The University Of Queensland
AUSTRALIA
CREATE CHANGE

Generative IR

@ ESSIR 2024

Guido Zuccon
ielab, The University of Queensland, Australia

www.ielab.io

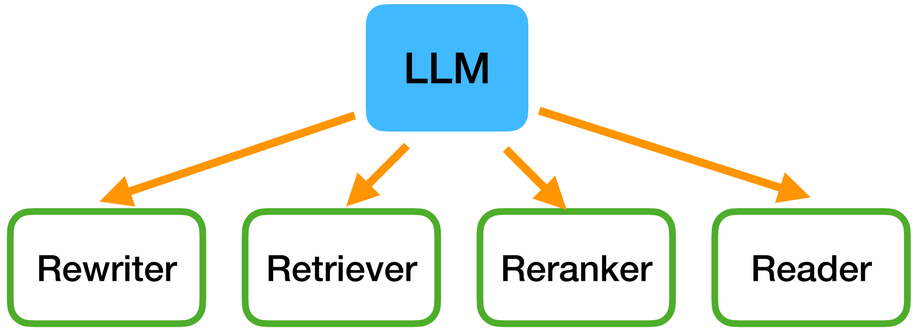
ie
lab

Focus of This Lecture

- Generative IR: The use of sequence-to-sequence language models to perform IR tasks
    - Encoder-decoder LLMs (e.g., T5) and decoder-only LLMs (e.g., GPT)
- In this lecture when we say LLMs we mean any of the two type of models above
    - i.e. we do not mean BERT-style models (encoder-only)

Three Directions in Generative IR

Prompting LLMs for Retrieval and Ranking

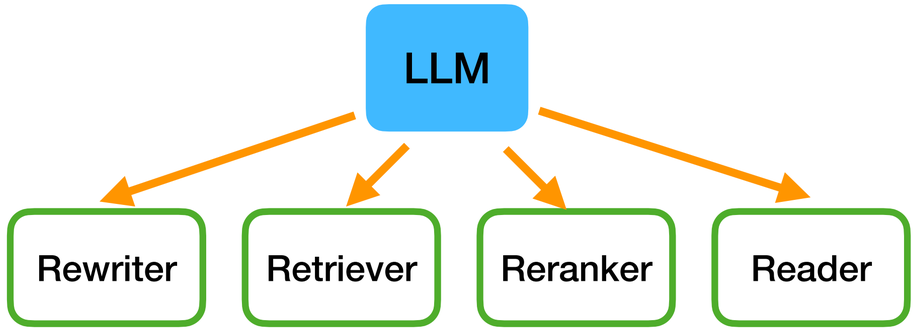


LLM-enhanced information retrieval

e.g. RankGPT, PromptReps

Three Directions in Generative IR

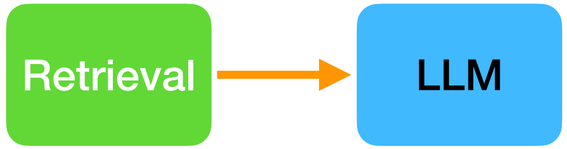
Prompting LLMs for Retrieval and Ranking



LLM-enhanced information retrieval

e.g. RankGPT, PromptReps

Retrieval Augmented Generation

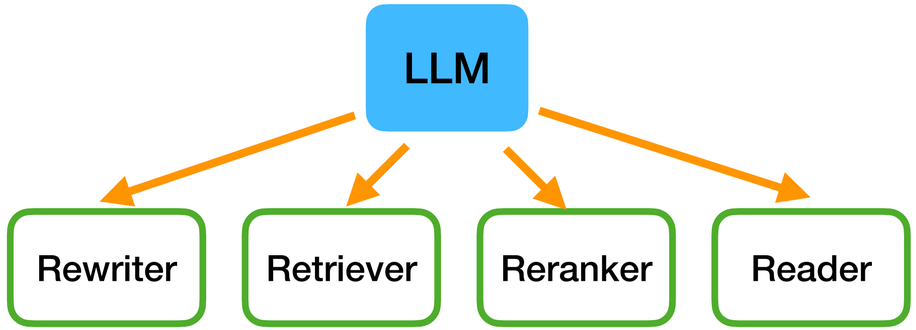


IR4LLM,
Retrieval-augmented LLM,
Reliable response generation

e.g. Self-RAG, BlendFilter, etc.

Three Directions in Generative IR

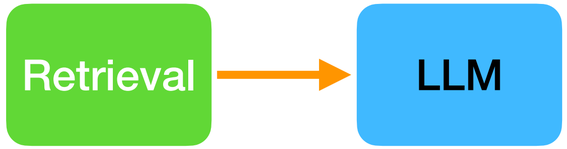
Prompting LLMs for Retrieval and Ranking



LLM-enhanced information retrieval

e.g. RankGPT, PromptReps

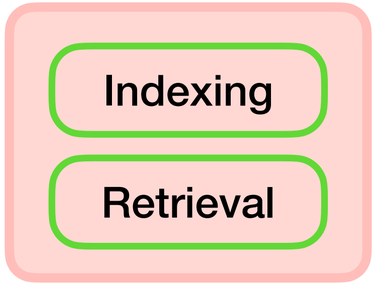
Retrieval Augmented Generation



IR4LLM, Retrieval-augmented LLM, Reliable response generation

e.g. Self-RAG, BlendFilter, etc.

Model-based IR



LLM as Indexer and Retriever, Generative [document] Retrieval (GR)

e.g. Differentiable Search Index (DSI)

ie lab

The University Of Queensland AUSTRALIA
CREATE CHANGE

Disclaimer

- Would **not have time to cover all the slides**: slides are made publicly available for offline consultation + pointers to more readings

- Most of the time can **only provide a few examples** of methods, **not** a **comprehensive** overview of all methods:

    1. sometimes **preference to more recent work** (i.e. arXiv, not peer-reviewed)-> **not necessarily perfect work**, but examples of very fresh directions
    (most recents on arXiv on July 1st, 2024)

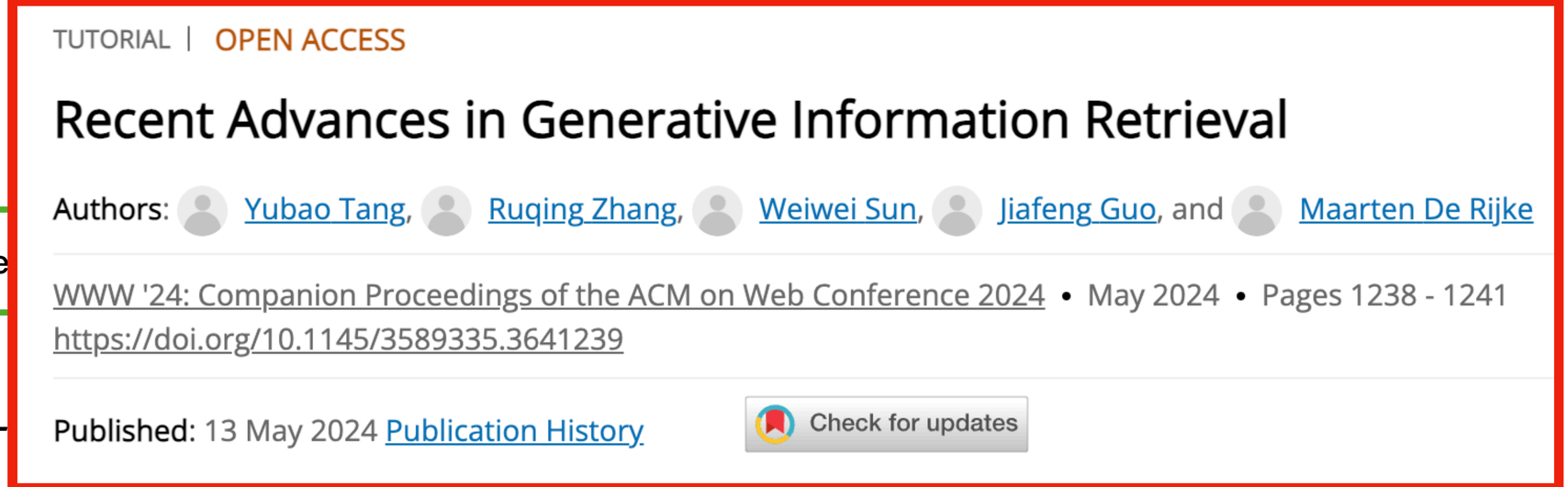
    2. Other times to more **fundamental** work

- Some methods are talked about in a **simplified/restricted/high-level** manner to give you the gist — always read the original paper to get more if interested

    - Please pardon my lingo sloppiness from time to time

Three Directions in Generative IR

Will spend the least amount of time on DSI



TUTORIAL | OPEN ACCESS

Recent Advances in Generative Information Retrieval

Authors: Yubao Tang, Ruqing Zhang, Weiwei Sun, Jiafeng Guo, and Maarten De Rijke

WWW '24: Companion Proceedings of the ACM on Web Conference 2024 • May 2024 • Pages 1238 - 1241
https://doi.org/10.1145/3589335.3641239

Published: 13 May 2024 Publication History

Check for updates

also coming to SIGIR 2024!

Model-based IR

Indexing

Retrieval

LLM as Indexer and Retriever, Generative [document] Retrieval (GR)

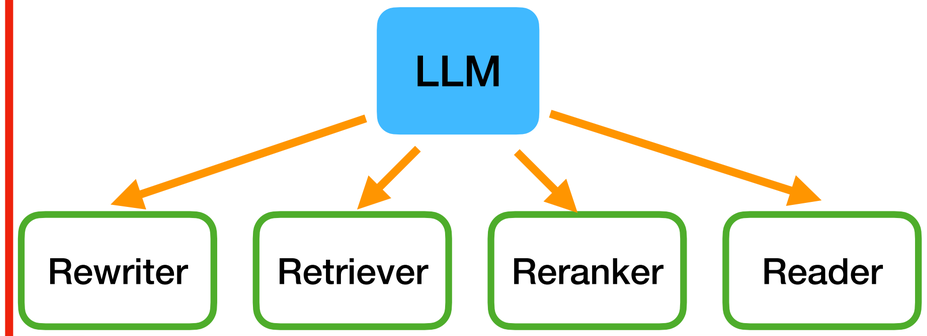
e.g. Differentiable Search Index (DSI)

ie lab

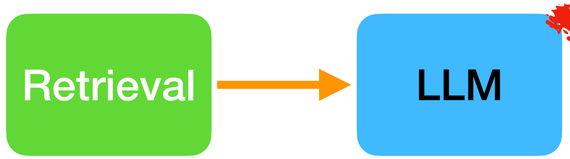
The University of Queensland
AUSTRALIA
CREATE CHANGE

Three Directions in Generative IR

Prompting LLMs for Retrieval and Ranking



Retrieval Augmented Generation



In the live presentation I incorrectly attributed a great tutorial on LLMs4IR from the Chinese IR community to BAAI — the correct attribution should be made to CFF. The event was part of the CCF Advanced Disciplines Lectures series. Correct link: https://ccf.org.cn/ADL147; videos are not yet online but are supposed to be released at https://dl.ccf.org.cn/

Some great tutorial resources on other aspects of Generative IR too

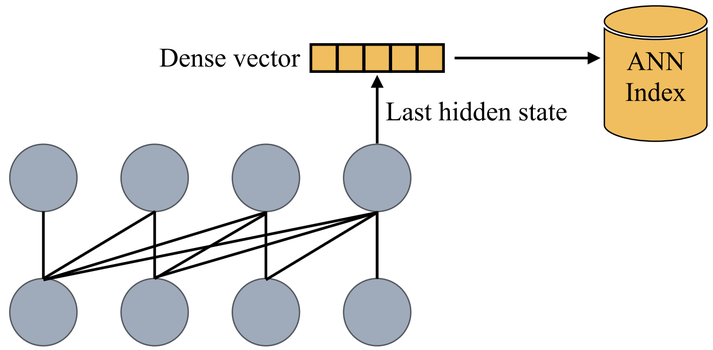
https://2024.baai.ac.cn/

Part 1: Prompting LLMs for Retrieval and Ranking

Key Intuitions

Can we ask LLMs to:

- **Retrieve:** generate representations for a document/query?
  - Then we can use representations to do matching, e.g. create embeddings and use them for dense retrieval

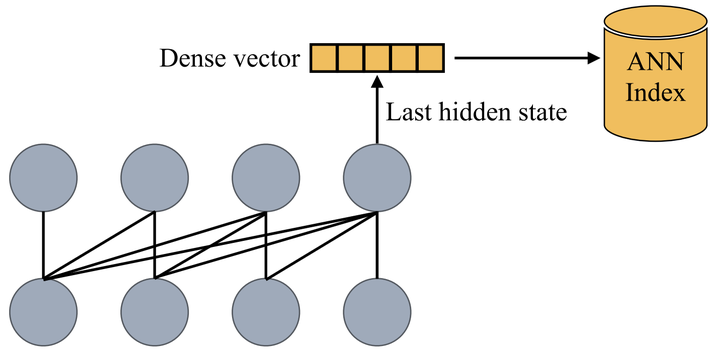


<System> *You are an AI assistant that can understand human language.*
<User> *Passage: "***[text]***". Use one word to represent the passage in a retrieval task. Make sure your word is in lowercase.*
<Assistant> *The word is: "*

Key Intuitions

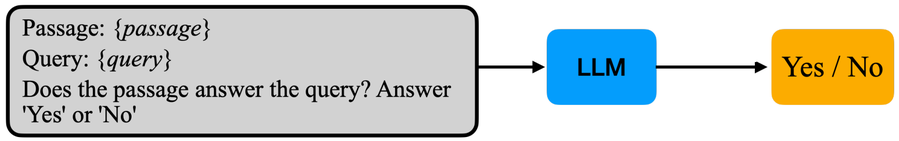
Can we ask LLMs to:

- **Retrieve:** generate representations for a document/query?
  - Then we can use representations to do matching, e.g. create embeddings and use them for dense retrieval



<System> *You are an AI assistant that can understand human language.*
<User> *Passage: "***[text]***". Use one word to represent the passage in a retrieval task. Make sure your word is in lowercase.*
<Assistant> *The word is: "*

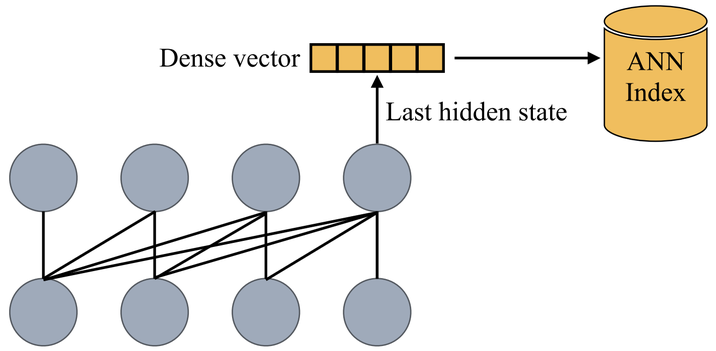
- **Ranking:** tell us the relevance of a document to a query?
  - Then we can use this indication of relevance (or relative relevance of n documents) to rank documents for the query



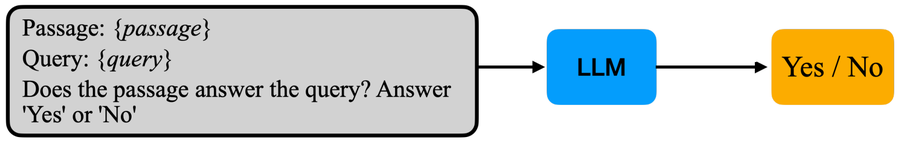
Key Intuitions

Can we ask LLMs to:

- **Retrieve:** generate representations for a document/query?
  - Then we can use representations to do matching, e.g. create embeddings and use them for dense retrieval
- **Ranking:** tell us the relevance of a document to a query?
  - Then we can use this indication of relevance (or relative relevance of n documents) to rank documents for the query



<System> *You are an AI assistant that can understand human language.*
<User> *Passage: "***[text]***". Use one word to represent the passage in a retrieval task. Make sure your word is in lowercase.*
<Assistant> *The word is: "*

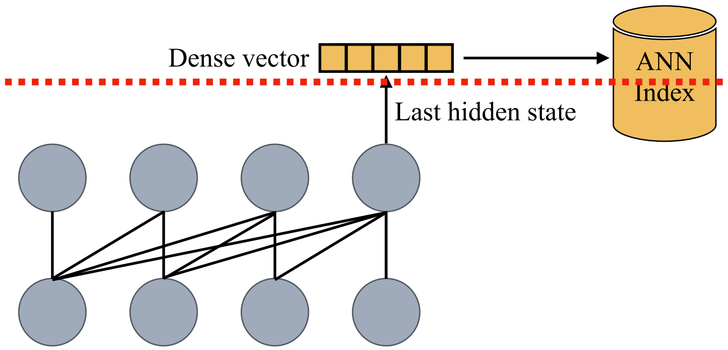


Idea: devise prompts/instructions to tell the LLM how to perform these tasks effectively

Key Intuitions

Can we ask LLMs to:

- **Retrieve:** generate representations for a document/query?
  - Then we can use representations to do matching, e.g. create embeddings and use them for dense retrieval



<System> *You are an AI assistant that can understand human language.*
<User> *Passage: "***[text]***". Use one word to represent the passage in a retrieval task. Make sure your word is in lowercase.*
<Assistant> *The word is: "*

- **Ranking:** tel
  - Then we
    relevanc
    query

Plan:

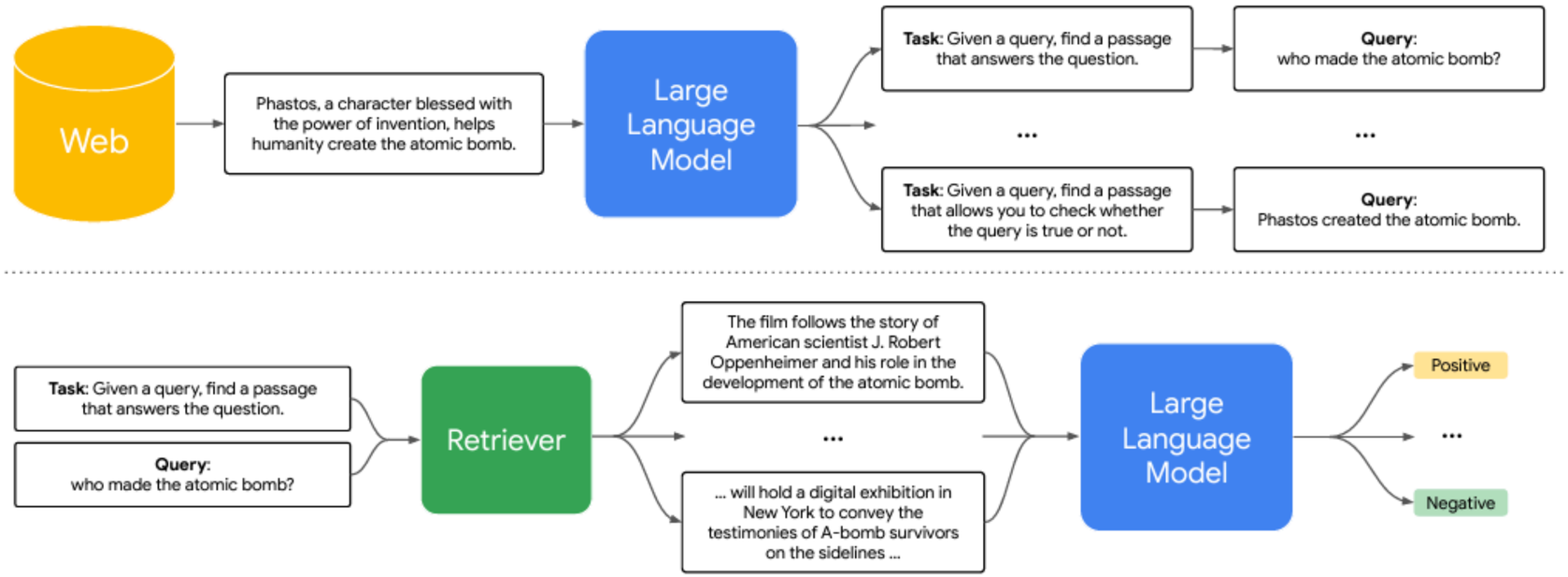
1. LLMs for **dense** bi-encoding:
   (i) leverage LLM for data generation (e.g. Gecko, E5)
   (ii) as backbone (e.g. E5, LLM2Vec)
2. LLMs as a zero-shot **hybrid** representation creation (PromptReps)

