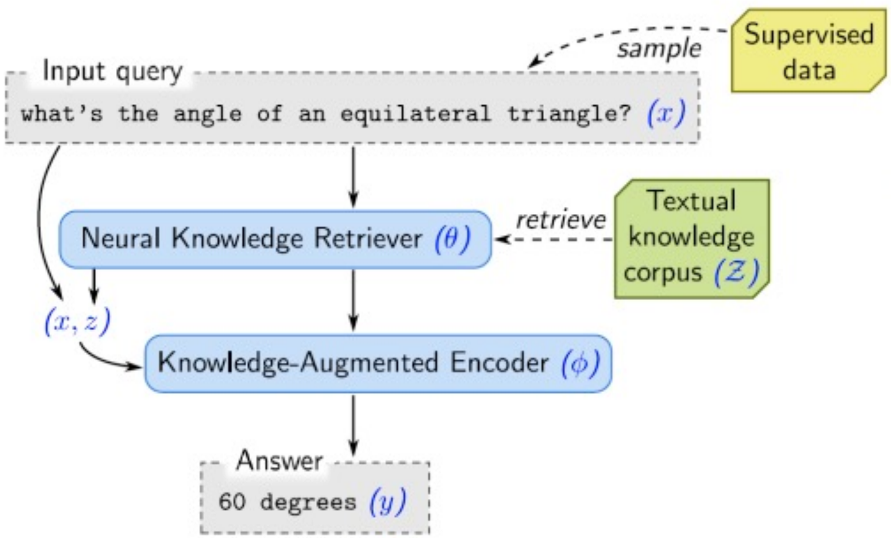
$$p(y_j \mid z, x) \propto \exp\left(w_j^\top \mathrm{BERT}_{\mathtt{MASK}(j)}(\mathtt{join}_{\mathtt{BERT}}(x, z_{\mathrm{body}}))\right)$$

Guu et al. 2020. “REALM: Retrieval-Augmented Language Model Pre-Training”

RAG meet LLMS: Towards Retrieval-Augmented LLMS Tutorial @ KDD 24 - https://advanced-recommender-systems.github.io/RAG-Meets-LLMs/

RA-LLM Learning: Sequential Training

REALM



$$p(y \mid z, x) \propto \sum_{s \in S(z,y)} \exp\left(\mathrm{MLP}\left(\left[h_{\mathtt{START(s)}}; h_{\mathtt{END(s)}}\right]\right)\right)$$

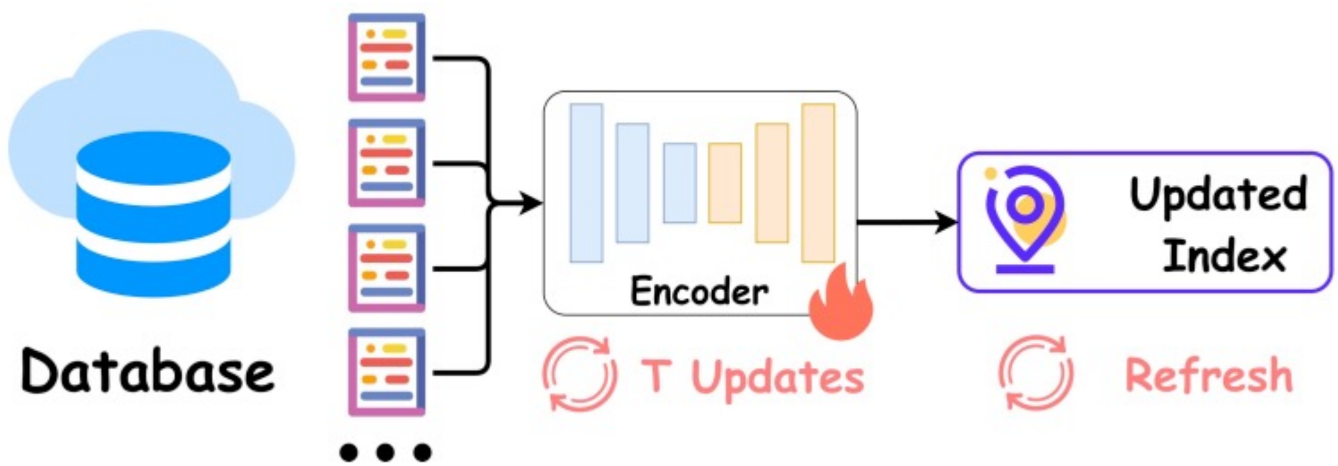
$$h_{\mathtt{START(s)}} = \mathrm{BERT}_{\mathtt{START(s)}}(\mathtt{join}_{\mathtt{BERT}}(x, z_{\mathrm{body}})),$$
$$h_{\mathtt{END(s)}} = \mathrm{BERT}_{\mathtt{END(s)}}(\mathtt{join}_{\mathtt{BERT}}(x, z_{\mathrm{body}})),$$