Yes / No

ie lab

Gecko embeddings:

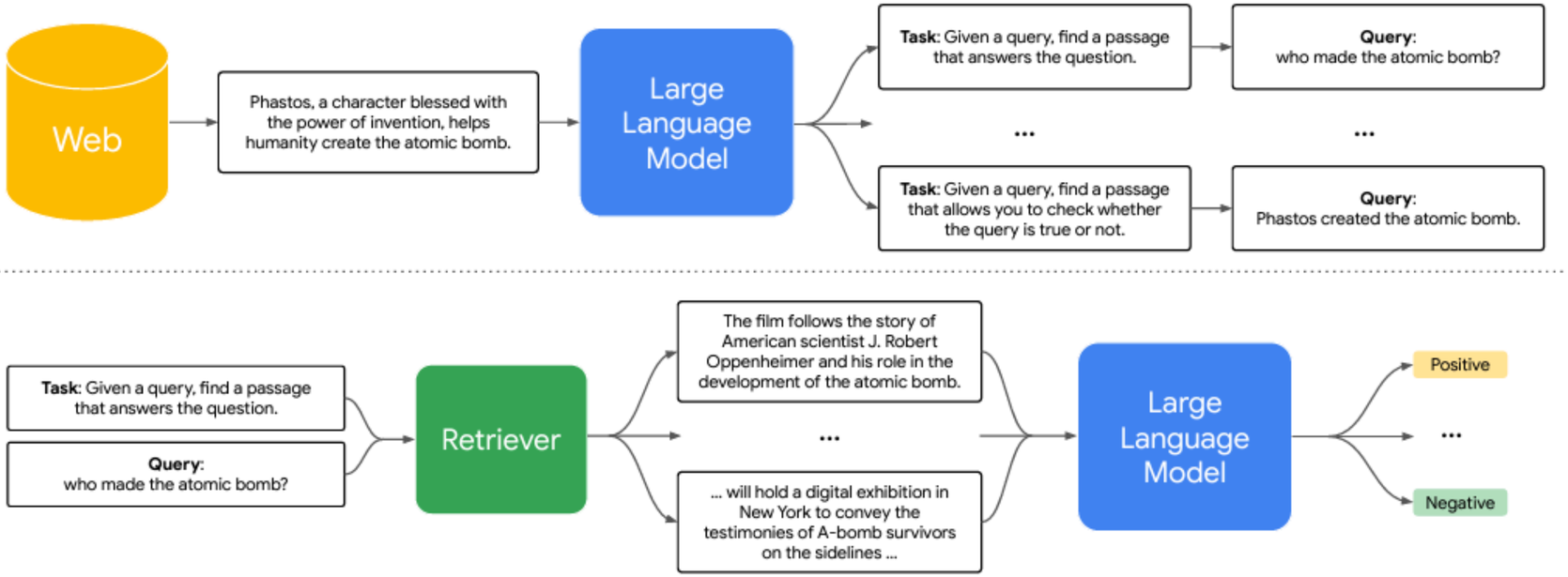
- The underlying LLM architecture is not specified; likely uses Google's Matryoshka embeddings technique
- Training relies on LLMs
    - LLMs used to generate a Few-shot Prompted Retrieval dataset (FRet) for knowledge distillation from LLM into embedding through 2 tasks: (1) diverse query generation, (2) positive & negative mining



Lee, J., Dai, Z., Ren, X., Chen, B., Cer, D., Cole, J.R., Hui, K., Boratko, M., Kapadia, R., Ding, W. and Luan, Y., 2024. Gecko: Versatile text embeddings distilled from large language models. *arXiv preprint arXiv:2403.20327*.

Gecko embeddings:

- The underlying LLM architecture is not specified; likely uses Google's Matryoshka embeddings technique
- Training relies on LLMs
    - LLMs used to generate a Few-shot Prompted Retrieval dataset (FRet) for knowledge distillation from LLM into embedding through 2 tasks: (1) diverse query generation, (2) positive & negative mining



(1) Query/task generation

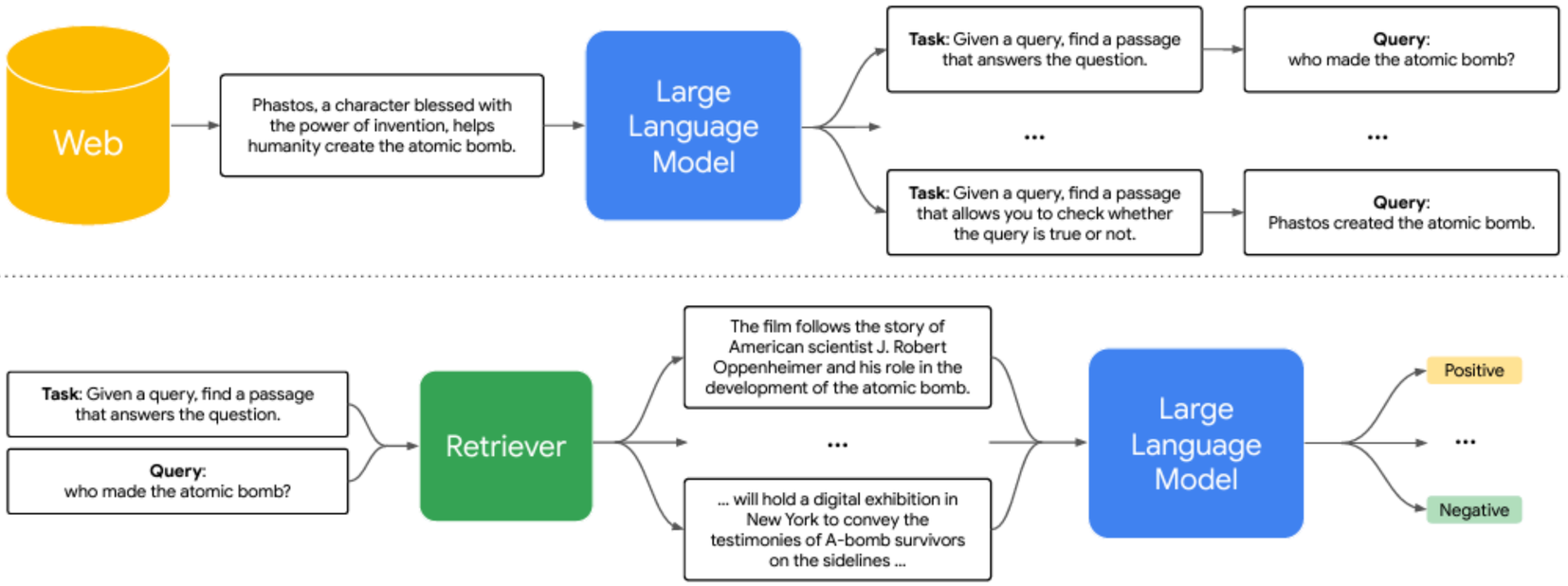
Prompt for task generation:

- *"Given a query, find a passage that has the answer to the query"* [question answering]
- *"Given a query, find a passage that allows you to check whether the query is true or not"* [fact checking)]

Lee, J., Dai, Z., Ren, X., Chen, B., Cer, D., Cole, J.R., Hui, K., Boratko, M., Kapadia, R., Ding, W. and Luan, Y., 2024. Gecko: Versatile text embeddings distilled from large language models. *arXiv preprint arXiv:2403.20327*.

Gecko embeddings:

- The underlying LLM architecture is not specified; likely uses Google's Matryoshka embeddings technique
- Training relies on LLMs
    - LLMs used to generate a Few-shot Prompted Retrieval dataset (FRet) for knowledge distillation from LLM into embedding through 2 tasks: (1) diverse query generation, (2) positive & negative mining



(2) Pos/Neg Mining

Observation: generated queries focus on particular aspect of original passage

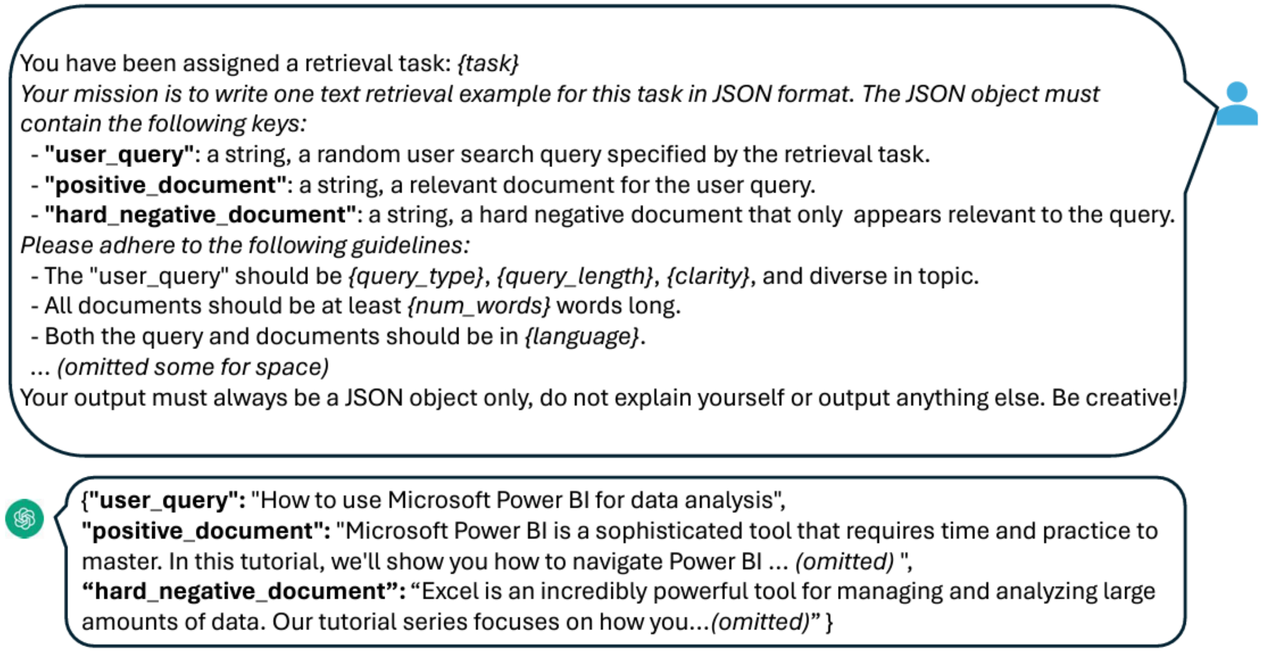
Sampling Method:

1. Use initial model trained with (q, $p$seed) pairs to retrieve top passages
2. Use an LLM to rank (2 tried: query likelihood (Sachin's UPR) and relevance classification (H. Zhuang's RG) — see later)
3. Reciprocal Rank Fusion of 1&2
4. Top ranked: p+; other tops or bottom p-

E5 with LLMs

- Original E5 embeddings are BERT-based; new E5 embedding are decoder-only LLM (Mistral-7b)
- E5-Mistral uses LLMs in two ways:

(A) **Synthetic data generation:** (1) prompt LLM (GPT4) to “brainstorm” a list of potential retrieval tasks, (2) generate <query, positive, hard negative> for each task.



Wang, L., Yang, N., Huang, X., Yang, L., Majumder, R. and Wei, F., 2023. Improving text embeddings with large language models. *arXiv preprint arXiv:2401.00368*.

E5 with LLMs

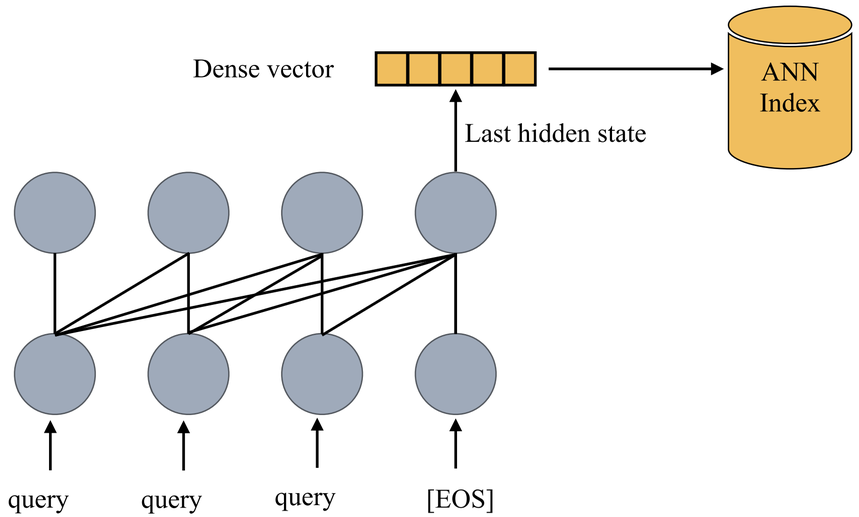
- Original E5 embeddings are BERT-based; new E5 embedding are decoder-only LLM (Mistral-7b)
- E5-Mistral uses LLMs in two ways:

(A) **Synthetic data generation:** (1) prompt LLM (GPT4) to “brainstorm” a list of potential retrieval tasks, (2) generate <query, positive, hard negative> for each task.

(B) **Embedding backbone:** (1) modify query to:

q+ inst = Instruct: {task_definition} \n Query: {q+}

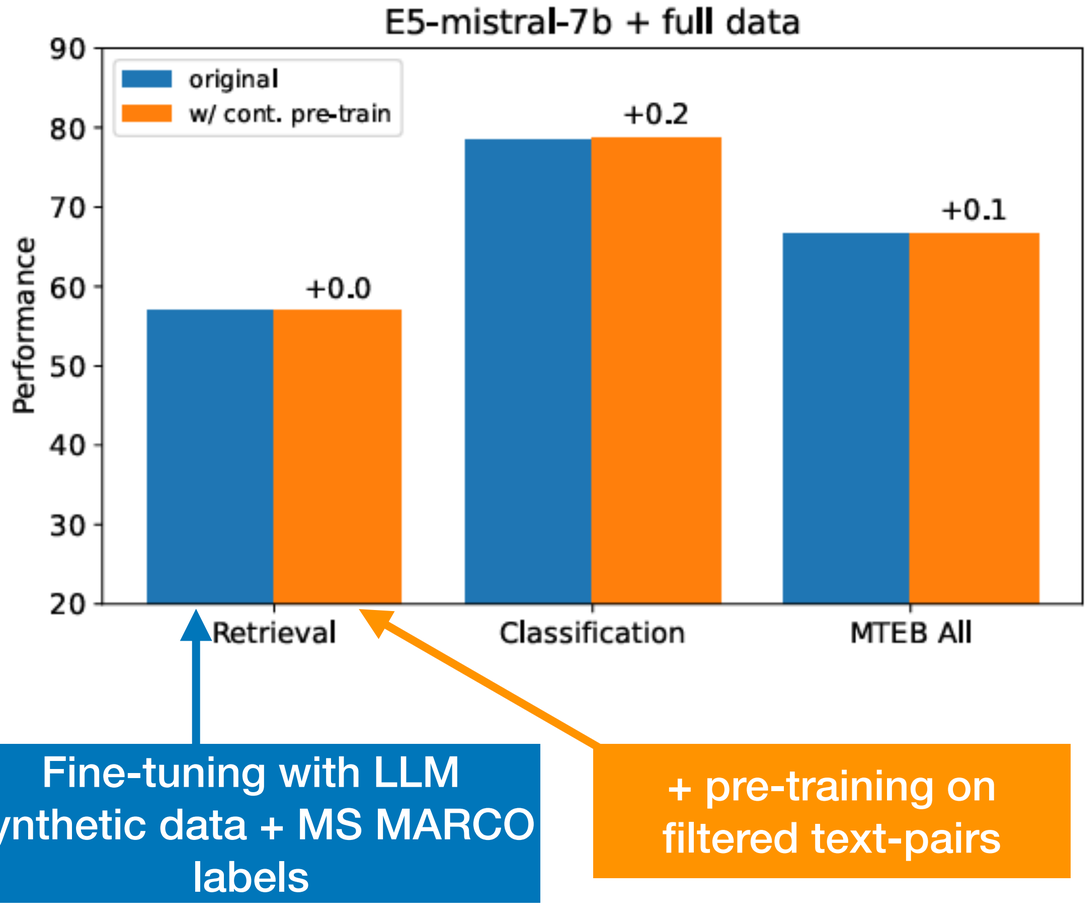
(2) append [EOS] token to end of query and document; feed them into LLM to get embeddings by taking last layer of [EOS] vector



- Embedding model then trained with standard contrastive loss over the in-batch negatives and hard negatives

E5 with LLMs

- Training in E5Mistral:
  - Contrastive pre-training: weakly supervised data from filtered text pairs (from E5 paper)
  - Contrastive fine-tuning: LLM generated synthetic data + MS MARCO labelled data
    [they also have version + 18 other datasets]
- Observation: contrastive *pre*-training has negligible impact on model quality.
  - extensive auto-regressive pre-training enables LLMs to acquire good text representations, and only fine-tuning required to obtain effective embeddings.



Wang, L., Yang, N., Huang, X., Yang, L., Majumder, R. and Wei, F., 2023. Improving text embeddings with large language models. *arXiv preprint arXiv:2401.00368*.

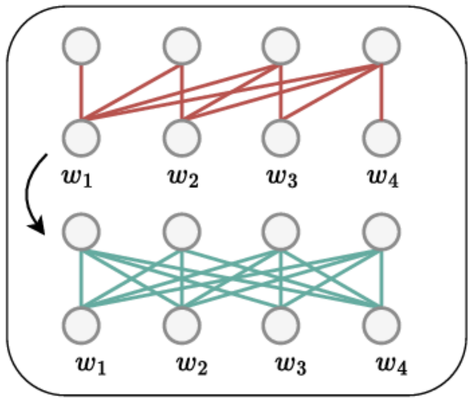
LLM2Vec

- Transform any decoder-only LLM into a strong text encoder
- Three key steps:

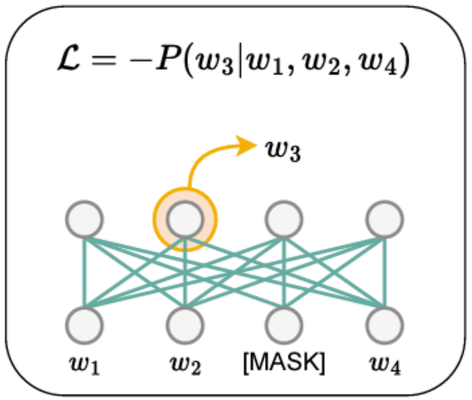
1. enable bidirectional attention,
2. masked next token prediction,
3. unsupervised contrastive learning

- (4) can add supervised contrastive learning

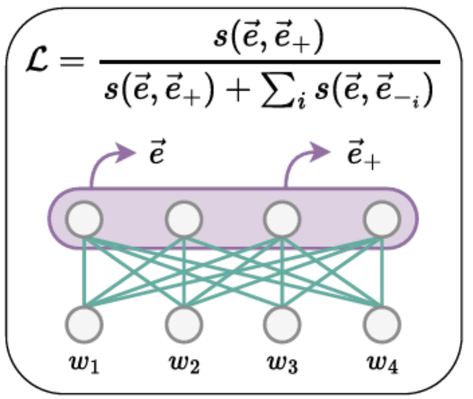
Enabling Bidirectional Attention



Masked Next Token Prediction



Unsupervised Contrastive Learning



Modify the self-attention mechanism in the decoder-only architecture by replacing the causal attention mask by an all-ones matrix

gives each token access to every other token in the sequence, converting it into a bidirectional LLM

simply enabling bidirectional attention decreases embedding performance: decoder-only LLM was not trained to attend to future tokens

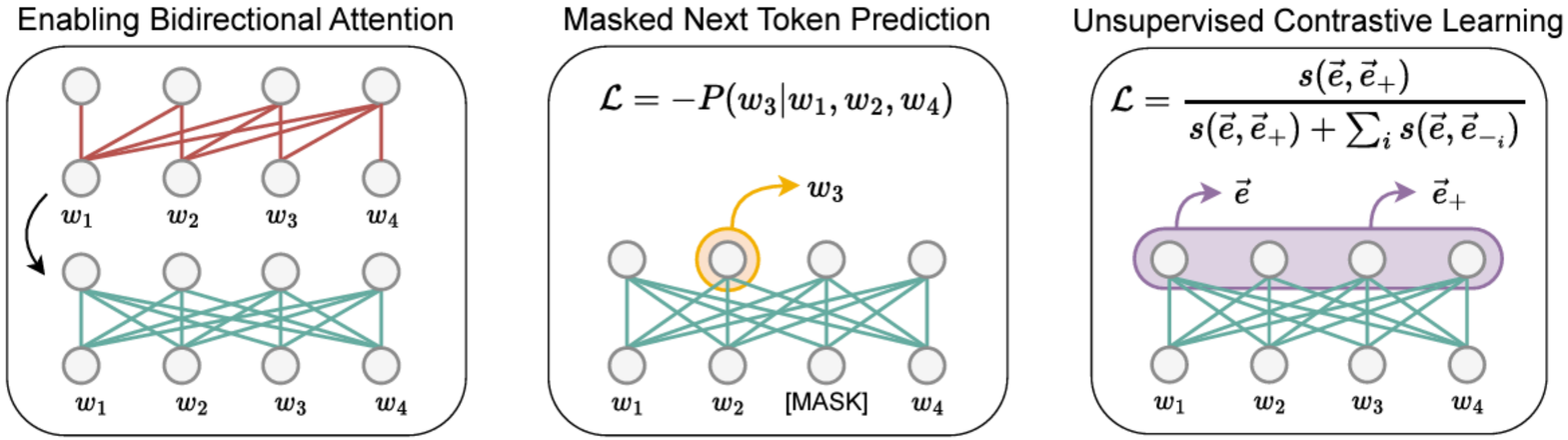
BehnamGhader, P., Adlakha, V., Mosbach, M., Bahdanau, D., Chapados, N. and Reddy, S., 2024. Llm2vec: Large language models are secretly powerful text encoders. *arXiv preprint arXiv:2404.05961*.

LLM2Vec

- Transform any decoder-only LLM into a strong text encoder
- Three key steps:

1. enable bidirectional attention,
2. masked next token prediction,
3. unsupervised contrastive learning

- (4) can add supervised contrastive learning



make the model aware of its bidirectional attention by adapting it via masked next token prediction

when predicting a masked token at position i, compute loss based on logits from token representation at position i – 1, not the masked position itself

No next sentence prediction objective in pre-training: not explicitly trained to capture the context of the entire sequence

ie lab

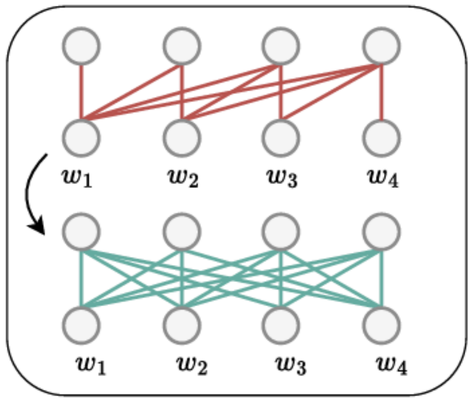
LLM2Vec

- Transform any decoder-only LLM into a strong text encoder
- Three key steps:

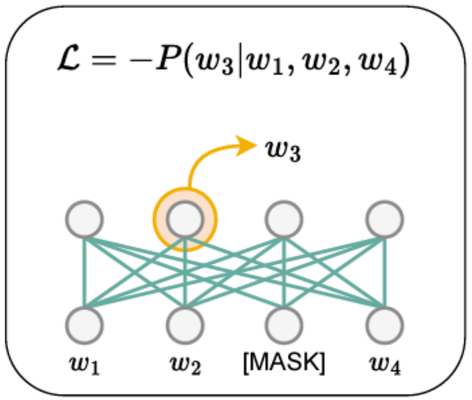
1. enable bidirectional attention,
2. masked next token prediction,
3. unsupervised contrastive learning

- (4) can add supervised contrastive learning

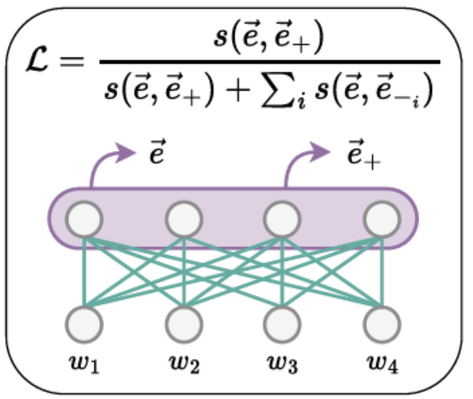
Enabling Bidirectional Attention



Masked Next Token Prediction



Unsupervised Contrastive Learning



unsupervised contrastive learning via SimCSE: pass input sequence into the model twice with independently sampled dropout masks

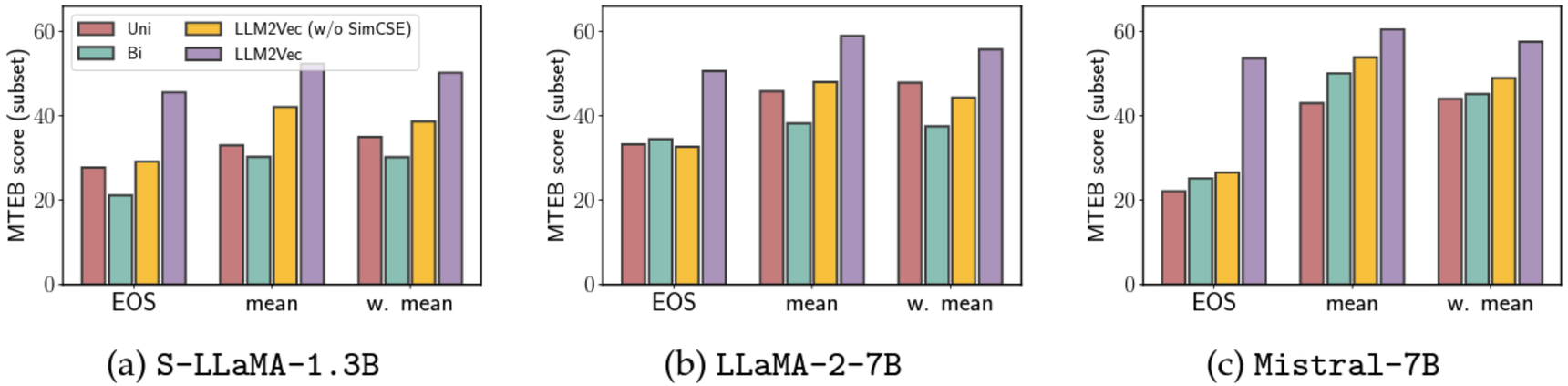
Results in two different representations for the same sequence

Train model to maximise similarity between the two sequences representations and minimise similarities with others in batch

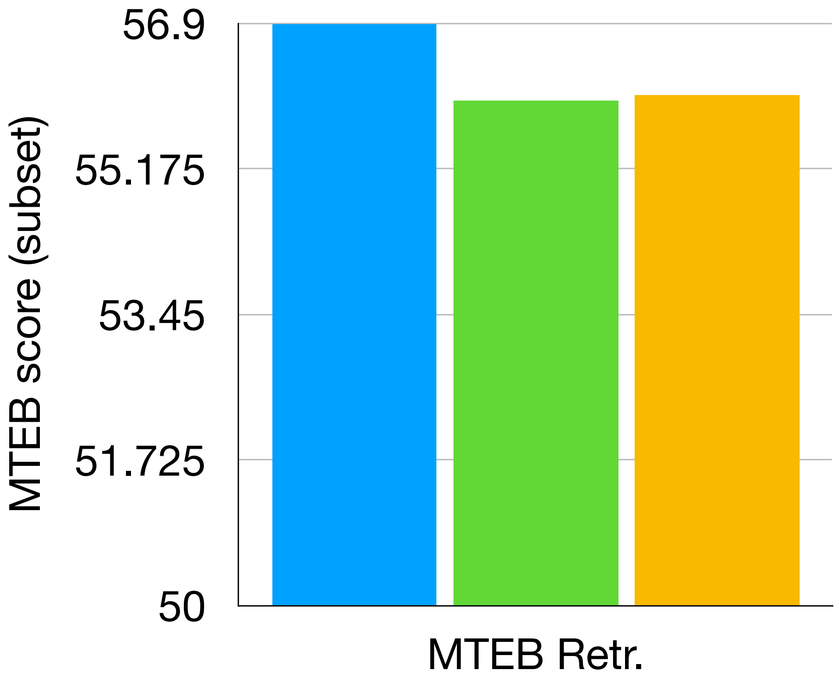
ie lab

LLM2Vec

To obtain representations, can use EOS, mean pooling or weighted mean pooling



(a) S-LLaMA-1.3B (b) LLaMA-2-7B (c) Mistral-7B



E5-Mistral-7B
LLM2Vec (w/o SimCSE)
LLM2Vec

Effectiveness of LLM2Vec with Mistral-7B backbone & contrastive fine-tuning is lower than E5-Mistral on MTEB Retrieval datasets

(Bigger differences in the re-ranking task)

PromptReps

- E5: gets embedding from last hidden layer of [EOS] token; requires (supervised) contrastive fine-tuning
- LLM2Vec: gets embedding from mean pooling of sequence tokens; requires (unsupervised/supervised) contrastive fine-tuning
- Can we engineer LLMs prompts to obtain an effective representation **without** need for contrastive fine-tuning?
  - PromptReps! — “zero-shot” generation of representations using LLMs

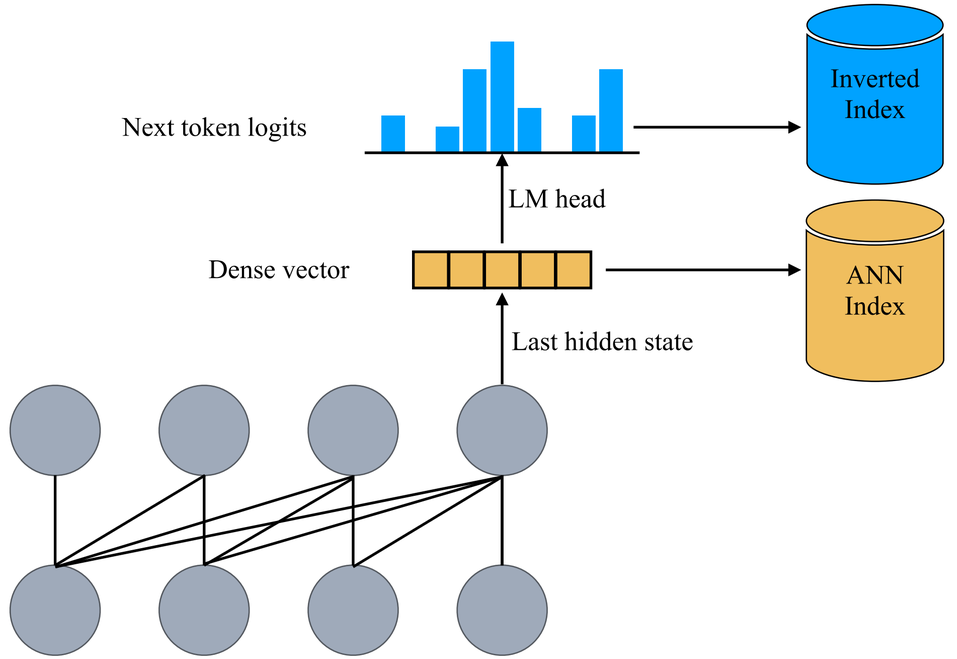
ie lab

Zhuang, S., Ma, X., Koopman, B., Lin, J. and Zuccon, G., 2024. PromptReps: Prompting Large Language Models to Generate Dense and Sparse Representations for Zero-Shot Document Retrieval. *arXiv preprint arXiv:2404.18424*.

PromptReps

PromptReps at a high level:

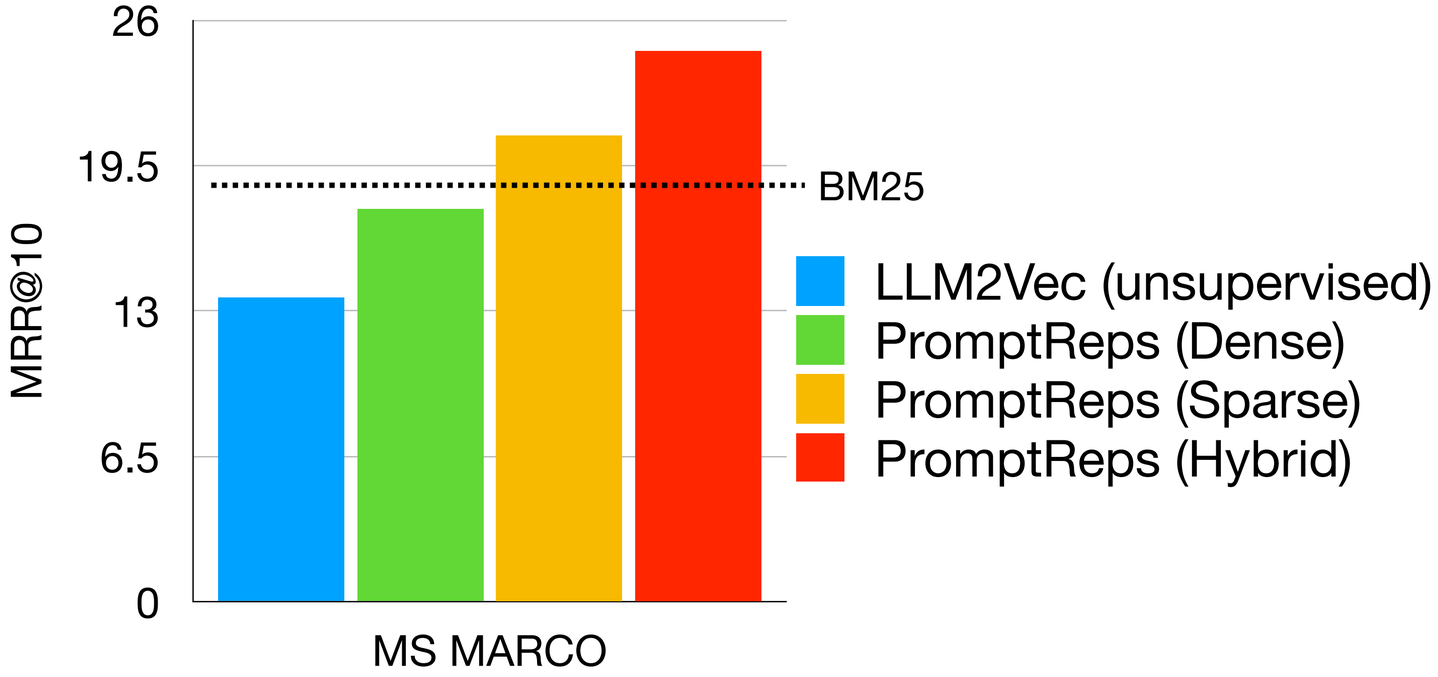
1. **prompt the LLM** to represent given text (doc/query) using **one word**
2. Use last **token hidden layer** to obtain a **dense** representation
3. Use **logits** associated with last token hidden layer to obtain a **sparse** representation over the LLM vocabulary
4. **Combine the two** representations to get a hybrid retriever



<System> *You are an AI assistant that can understand human language.*
<User> *Passage: "***[text]***". Use one word to represent the passage in a retrieval task. Make sure your word is in lowercase.*
<Assistant> *The word is: "*

Zhuang, S., Ma, X., Koopman, B., Lin, J. and Zuccon, G., 2024. PromptReps: Prompting Large Language Models to Generate Dense and Sparse Representations for Zero-Shot Document Retrieval. *arXiv preprint arXiv:2404.18424*.

PromptReps

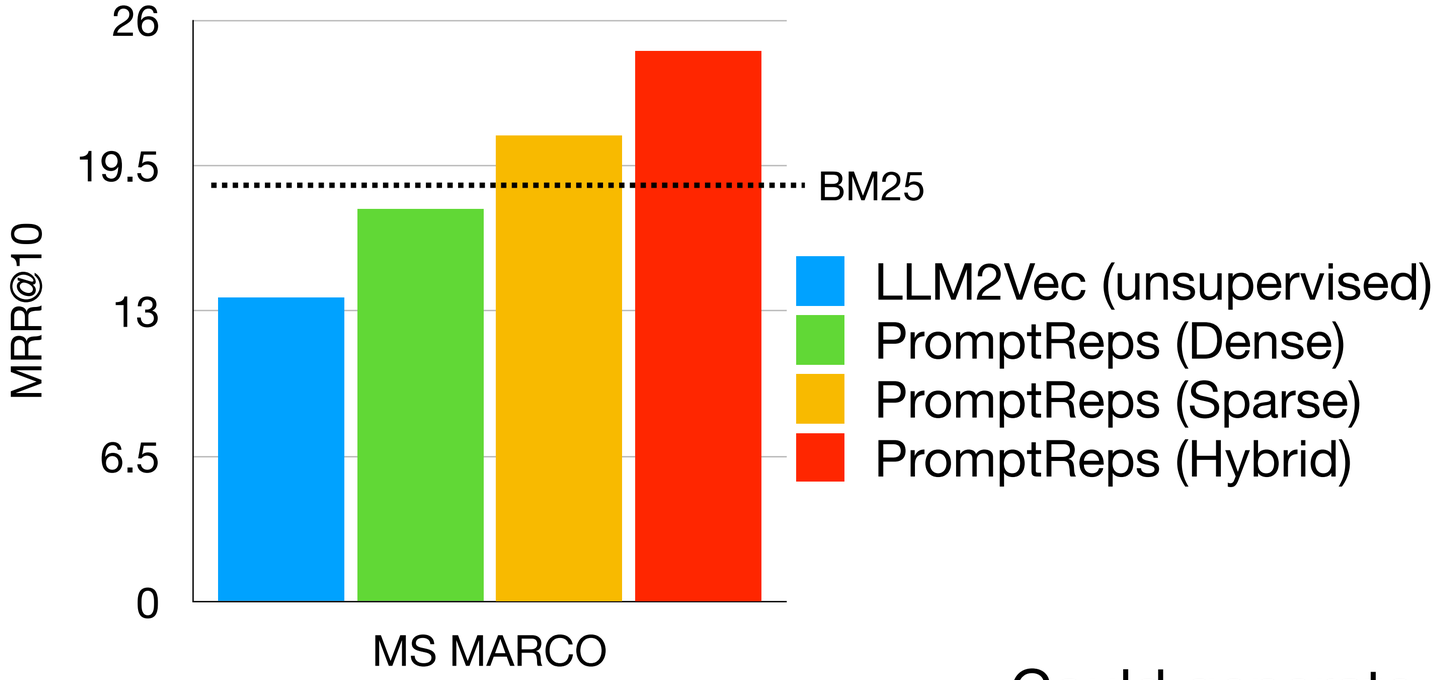


Combining Dense & Sparse representation allows effective zero-shot retrieval (no training of representations)

[LLM2Vec baseline here trained unsupervised with SimCSE]

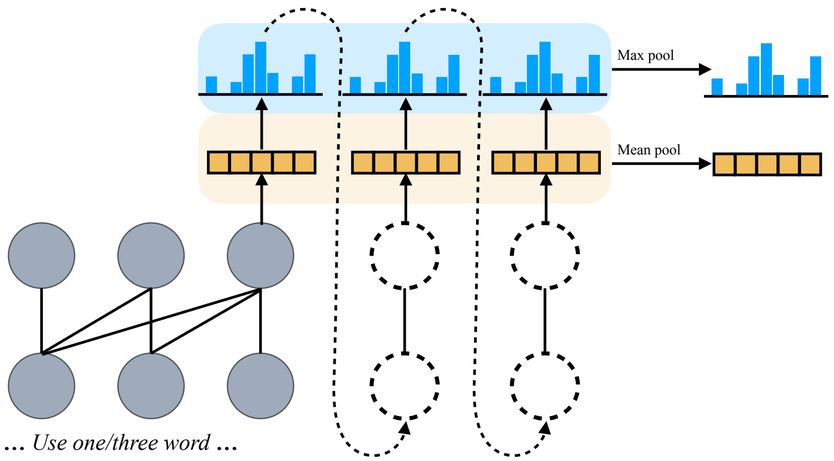
Zhuang, S., Ma, X., Koopman, B., Lin, J. and Zuccon, G., 2024. PromptReps: Prompting Large Language Models to Generate Dense and Sparse Representations for Zero-Shot Document Retrieval. *arXiv preprint arXiv:2404.18424*.