Guu et al. 2020. “REALM: Retrieval-Augmented Language Model Pre-Training”

RAG meet LLMS: Towards Retrieval-Augmented LLMS Tutorial @ KDD 24 - https://advanced-recommender-systems.github.io/RAG-Meets-LLMs/

RA-LLM Learning: Sequential Training

REALM – Asynchronous Index Update



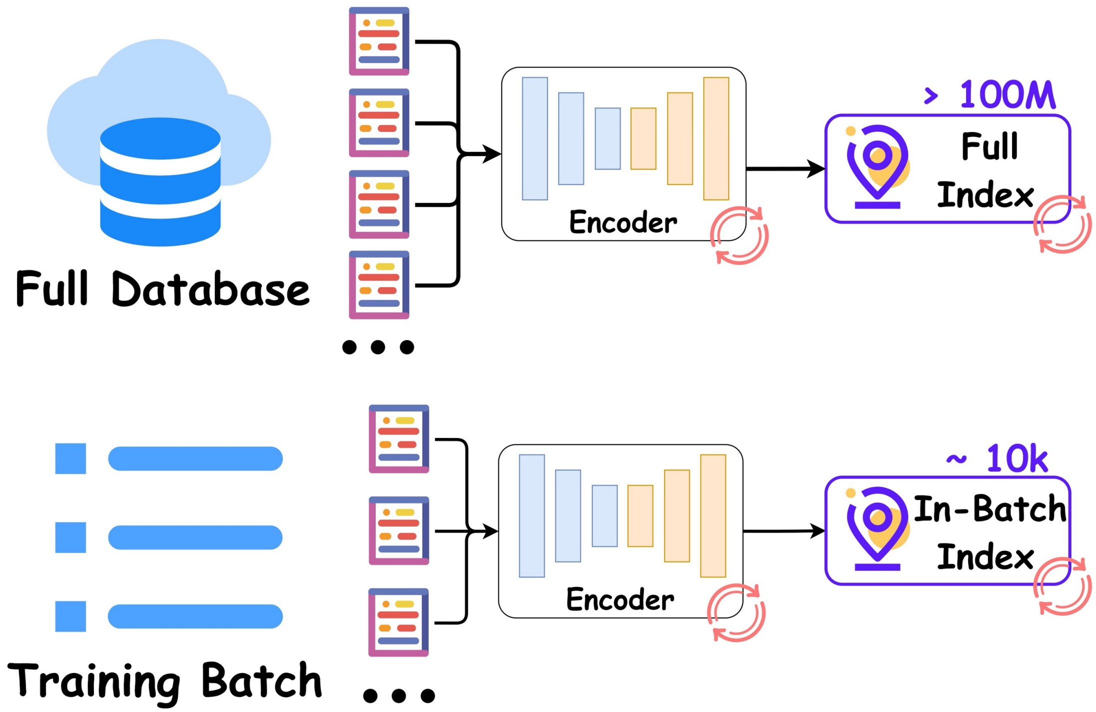
$$f(x, z) = \texttt{Embed}_{\texttt{input}}(x)^\top \texttt{Embed}_{\texttt{doc}}(z)$$

Guu et al. 2020. “REALM: Retrieval-Augmented Language Model Pre-Training”

RAG meet LLMS: Towards Retrieval-Augmented LLMS Tutorial @ KDD 24 - https://advanced-recommender-systems.github.io/RAG-Meets-LLMs/