PromptReps



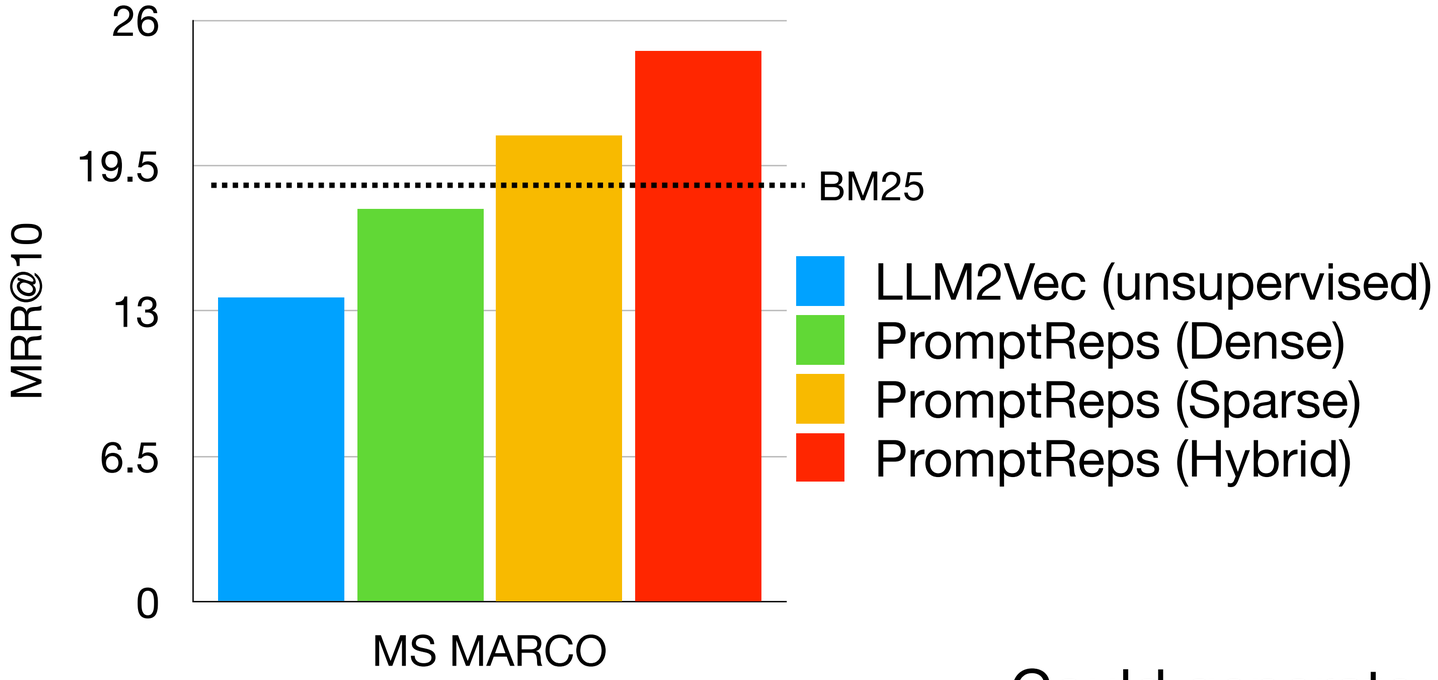
Combining Dense & Sparse representation allows effective zero-shot retrieval (no training of representations)

[LLM2Vec baseline here trained unsupervised with SimCSE]



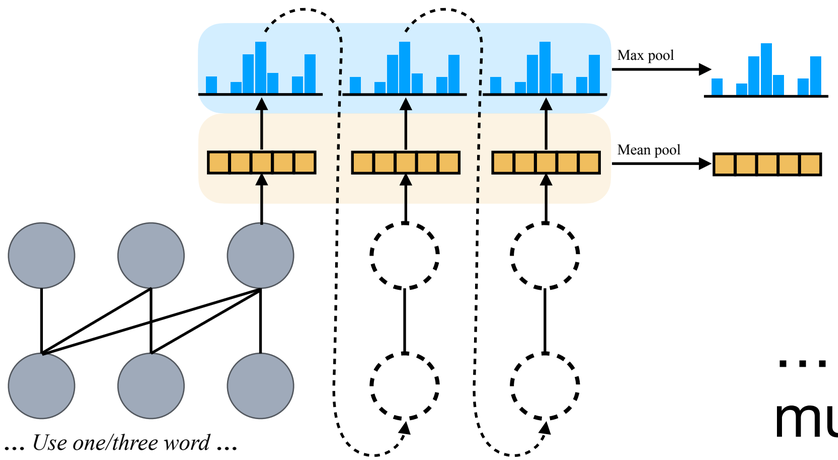
Could generate more words…

PromptReps



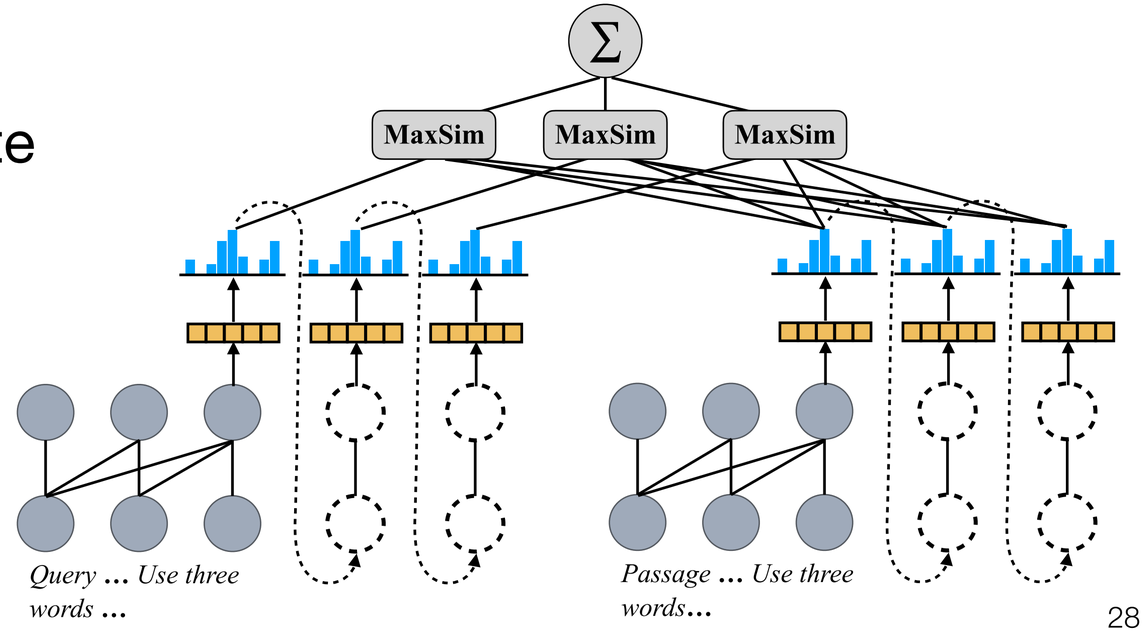
Combining Dense & Sparse representation allows effective zero-shot retrieval (no training of representations)

[LLM2Vec baseline here trained unsupervised with SimCSE]



Could generate more words…

… and use multiple tokens



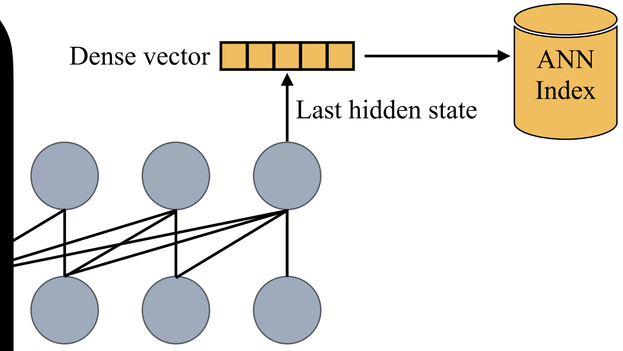
Key Intuitions

Can we ask LLMs to:

- **Retrieve:** generate repres
  - Then we can use repres
    create embeddings and

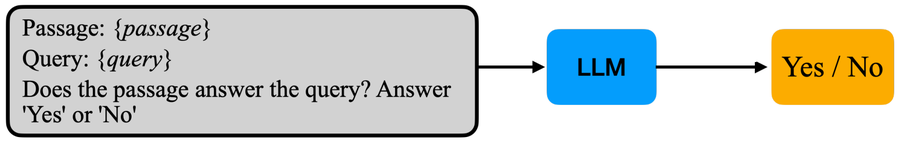
Plan:

1) Four "families" of LLM rankers
2) On prompt variations
3) On prompt optimisation



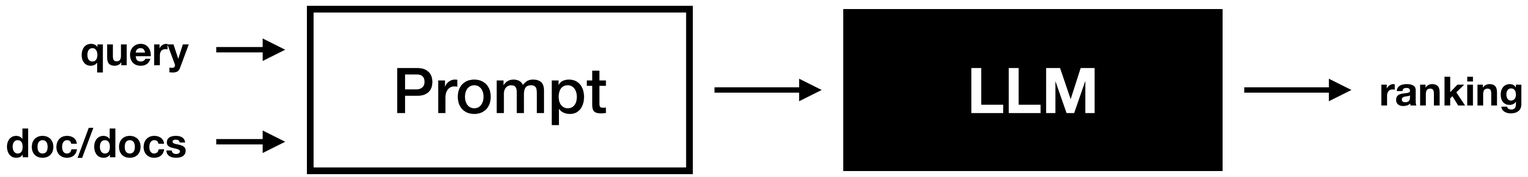
*You are an AI assistant that can understand human language. assage: "**[text]**". Use one word to represent the passage in a retrieval task. Make sure your word is in lowercase.*
***<Assistant>** The word is: "*

- **Ranking:** tell us the relevance of a document to a query?
  - Then we can use this indication of relevance (or relative relevance of n documents) to rank documents for the query



Idea: devise prompts/instructions to tell them LLM how to perform these tasks effectively

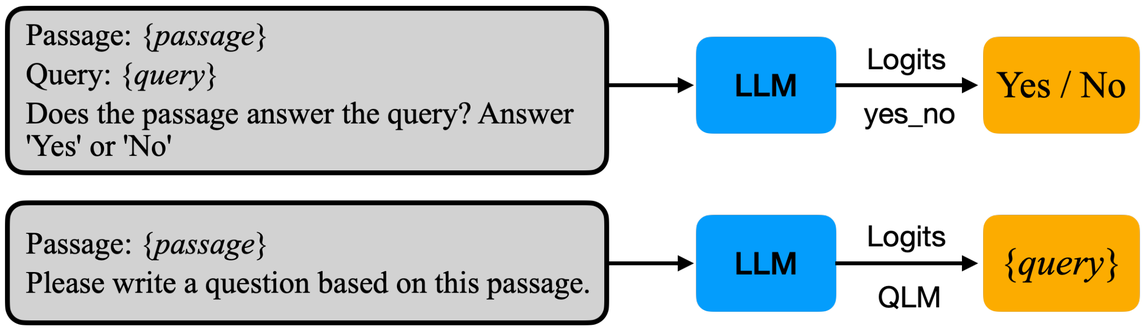
LLMs as Rankers



- Four main families, characterised by how documents are passed in the prompt and how relevance of document to query is determined
- All are “zero-shot”: i.e. once you obtained the pre-trained, instruction tuned LLM, no need to do contrastive training

LLMs as Rankers: Pointwise

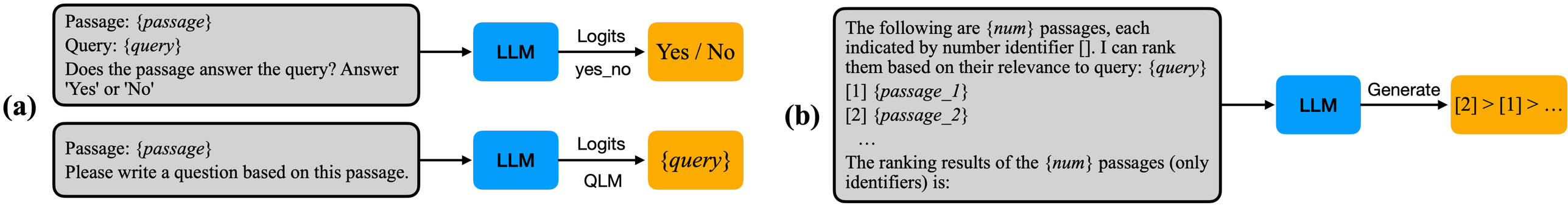
(a)



Liang, P., Bommasani, R., Lee, T., Tsipras, D., Soylu, D., Yasunaga, M., Zhang, Y., Narayanan, D., Wu, Y., Kumar, A. and Newman, B., 2022. Holistic evaluation of language models. *arXiv preprint arXiv:2211.09110*.

Sachan, D.S., Lewis, M., Joshi, M., Aghajanyan, A., Yih, W.T., Pineau, J. and Zettlemoyer, L., 2022. Improving passage retrieval with zero-shot question generation. *arXiv preprint arXiv:2204.07496*.

LLMs as Rankers: Listwise

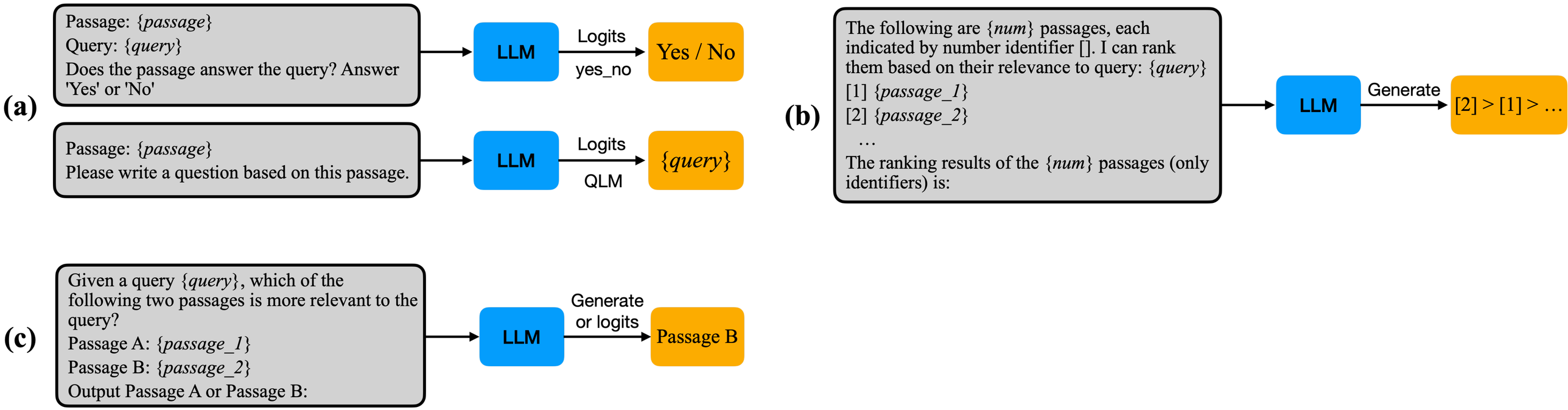


Ma, X., Zhang, X., Pradeep, R. and Lin, J., 2023. Zero-Shot Listwise Document Reranking with a Large Language Model. *arXiv preprint arXiv:2305.02156*.

Pradeep, R., Sharifymoghaddam, S. and Lin, J., 2023. RankVicuna: Zero-Shot Listwise Document Reranking with Open-Source Large Language Models. *arXiv preprint arXiv:2309.15088*.

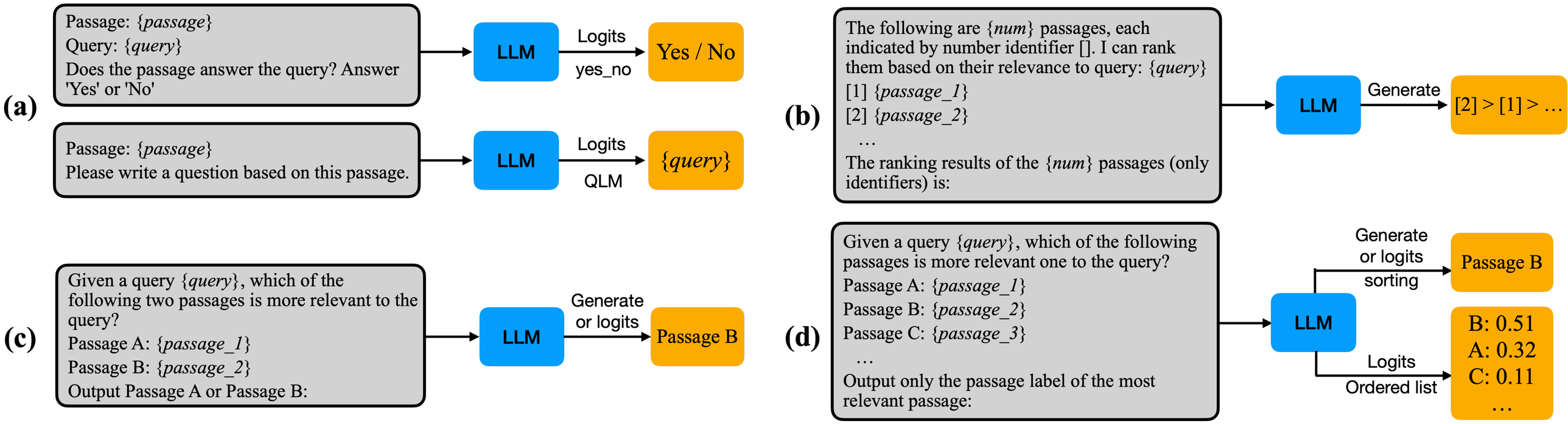
Sun, W., Yan, L., Ma, X., Ren, P., Yin, D. and Ren, Z., 2023. Is ChatGPT Good at Search? Investigating Large Language Models as Re-Ranking Agent. *arXiv preprint arXiv:2304.09542*.

LLMs as Rankers: Pairwise



Qin, Z., Jagerman, R., Hui, K., Zhuang, H., Wu, J., Shen, J., Liu, T., Liu, J., Metzler, D., Wang, X. and Bendersky, M., 2023. Large language models are effective text rankers with pairwise ranking prompting. *arXiv preprint arXiv:2306.17563*.

LLMs as Rankers: Setwise

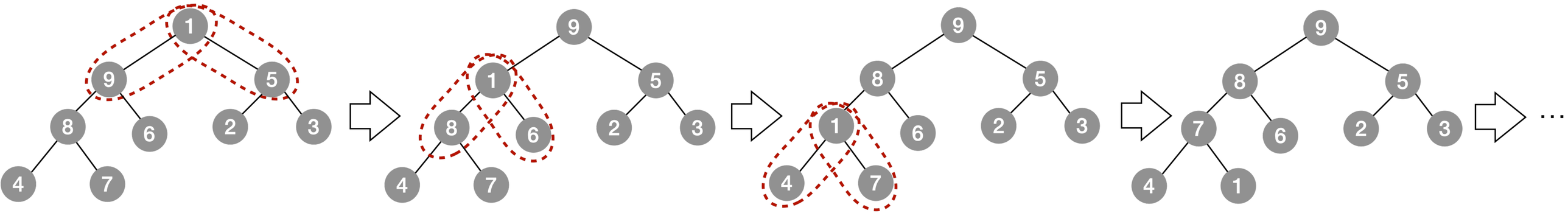


Zhuang, S., Zhuang, H., Koopman, B. and Zuccon, G., 2023. A Setwise Approach for Effective and Highly Efficient Zero-shot Ranking with Large Language Models. *arXiv preprint arXiv:2310.09497*.

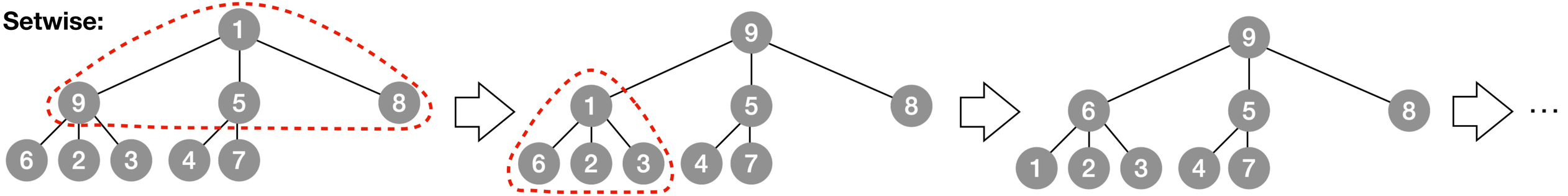
LLMs as Rankers: Setwise

- Setwise offers two advantages
  - Compared to listwise: It can rely on logits rather than generation: faster
  - Compared to pairwise: it requires less comparisons (i.e. inferences with LLM)

Pairwise:



Setwise:



LLMs as Rankers

Method	NDCG@10	TREC DL 2019	TREC DL 2020
setwise.bubblesort	0.6800	$4.62	$4.67
pairwise.allpair	0.6783	$90.59	$90.6
setwise.heapsort	0.6743	$1.28	$1.27
pointwise.yes_no	0.6398	$0.48	$0.49
listwise.generation	0.5929	$3.49	$3.75
pointwise.qlm	0.5343	$0.46	$0.46

Zhuang, S., Zhuang, H., Koopman, B. and Zuccon, G., 2023. A Setwise Approach for Effective and Highly Efficient Zero-shot Ranking with Large Language Models. *arXiv preprint arXiv:2310.09497*.

LLMs as Rankers

Method	NDCG@10	TREC DL 2019	TREC DL 2020
pairwise.heapsort	0.6800	$3.39	$3.4
setwise.bubblesort	0.6800	$4.62	$4.67
pairwise.allpair	0.6783	$90.59	$90.6
listwise.likelihood	0.6745	$2.83	$2.86
setwise.heapsort	0.6743	$1.28	$1.27
pairwise.bubblesort	0.6550	$12.28	$11.89
pointwise.yes_no	0.6398	$0.48	$0.49
listwise.generation	0.5929	$3.49	$3.75
pointwise.qlm	0.5343	$0.46	$0.46

0.8% loss in nDCG

62% reduction in cost

Zhuang, S., Zhuang, H., Koopman, B. and Zuccon, G., 2023. A Setwise Approach for Effective and Highly Efficient Zero-shot Ranking with Large Language Models. *arXiv preprint arXiv:2310.09497*.

Problems with ListWise

- LLM has a **maximum context(prompt) length**, which limits the number of passages in a list
- Because ranking needs to be broken into many sublists, **listwise requires many inferences** -> **high latency**
  - Also **generating** the string with the **ranking** can be **difficult & slow**: LLM may output wrong format, requiring to re-generate, or produces additional content (e.g. explanation), making generation slow

ie lab

Liu, Q., Wang, B., Wang, N. and Mao, J., 2024. Leveraging Passage Embeddings for Efficient Listwise Reranking with Large Language Models. *arXiv preprint arXiv:2406.14848*.

PE-Rank: passage compression

- LLM has a **maximum context(prompt) length**, which limits the number of passages in a list
- Because ranking needs to be broken into many sublists, **listwise requires many inferences** -> **high latency**
  - Also **generating** the string with the **ranking** can be **difficult & slow**: LLM may output wrong format, requiring to re-generate, or produces additional content (e.g. explanation), making generation slow
- PE-Rank:
  1. Treat passage as a single special token
  2. Use the passage embedding as token representation

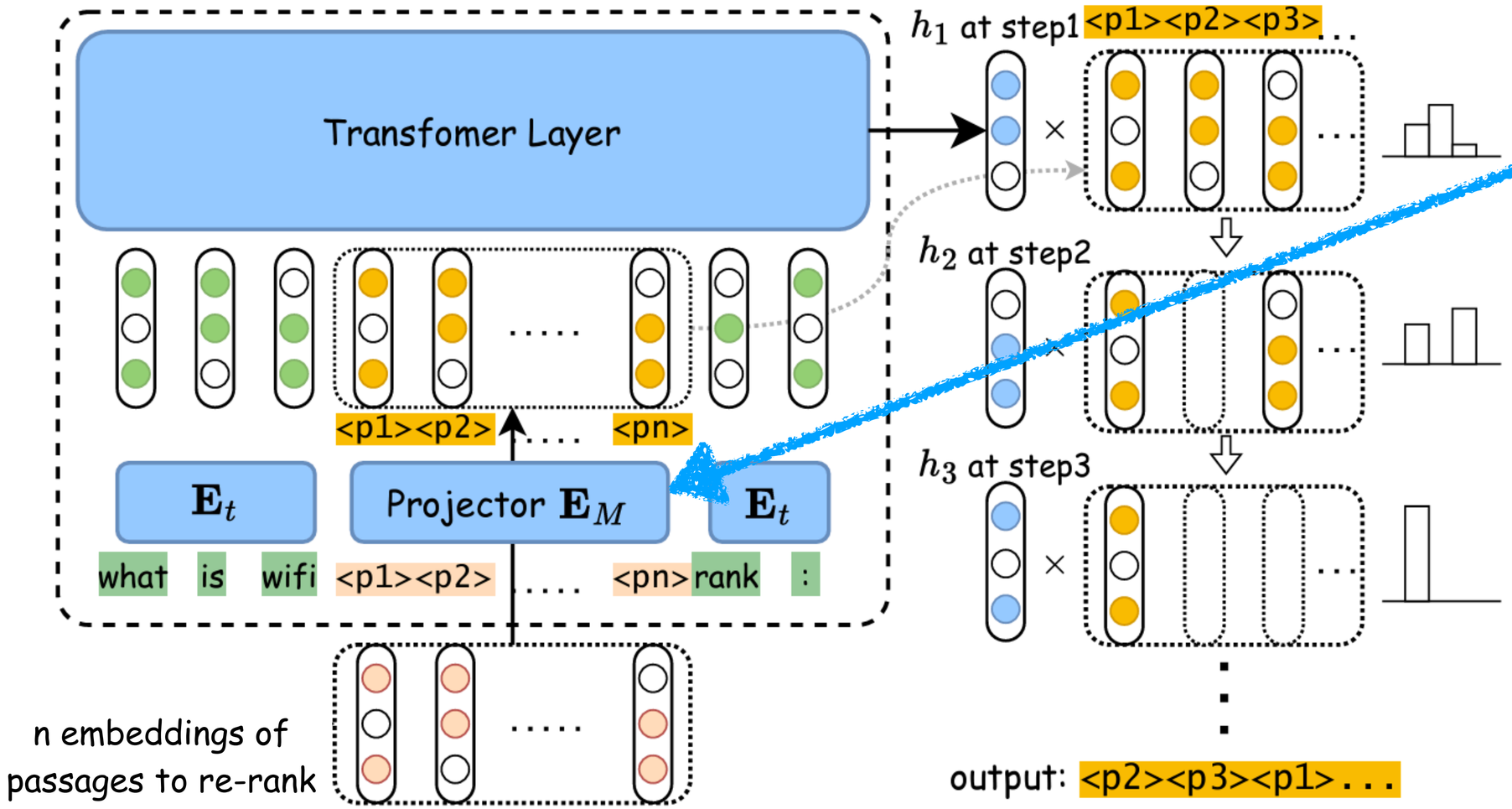
     Reduce input length
  3. Dynamic constrain of decoding space for special tokens

     Accelerate decoding process

ie lab

Liu, Q., Wang, B., Wang, N. and Mao, J., 2024. Leveraging Passage Embeddings for Efficient Listwise Reranking with Large Language Models. *arXiv preprint arXiv:2406.14848*.

PE-Rank: passage compression

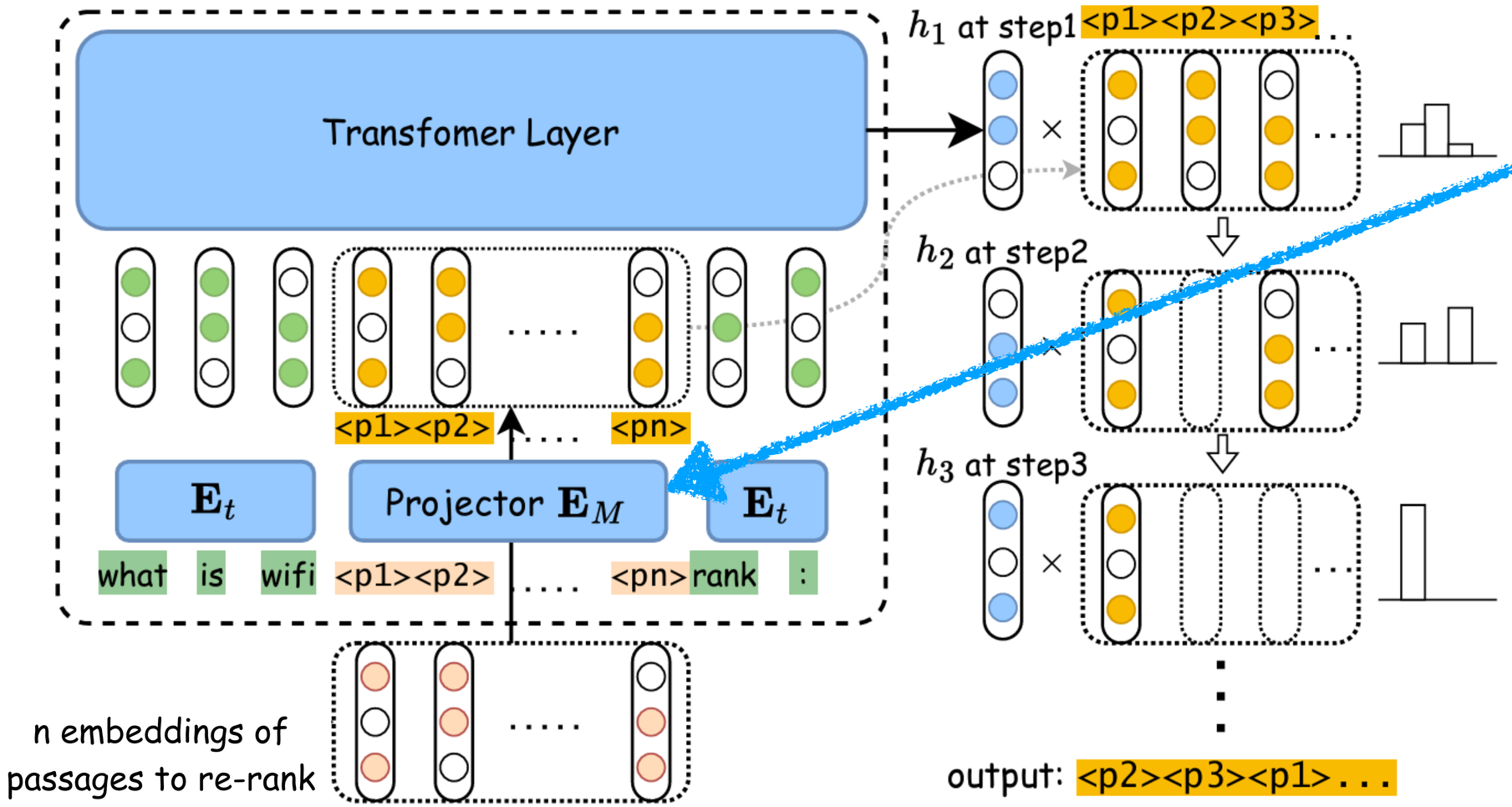


This is needed because of dimensional and distributional differences between passage embeddings and LLM token embeddings

EM is a 2-layer MLP that acts as a learned mapping function

Liu, Q., Wang, B., Wang, N. and Mao, J., 2024. Leveraging Passage Embeddings for Efficient Listwise Reranking with Large Language Models. *arXiv preprint arXiv:2406.14848*.

PE-Rank: passage compression



Training: text reconstruction task

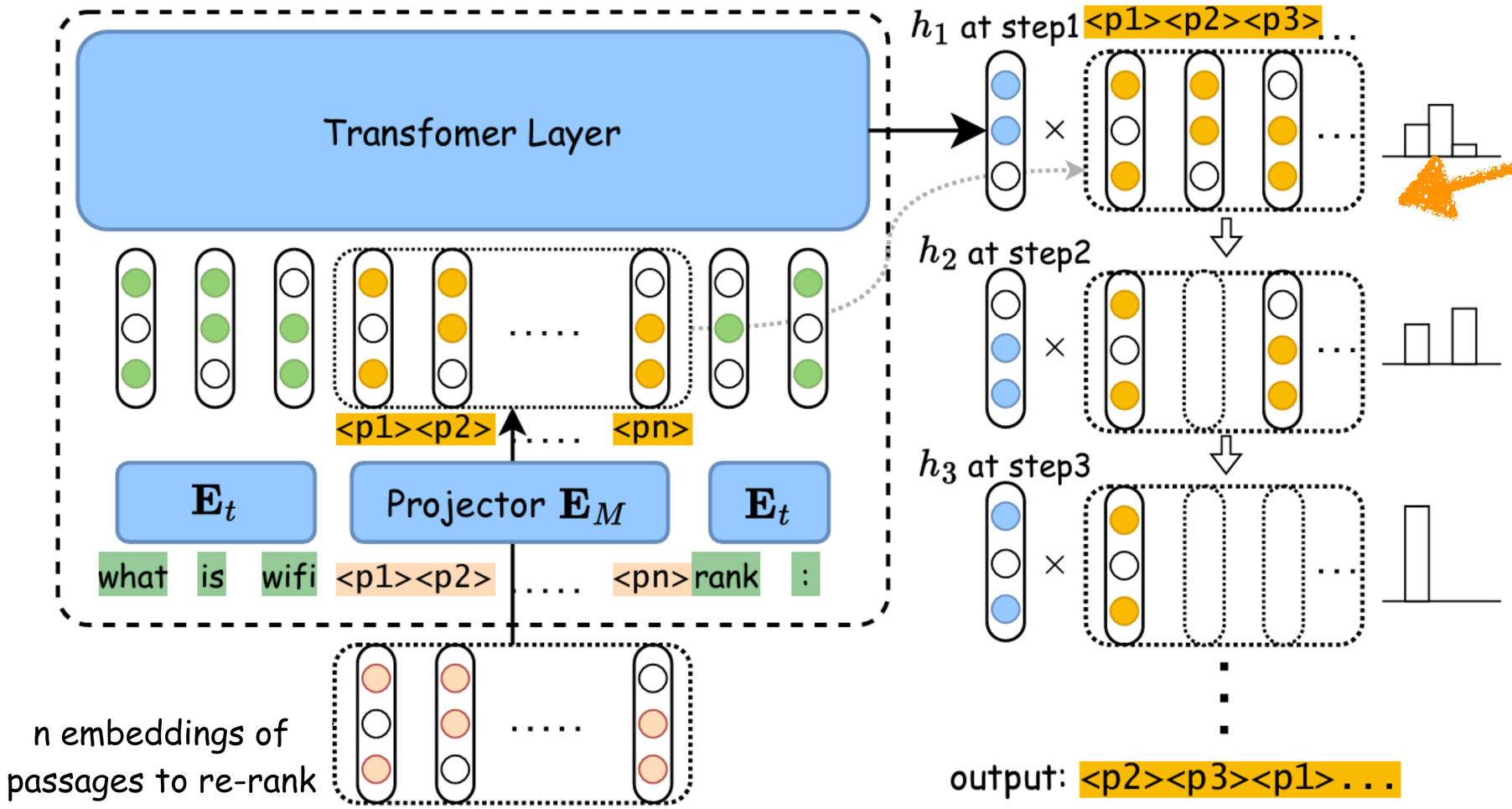
1. encode t, pass to MLP
2. Take transformed embedding, pass it to LLM
3. Prompt LLM to reconstruct text

(Only MLP is trained, LLM & embeddings are frozen)

Training data: Wikipedia

Liu, Q., Wang, B., Wang, N. and Mao, J., 2024. Leveraging Passage Embeddings for Efficient Listwise Reranking with Large Language Models. *arXiv preprint arXiv:2406.14848*.

PE-Rank: passage compression



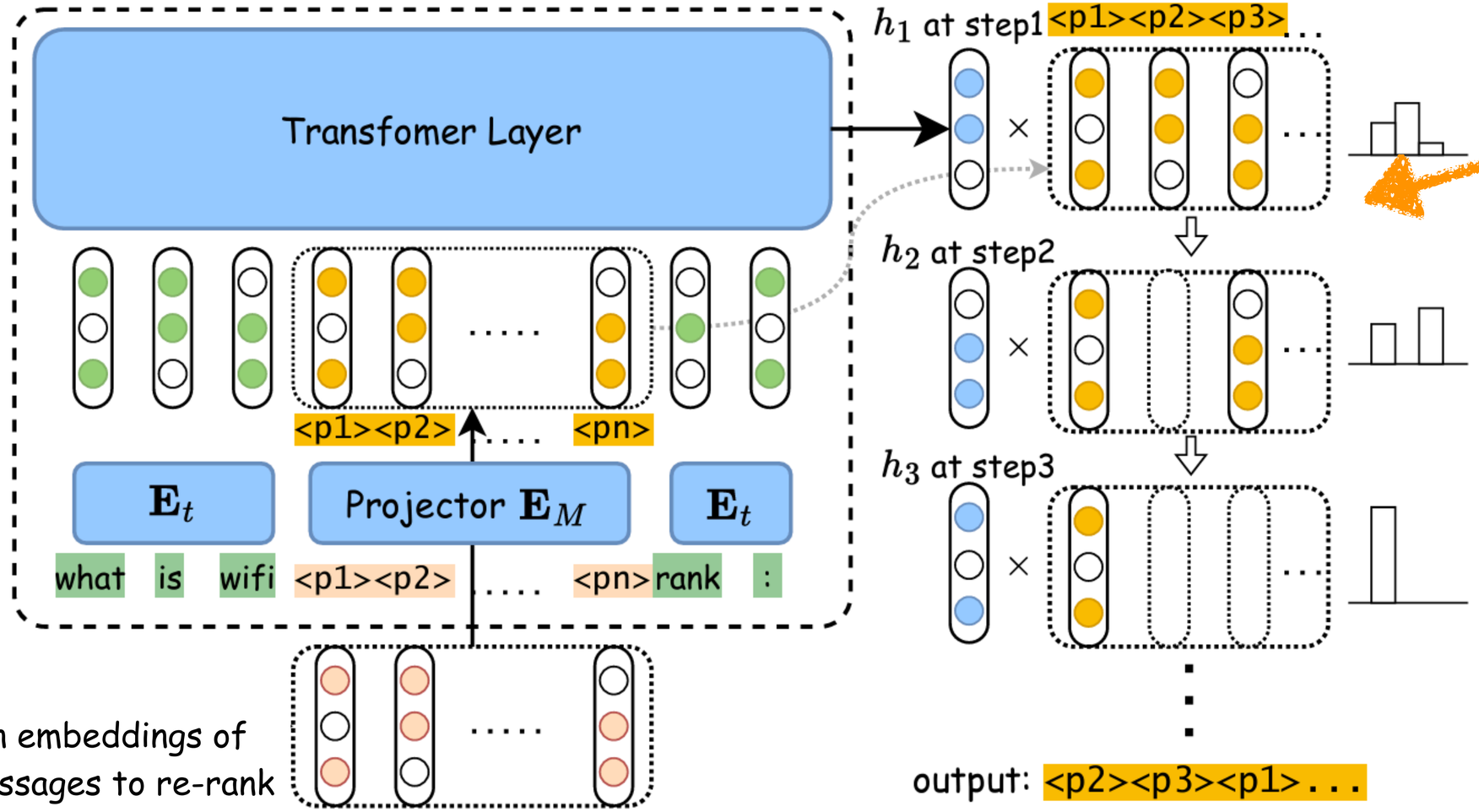
Constrain generation to only special tokens

* smaller vocabulary -> faster inference

* remove already generated tokens -> further inference acceleration

Liu, Q., Wang, B., Wang, N. and Mao, J., 2024. Leveraging Passage Embeddings for Efficient Listwise Reranking with Large Language Models. *arXiv preprint arXiv:2406.14848*.

PE-Rank: passage compression



Training: sequential ranking learning process — at each step, provide previous decoded ranking, then maximise probability of generating next most relevant passage (i.e. ListMLE)

+ content loss + distillation loss

Training data: MS MARCO

PE-Rank: passage compression

Common listwise

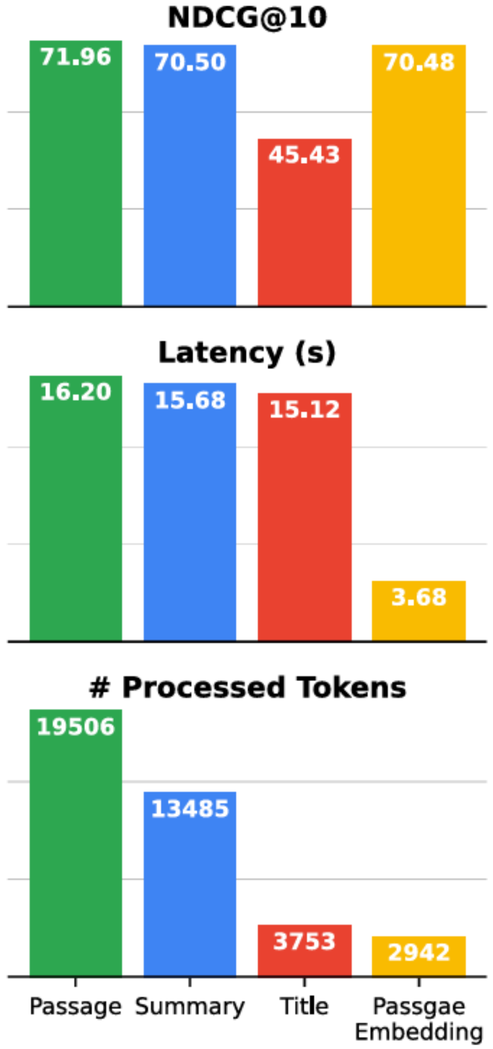
The following are passages related to query #{query}.
Passage 1: #{passage 1}
...
Rank these passages based on their relevance to the query.