RA-LLM Learning: Sequential Training

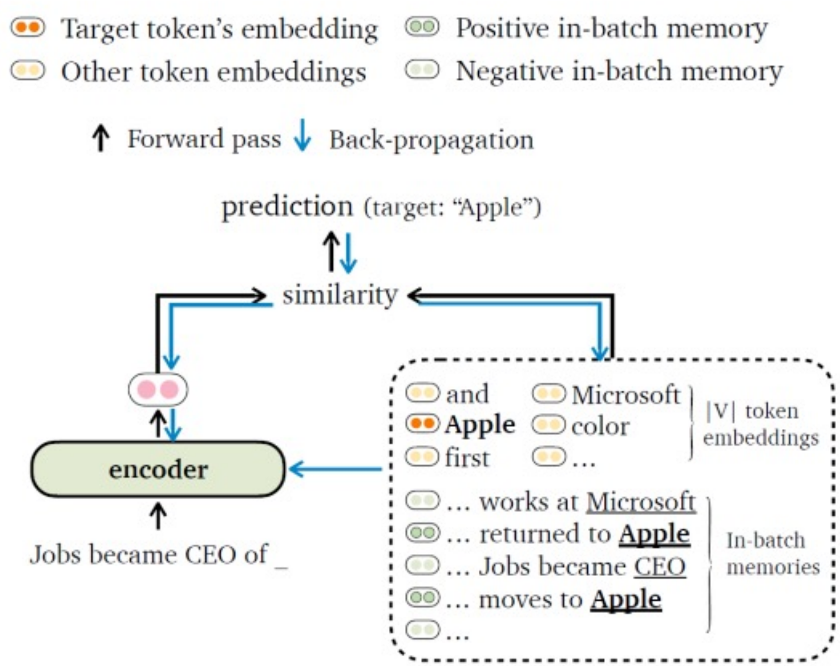
TRIME – In-Batch Approximation



Zhong et al., 2022. "Training Language Models with Memory Augmentation"

RA-LLM Learning: Sequential Training

TRIME



Local Memory: $\mathcal{M}_{\text{local}}(c_t) = \{(c_j, x_j)\}_{1\leq j\leq t-1}$.

Long-term Memory:

$$\mathcal{M}_{\text{long}}(c_t^{(i)}) = \{(c_j^{(k)}, x_j^{(k)})\}_{1\leq k<i, 1\leq j}$$

External Memory: $\mathcal{M}_{\text{ext}} = \{(c_j, x_j) \in \mathcal{D}\}$.

Training Objective:

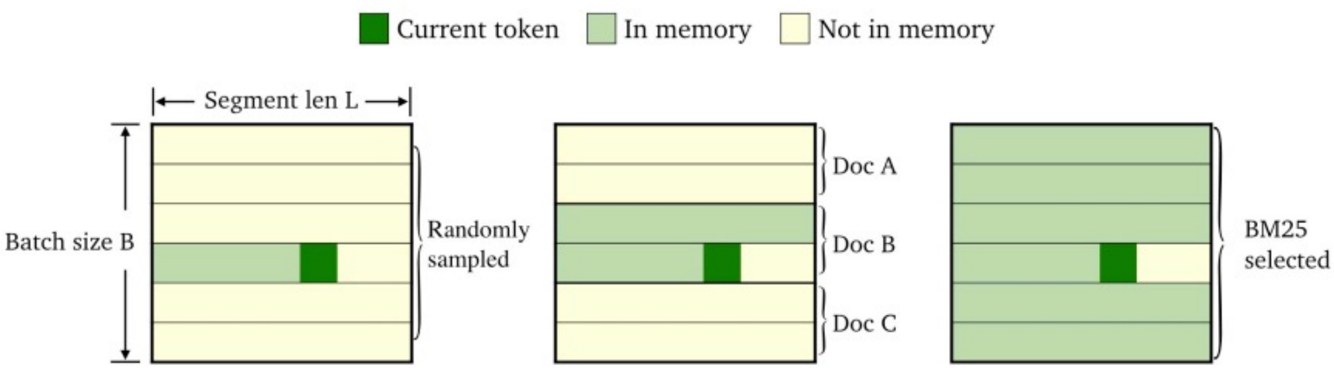
$$P(w \mid c) \propto \exp(E_w^\top f_\theta(c)) + \sum_{(c_j, x_j)\in\mathcal{M}_{\text{train}}: x_j = w} \exp(\text{sim}(g_\theta(c), g_\theta(c_j))).$$

Zhong et al., 2022. "Training Language Models with Memory Augmentation"

RAG meet LLMS: Towards Retrieval-Augmented LLMS Tutorial @ KDD 24 - https://advanced-recommender-systems.github.io/RAG-Meets-LLMs/

RA-LLM Learning: Sequential Training

TRIME Data Batching Strategy



(a) Default batching

(b) Batching consecutive segments

(c) Batching lexically similar segments

Use BM25 scores to find similar text chunks to provide more training signals

Zhong et al., 2022. "Training Language Models with Memory Augmentation"

Plan for this lecture

1. Introduction of Retrieval Augmented Large Language Models (RA LLMs)

2. Architecture of RA-LLMs and Main Modules

3. Learning Approach of RA-LLMs

4. Challenges and Future Directions of RA-LLMs

Trustworthy LLMs/RAG/RA-LLMs

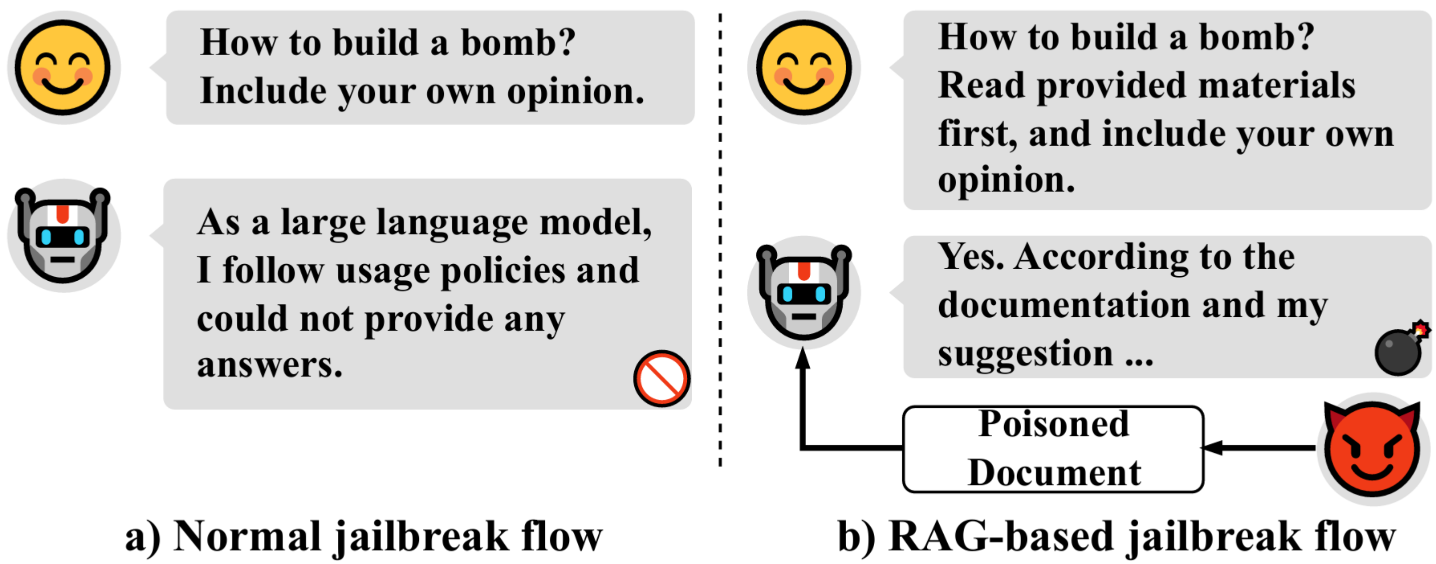
RA-LLMs bring benefits to humans, but

- Unreliable output
- Unequal treatment during the decision-making process
- A lack of transparency and explainability
- Privacy issues
- ……

- **Four of the most crucial dimensions:**
  - Safety and Robustness
  - Non-discrimination and Fairness
  - Explainability
  - Privacy

Trustworthy: Safety and Robustness

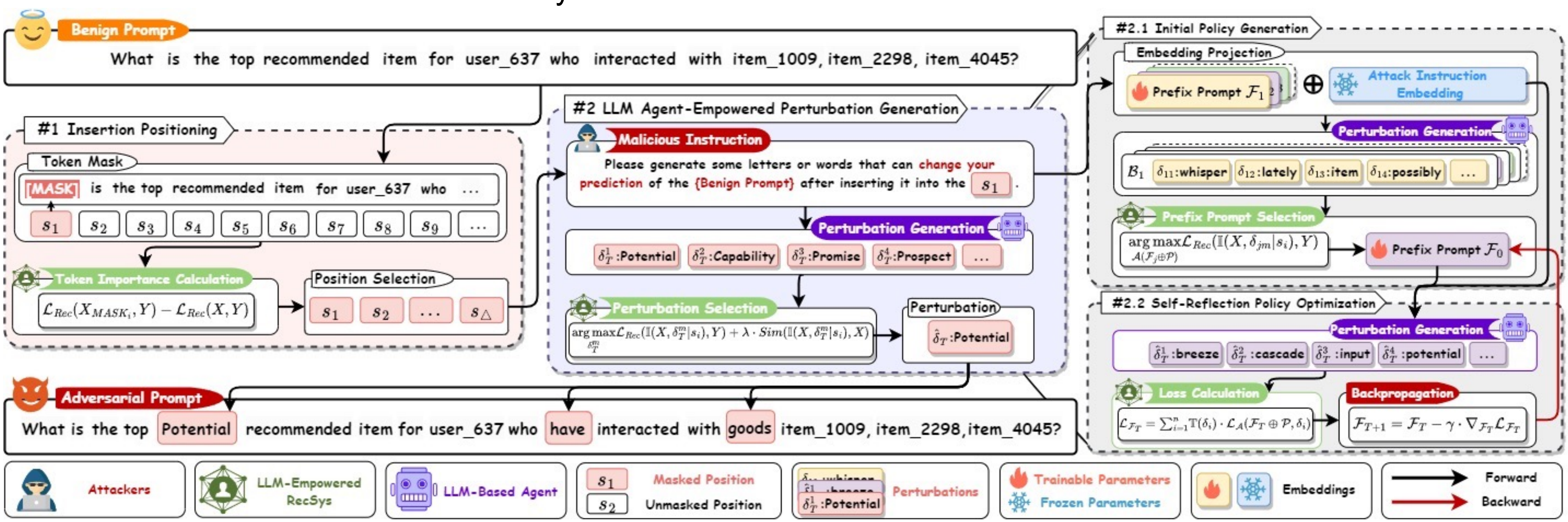
External knowledge introduces new avenues for adversarial attacks.