[2] > [3] > [1] ...

PE-Rank

The following are passages related to query #{query}, each with a special token representing the passage enclosed in [].
Passage 1: [<p1>]
...
Rank these passages based on their relevance to the query.

<p2><p3><p1> ...



PE-Rank: passage compression

Common listwise

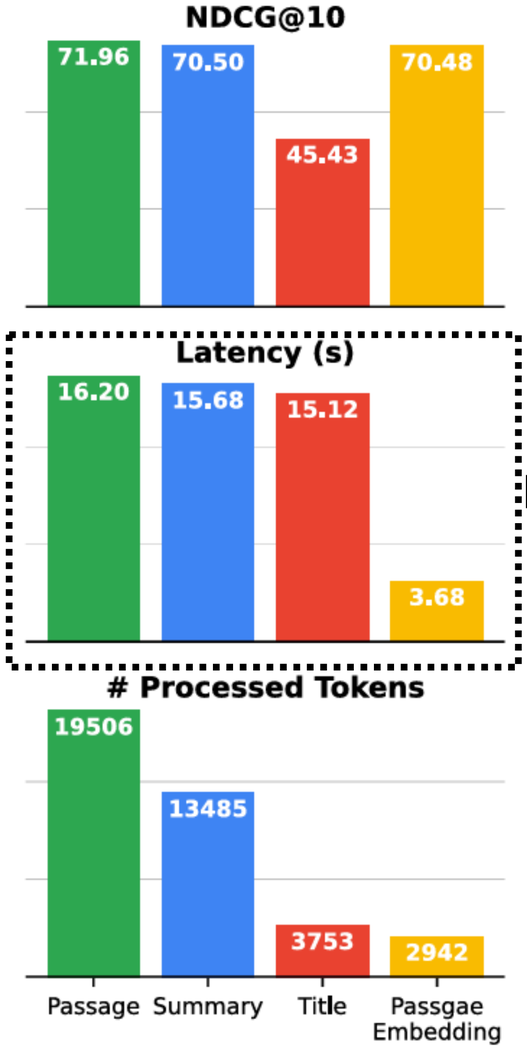
The following are passages related to query #{query}.
Passage 1: #{passage 1}
...
Rank these passages based on their relevance to the query.

[2] > [3] > [1] ...

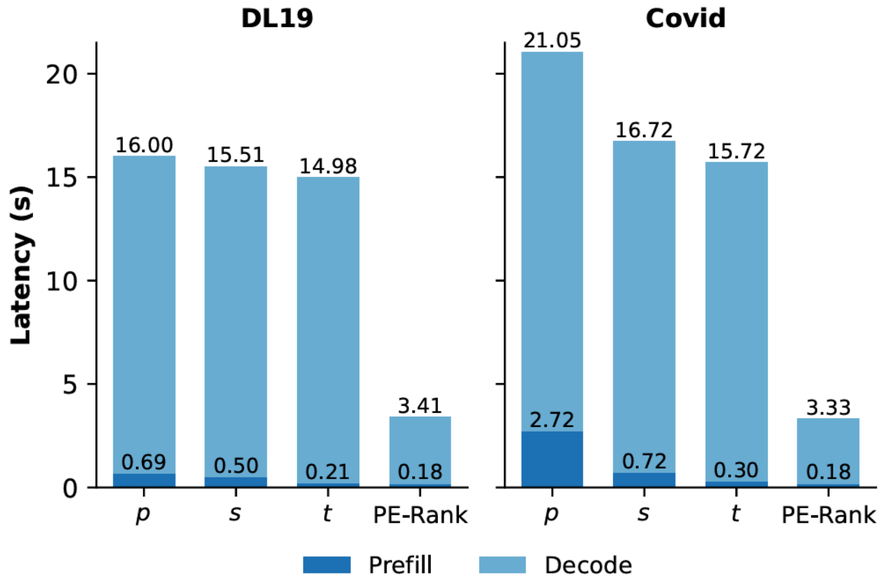
PE-Rank

The following are passages related to query #{query}, each with a special token representing the passage enclosed in [].
Passage 1: [<p1>]
...
Rank these passages based on their relevance to the query.

<p2><p3><p1> ...



Most latency advantages arise from decoding, little gain in profiling phase



Impact of Prompt Variations on LLM Rankers

- Prompts proposed for different LLM ranker methods vary largely
- Not just in terms of instruction for ranking, but also for (unrelated) additional wordings, e.g. role playing, and ordering of components (e.g. passage first, then query — or vice versa?)
- What are the effects of prompt wordings on methods? What makes a good prompt?

Method	Prompt
PRP, Qin et al.	Passage: {text} Query: {query} Does the passage answer the query?
RankGPT, Sun et al.	You are RankGPT, an intelligent assistant that can rank passages based on their relevancy to the query. I will provide you with num passages, each indicated by number identifier []. Rank the passages based on their relevance to query: {query}. {PASSAGES} Search Query: {query}. Rank the num passages above based on their relevance to the search query. The passages should be listed in descending order using identifiers. The most relevant passages should be listed first. The output format should be [] > [], e.g., [1] > [2]. Only response the ranking results, do not say any word or explain.

Role Playing

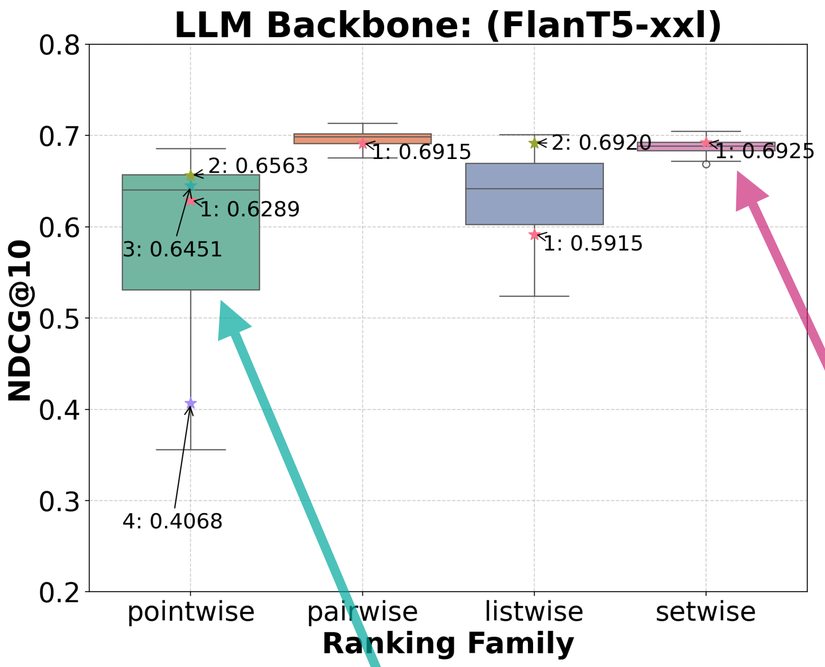
Formatting Instr.

Restriction on Output

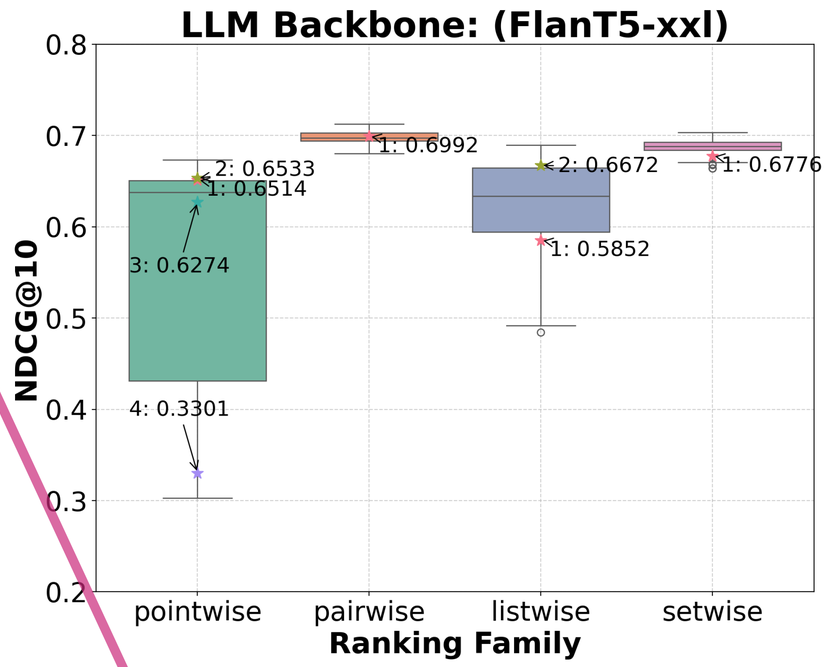
Sun, S., Zhuang, S., Wang, S. and Zuccon, G., 2024. An Investigation of Prompt Variations for Zero-shot LLM-based Rankers. *arXiv preprint arXiv:2406.14117*.

Impact of Prompt Variations on LLM Rankers

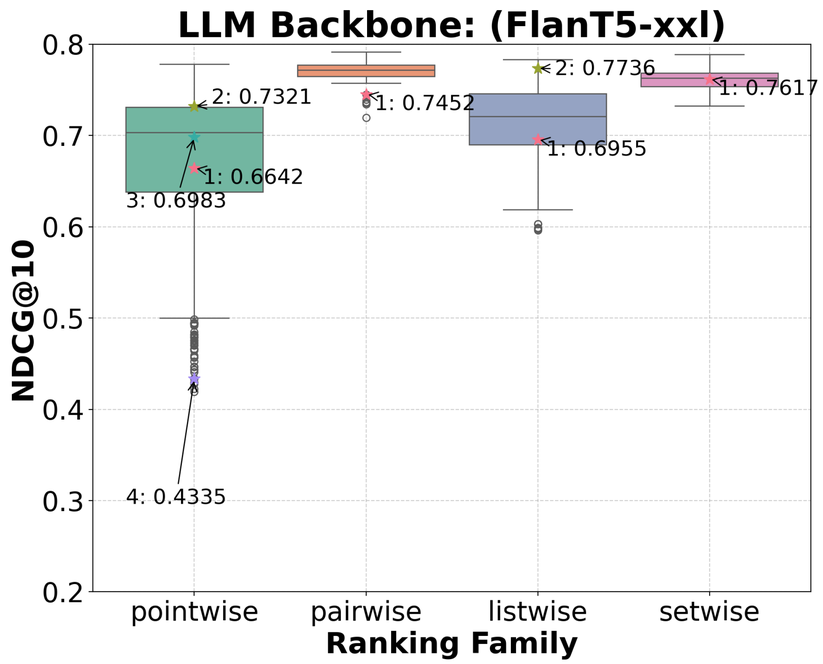
TREC DL 2019



TREC DL 2020



COVID (BEIR)



Pointwise, listwise often display large variations of effectiveness across different prompts

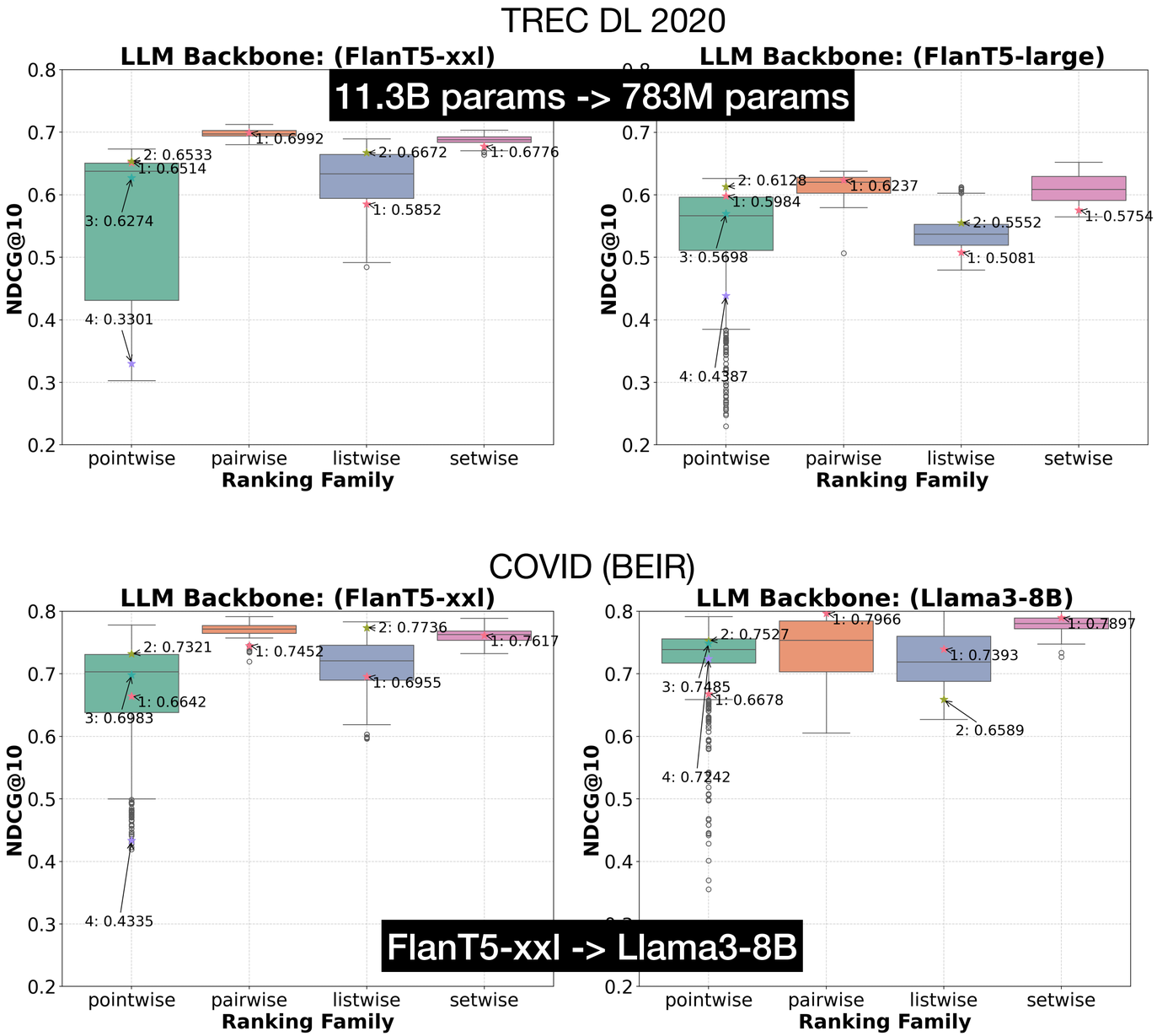
Pairwise, set wise very robust across different prompts

For all approaches, there are better prompts than the one in the original paper

ie lab

Sun, S., Zhuang, S., Wang, S. and Zuccon, G., 2024. An Investigation of Prompt Variations for Zero-shot LLM-based Rankers. *arXiv preprint arXiv:2406.14117*.

Impact of Prompt Variations on LLM Rankers



But specific behaviour depends also on

- Size of LLM
- The actual LLM architecture/ training

From Manual Prompting to Automatic Prompt Engineering

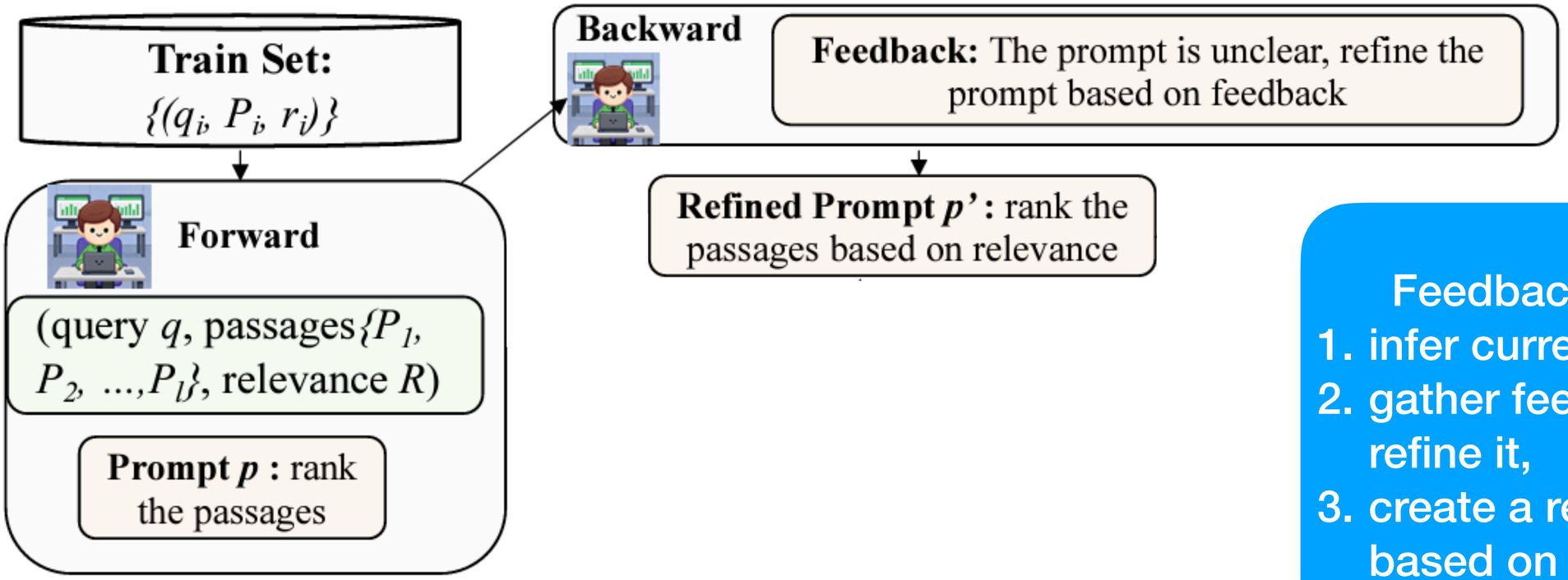
- LLM rankers **effectiveness varies** across prompt **wording**
- Can we **automatically identify** create effective **prompt**?
- Direct application of current LLM automatic prompt engineering challenging cause ranking task more complex than classification task:
    1. Requires comprehension of query and of relevance of passages
    2. input-output demonstrations for relevance ranking are more complex than those for language modeling.
        - Input: query, passage. Output: relevance — which is not unique to query-passage pair (others may have same relevance level

ie lab

The University Of Queensland AUSTRALIA

CREATE CHANGE

From Manual Prompting to Automatic Prompt Engineering: APEER

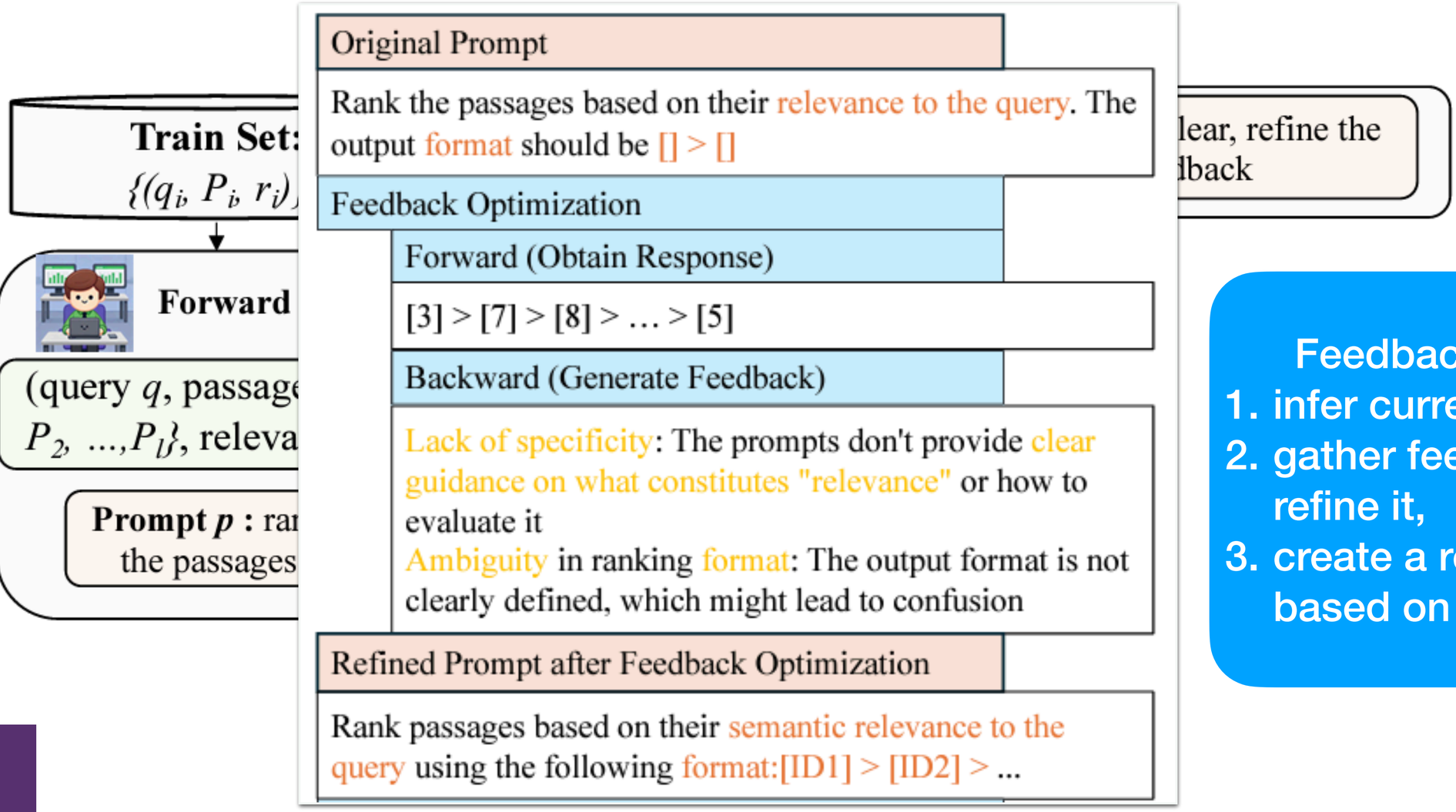


Feedback optimization:

1. infer current prompt,
2. gather feedback on how to refine it,
3. create a refined prompt based on feedback

Jin, C., Peng, H., Zhao, S., Wang, Z., Xu, W., Han, L., Zhao, J., Zhong, K., Rajasekaran, S. and Metaxas, D.N., 2024. APEER: Automatic Prompt Engineering Enhances Large Language Model Reranking. *arXiv preprint arXiv:2406.14449*.