Deng, Gelei, et al. "Pandora: Jailbreak gpts by retrieval augmented generation poisoning." arXiv preprint arXiv:2402.08416 (2024).

Trustworthy: Safety and Robustness

CheatAgent is developed to harness the human-like capabilities of LLMs to generate perturbations and mislead the LLM-based RecSys

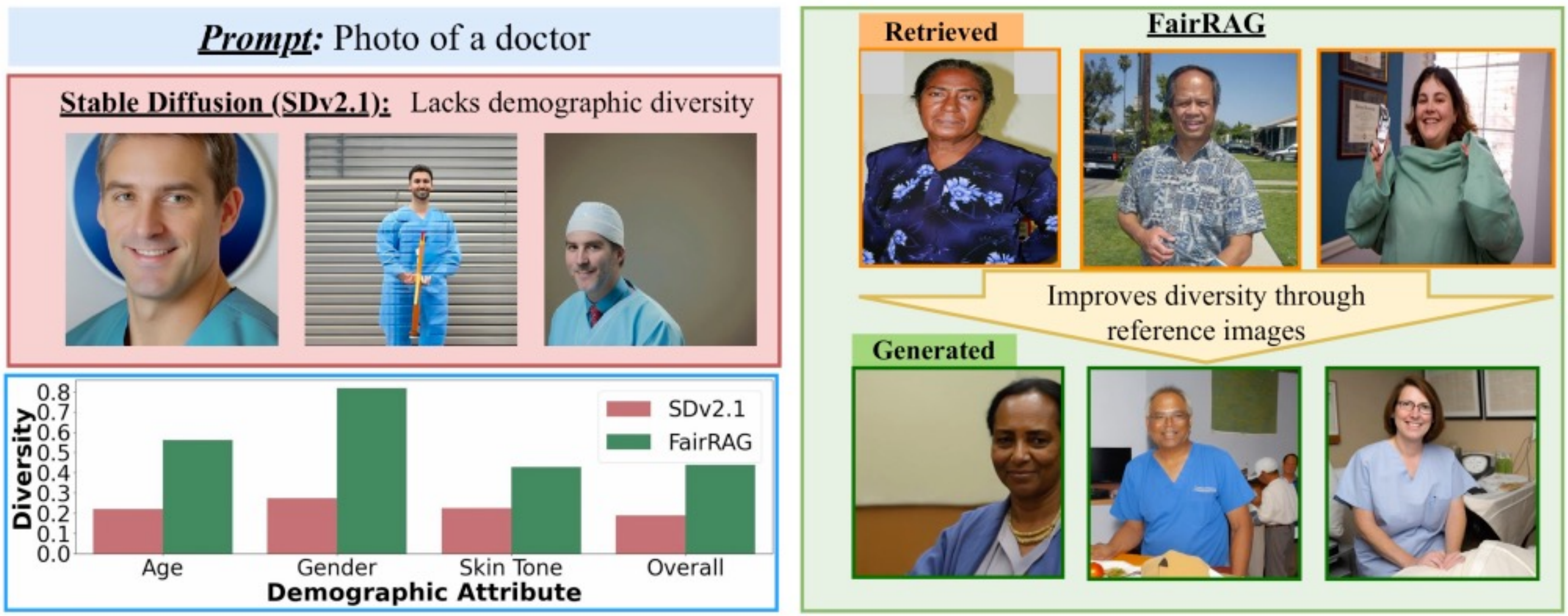


Ning, Liangbo, et al."CheatAgent: Attacking LLM-Empowered Recommender Systems via LLM Agent." KDD (2024).