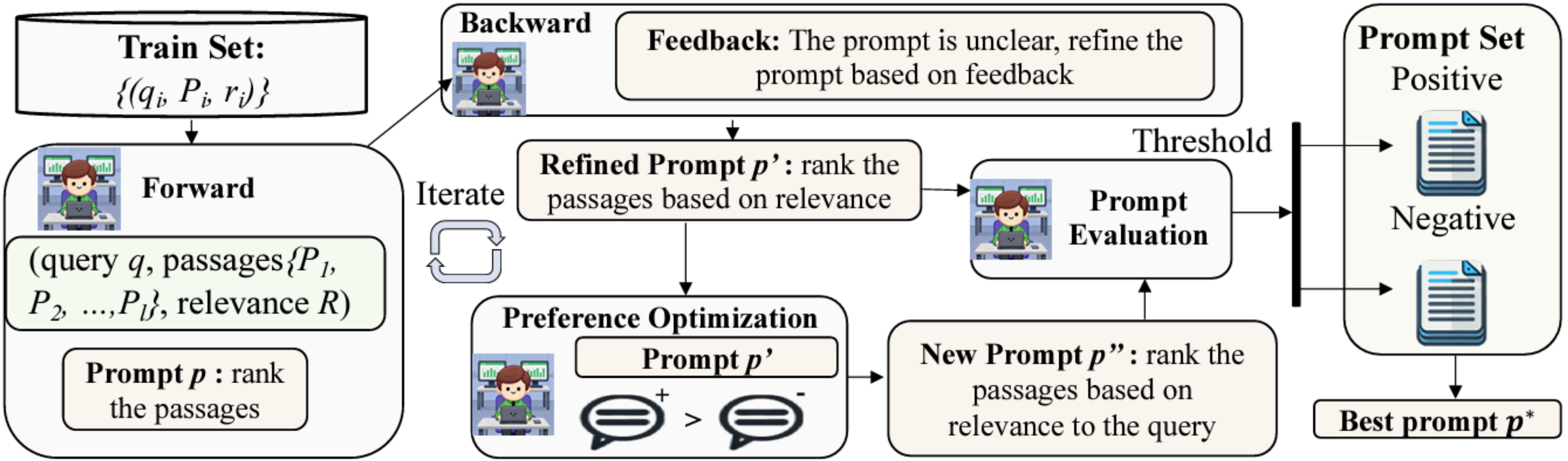
From Manual Prompting to Automatic Prompt Engineering: APEER



Feedback optimization:

1. infer current prompt,
2. gather feedback on how to refine it,
3. create a refined prompt based on feedback

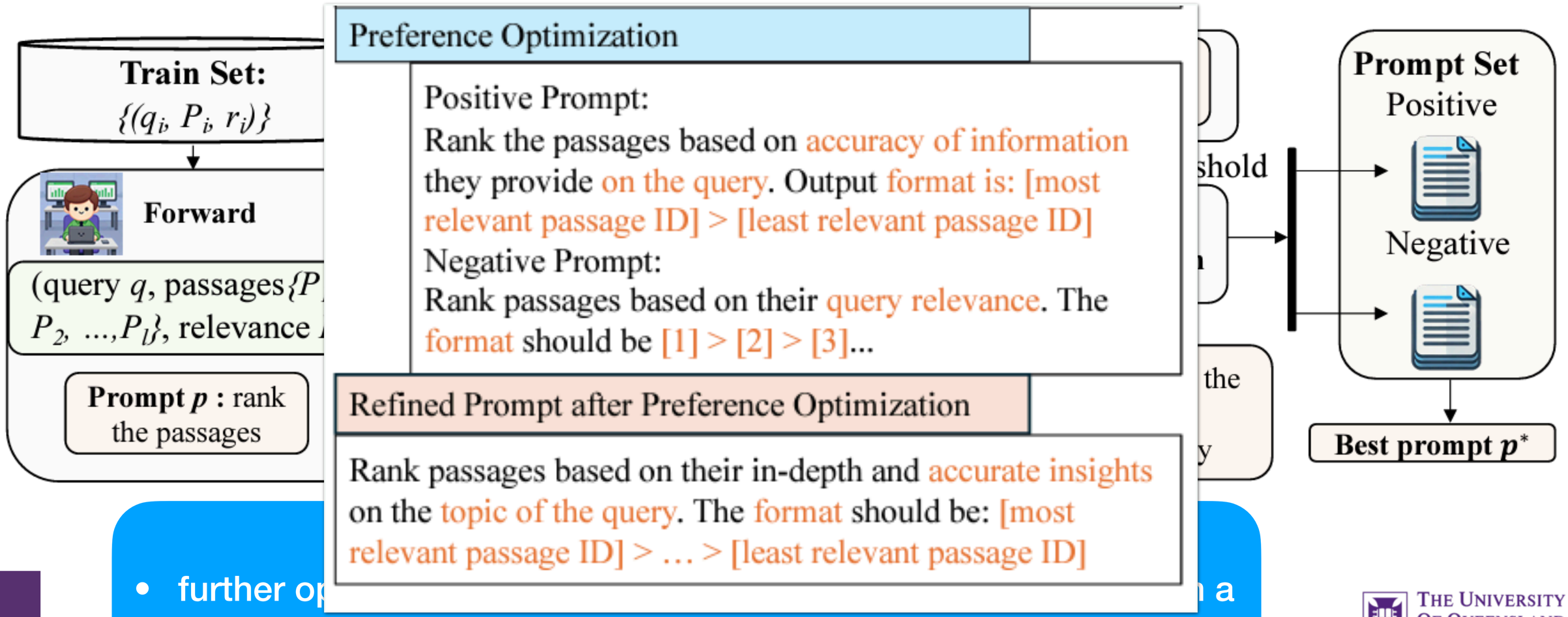
From Manual Prompting to Automatic Prompt Engineering: APEER



Preference optimization:

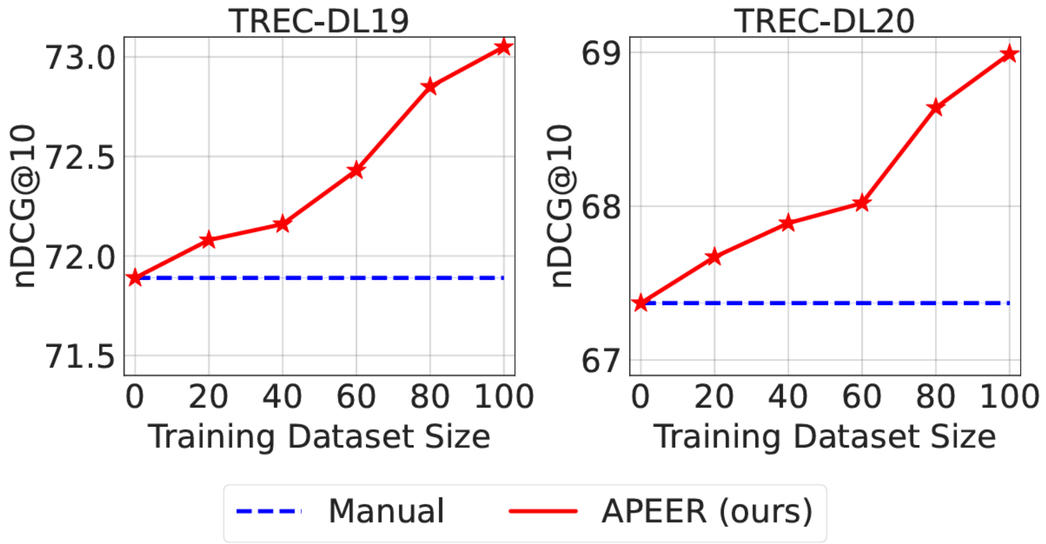
- further optimize refined prompt by learning preferences through a set of positive and negative prompt demonstrations

From Manual Prompting to Automatic Prompt Engineering: APEER



- further op… …n a set of positive and negative prompt demonstrations

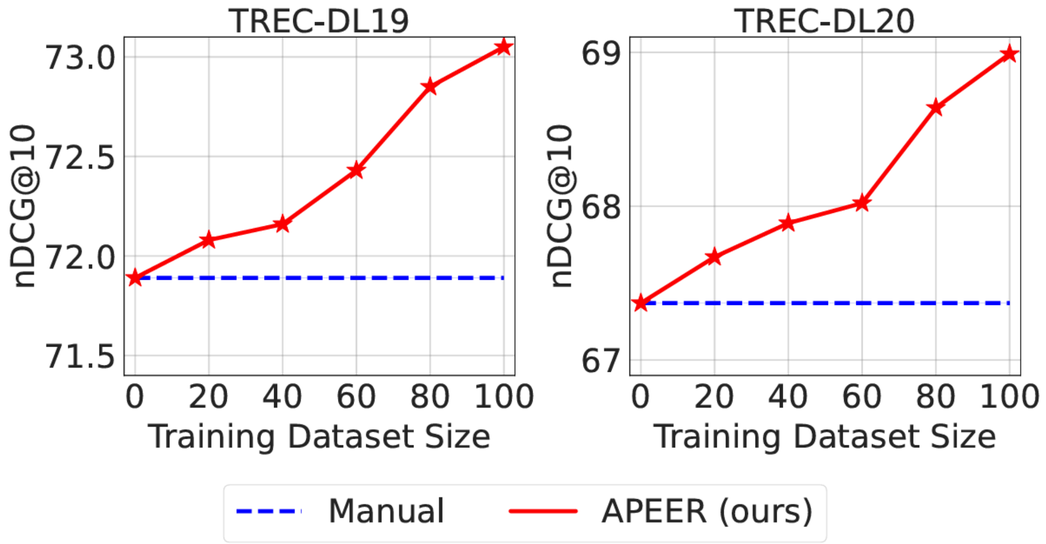
From Manual Prompting to Automatic Prompt Engineering: APEER

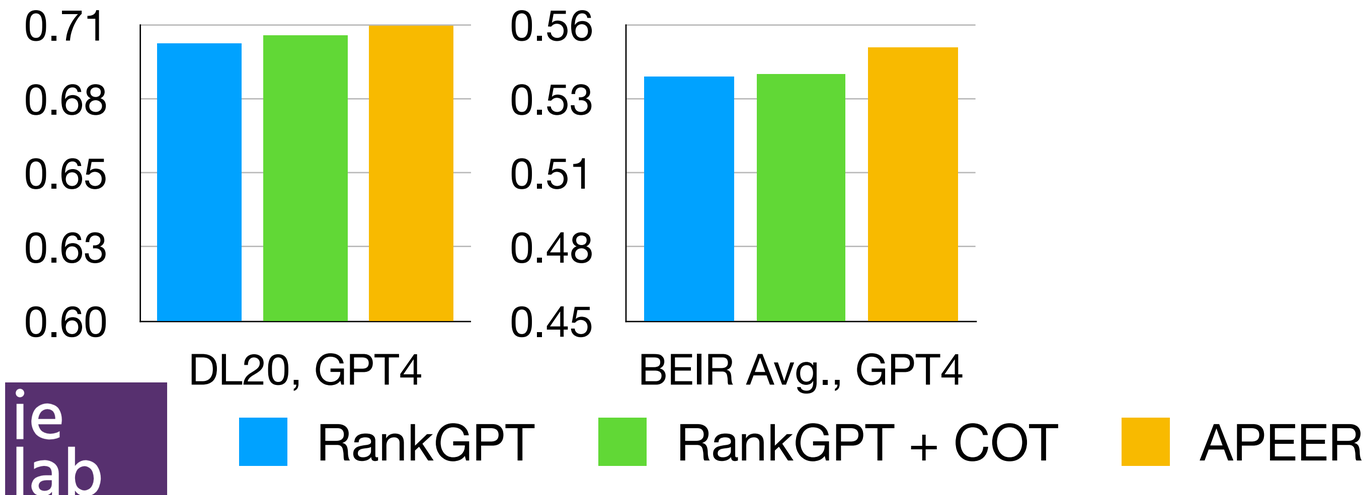


- **More training data** consistently creates **better and better prompts**
- At the cost of additional computational costs for training

Jin, C., Peng, H., Zhao, S., Wang, Z., Xu, W., Han, L., Zhao, J., Zhong, K., Rajasekaran, S. and Metaxas, D.N., 2024. APEER: Automatic Prompt Engineering Enhances Large Language Model Reranking. *arXiv preprint arXiv:2406.14449*.

From Manual Prompting to Automatic Prompt Engineering: APEER



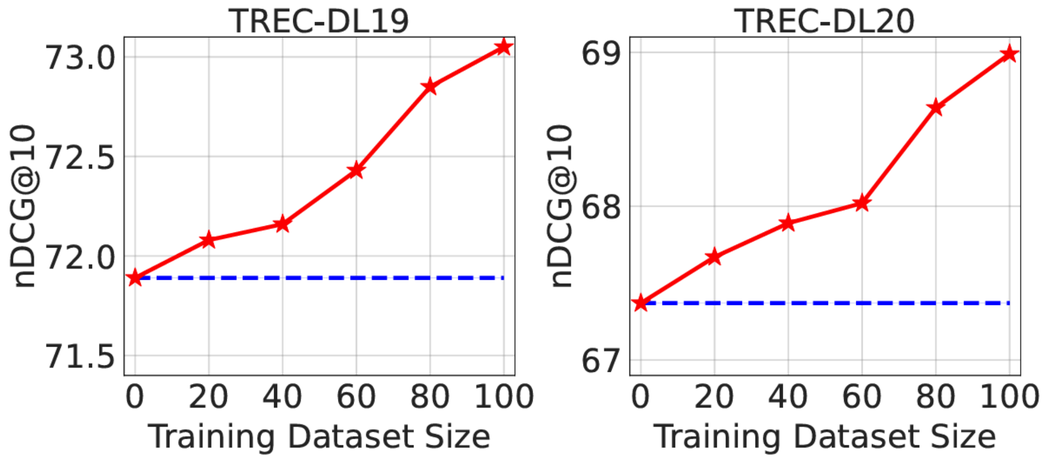


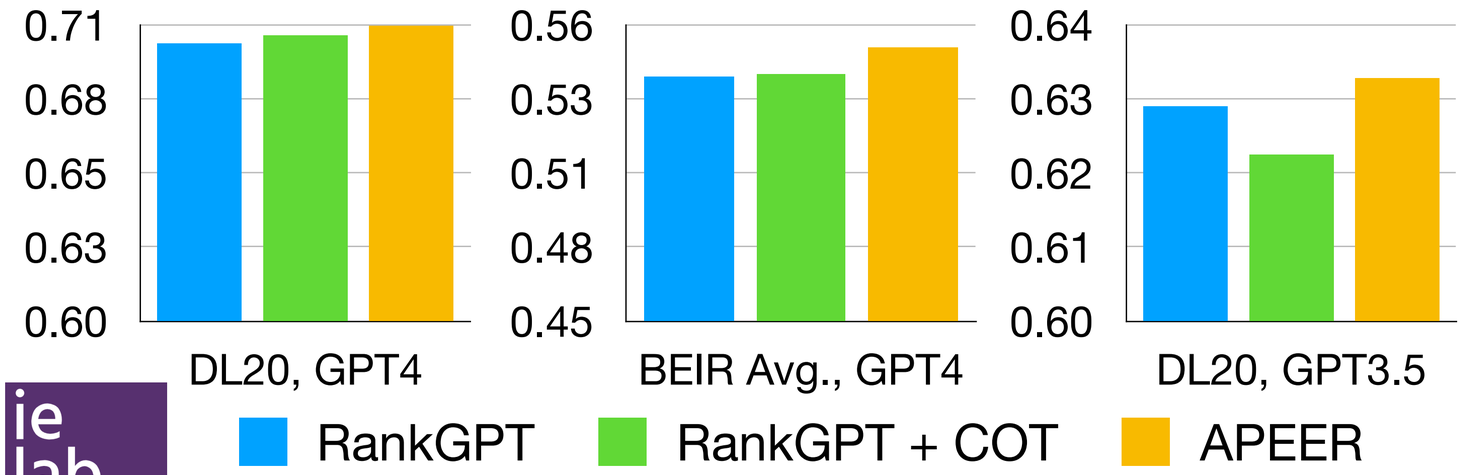
- **More training data** consistently creates **better and better prompts**
- At the cost of additional computational costs for training
- Prompts created with APEER trained on MS MARCO also **transfer** to **out of domain** datasets (BEIR)

ie lab

The University Of Queensland AUSTRALIA
CREATE CHANGE

From Manual Prompting to Automatic Prompt Engineering: APEER





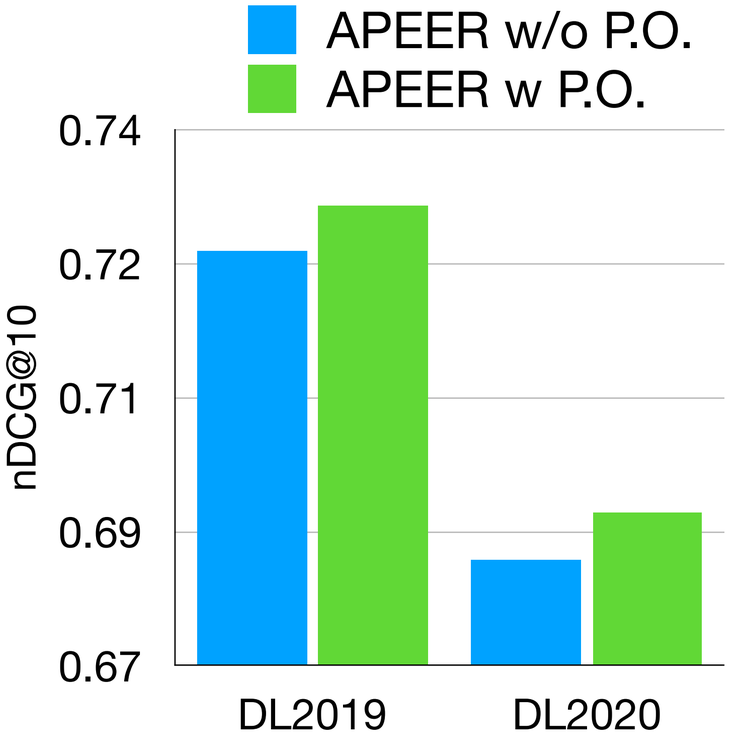
- More training data consistently creates better and better prompts
- At the cost of additional computational costs for training
- Prompts created with APEER trained on MS MARCO also **transfer** to **out of domain** datasets (BEIR)
- Prompts can also **transfer** between **models**,
  e.g. GPT4->GPT3.5; Qwen2 -> Llama3

ie lab

The University Of Queensland AUSTRALIA
CREATE CHANGE

More on Preference Optimisation Phase

- Catalog a collection of potential positive and negative responses within H_{pos} and H_{neg}
- Utilize 2t pairs of p_{pos},p_{neg} for demonstration
- Generate a new prompt, evaluate its performance on validation dataset relative to initial prompt, insert it in H_{pos} and H_{neg} accordingly
- Feedback Optimization: acts as local optimizer for current batch
- Preference Optimization: extends local optimization by globally aligning the local optimized prompts towards superior global prompts