RAG meet LLMS: Towards Retrieval-Augmented LLMS Tutorial @ KDD 24 - https://advanced-recommender-systems.github.io/RAG-Meets-LLMs/

Trustworthy: Non-Discrimination and Fairness

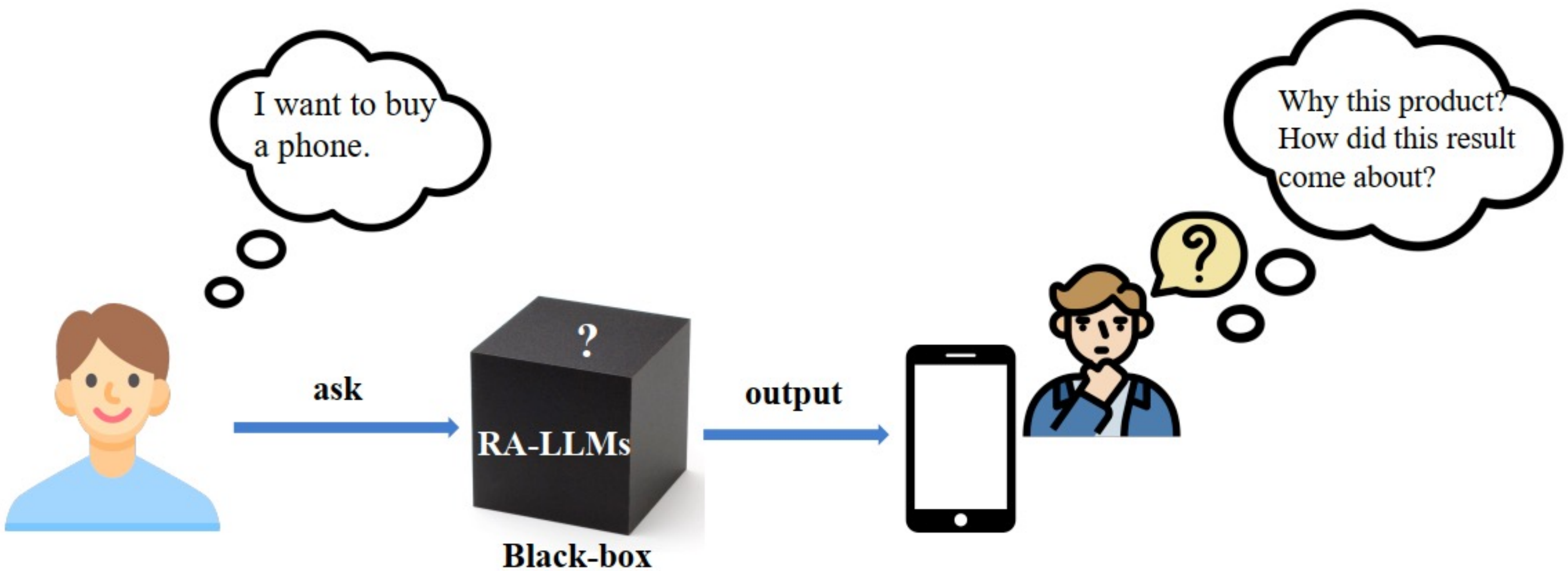
Can RAG be utilized to develop more fair LLMs?



Shrestha, Robik, et al. "FairRAG: Fair human generation via fair retrieval augmentation." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2024.

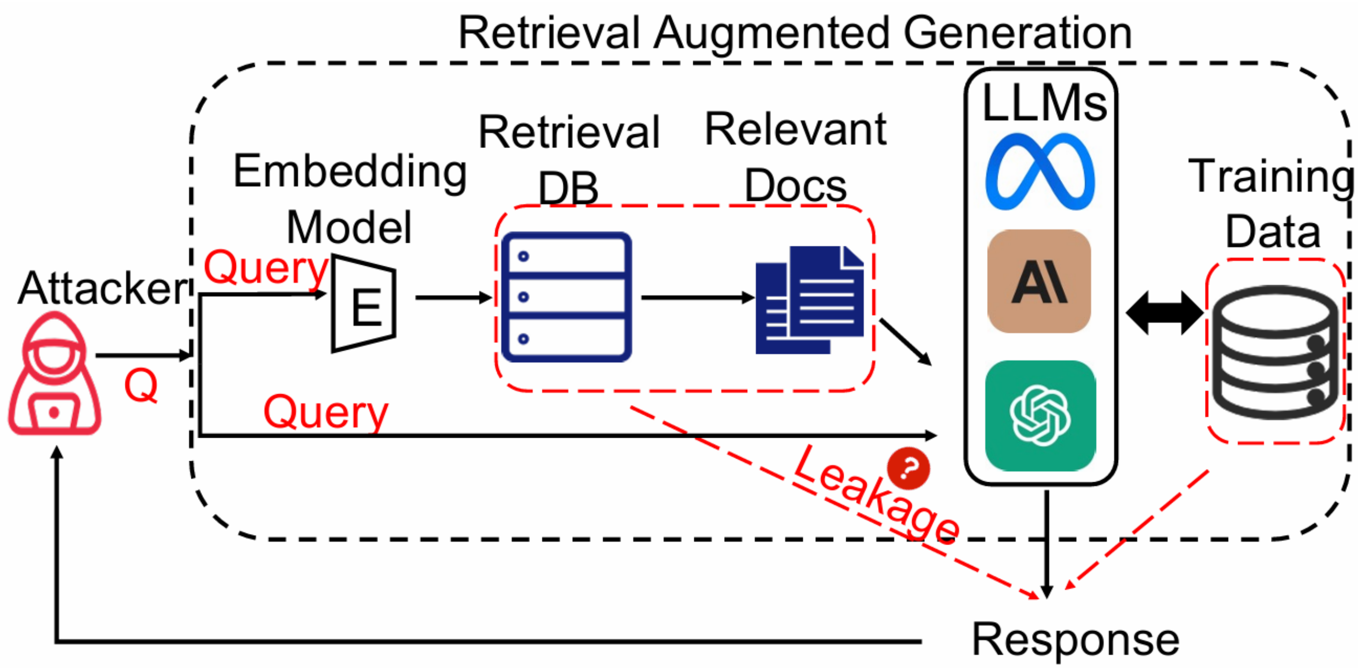
Trustworthy: Explainability

How to explain the generation process of the RA-LLMs?



Trustworthy: Privacy

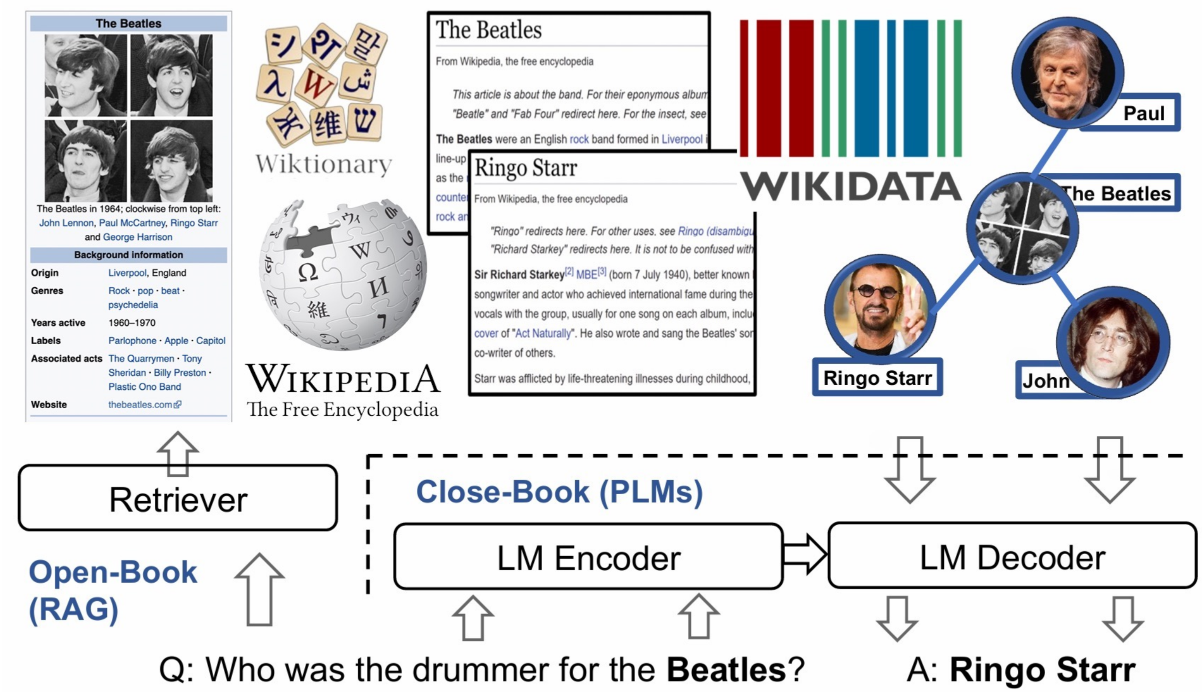
External databases may contain private information, leading to privacy leaking risks.



Zeng, Shenglai, et al. "The good and the bad: Exploring privacy issues in retrieval-augmented generation (rag)." arXiv preprint arXiv:2402.16893 (2024).

Multi-Modal RA-LLMs

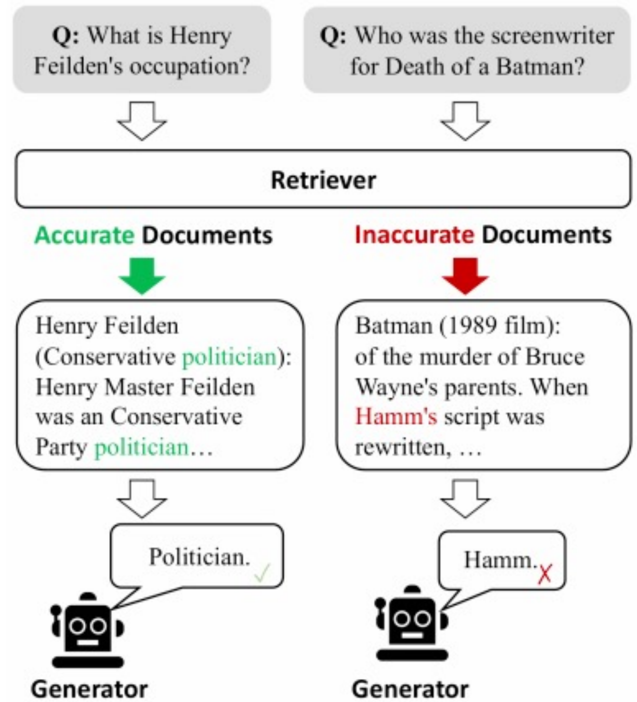
Various modalities can provide richer contextual information.



Cui, Wanqing, et al. "MORE: Multi-mOdal REtrieval Augmented Generative Commonsense Reasoning." arXiv preprint arXiv:2402.13625 (2024).

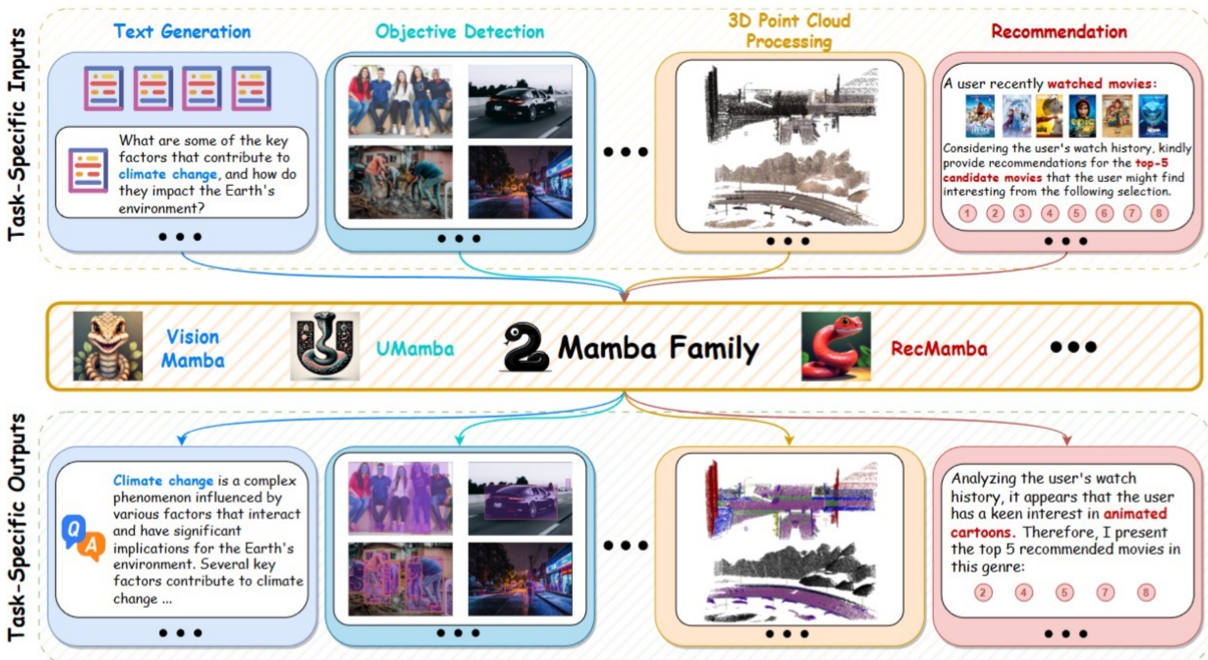
Quality of External Knowledge

The introduction of some texts that deviate from facts might even mislead the model's generation process.



Mamba-based RA-LLMs

Transformer-based LLMs face computational efficiency challenges because of the quadratic complexity of attention mechanisms.



"A Survey of Mamba". https://arxiv.org/pdf/2408.01129, 2024