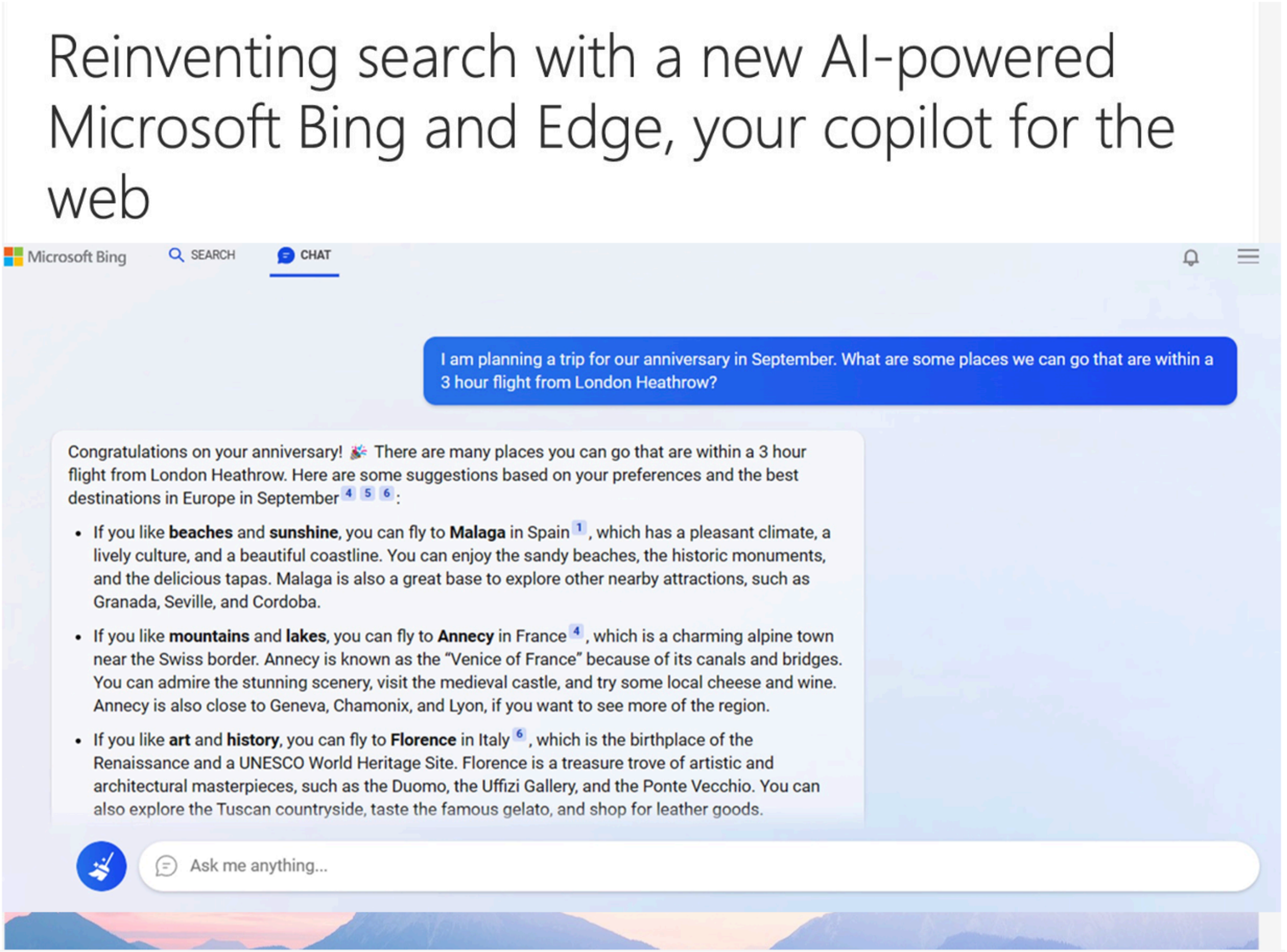


Preference Optimisation beneficial but gains minor (~0.8%)

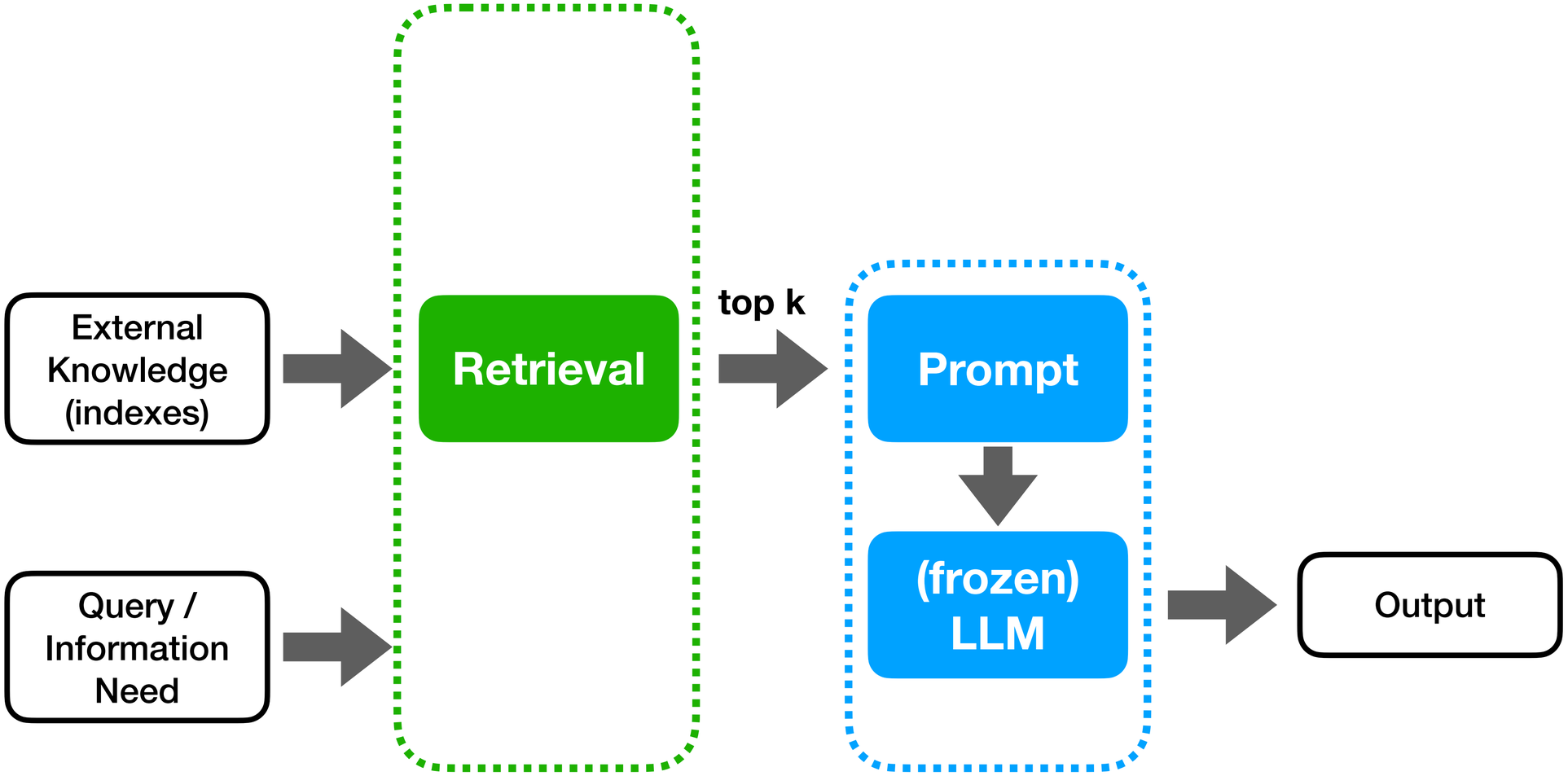
Jin, C., Peng, H., Zhao, S., Wang, Z., Xu, W., Han, L., Zhao, J., Zhong, K., Rajasekaran, S. and Metaxas, D.N., 2024. APEER: Automatic Prompt Engineering Enhances Large Language Model Reranking. *arXiv preprint arXiv:2406.14449*.

Part 2: Retrieval Augmented Generation

Retrieval Augmented Generation



The RAG Pipeline

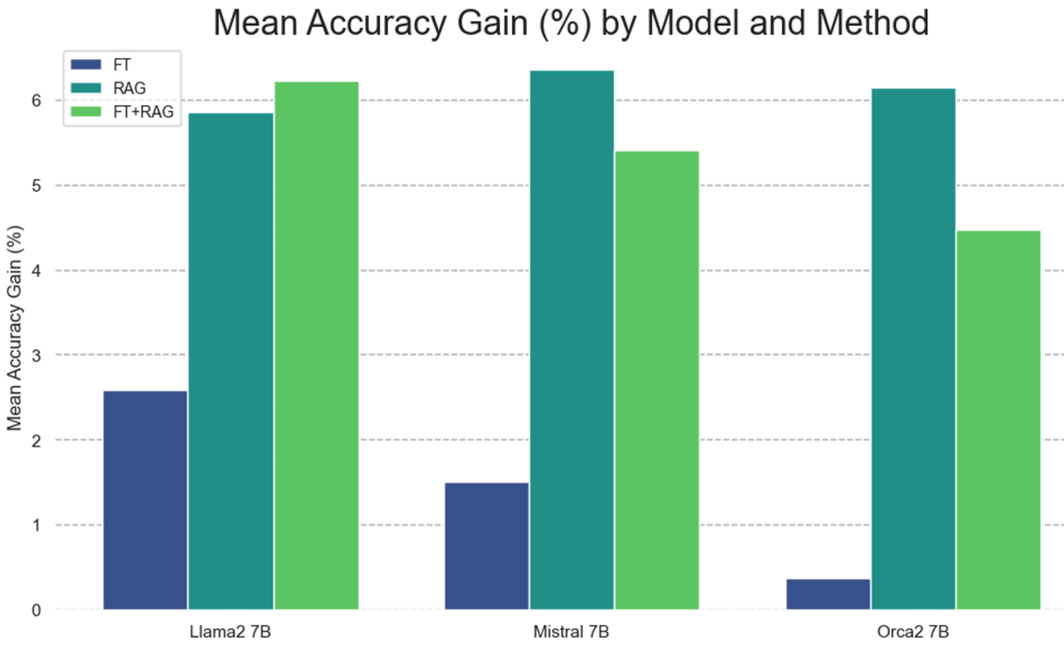


Why RAG?

- **LLMs encapsulate** a vast amount of factual **information** within their **pre-trained weights**
- This **knowledge** is inherently **limited**, relying heavily on the characteristics of the training data
- Solution: use **external datasets** to incorporate **new information** or refine the capabilities of LLMs

Why RAG?

- **LLMs encapsulate** a vast amount of factual **information** within their **pre-trained weights**
- This **knowledge** is inherently **limited**, relying heavily on the characteristics of the training data
- Solution: use **external datasets** to incorporate **new information** or refine the capabilities of LLMs
- Two directions to doing this:
    1. **unsupervised fine-tuning**
    2. retrieval-augmented generation (**RAG**)

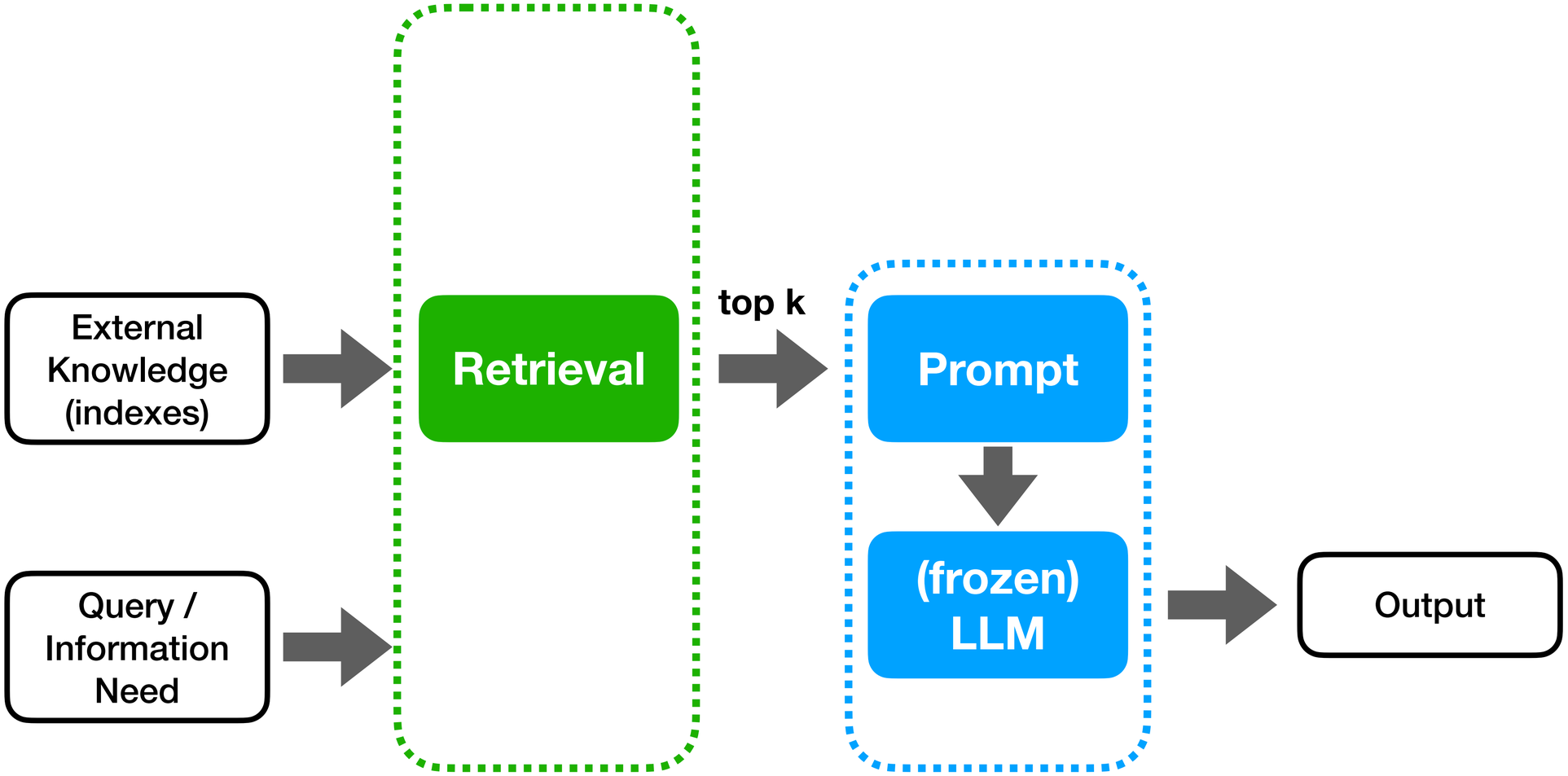


- Ovadia et al.: unsupervised fine-tuning offers some improvement, but **RAG consistently outperforms it**, both for existing knowledge encountered during training and entirely new knowledge
    - LLMs struggle to learn new factual information through unsupervised fine-tuning

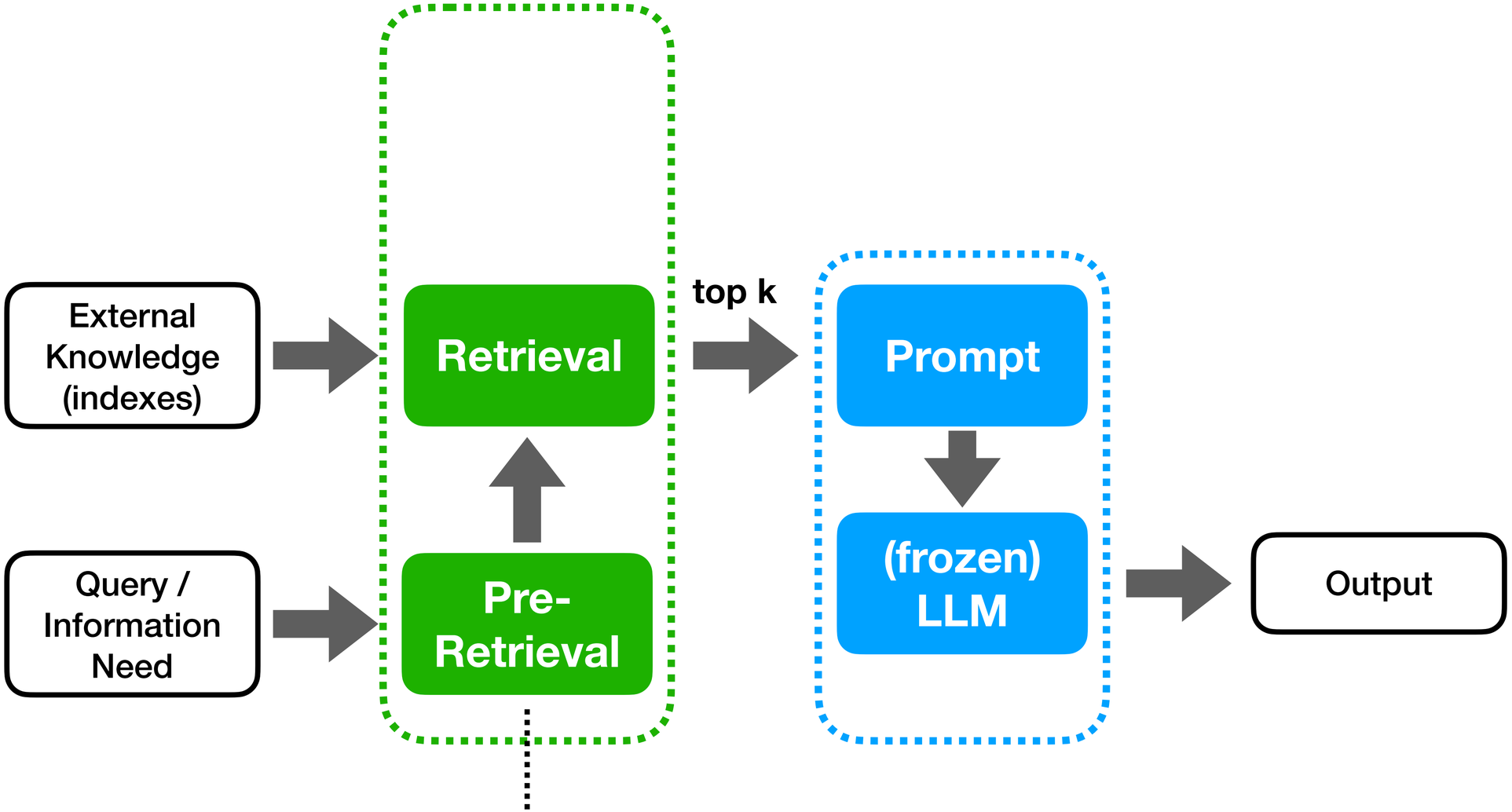
ie lab

Ovadia, O., Brief, M., Mishaeli, M. and Elisha, O., 2023. Fine-tuning or retrieval? comparing knowledge injection in llms. *arXiv preprint arXiv:2312.05934*.

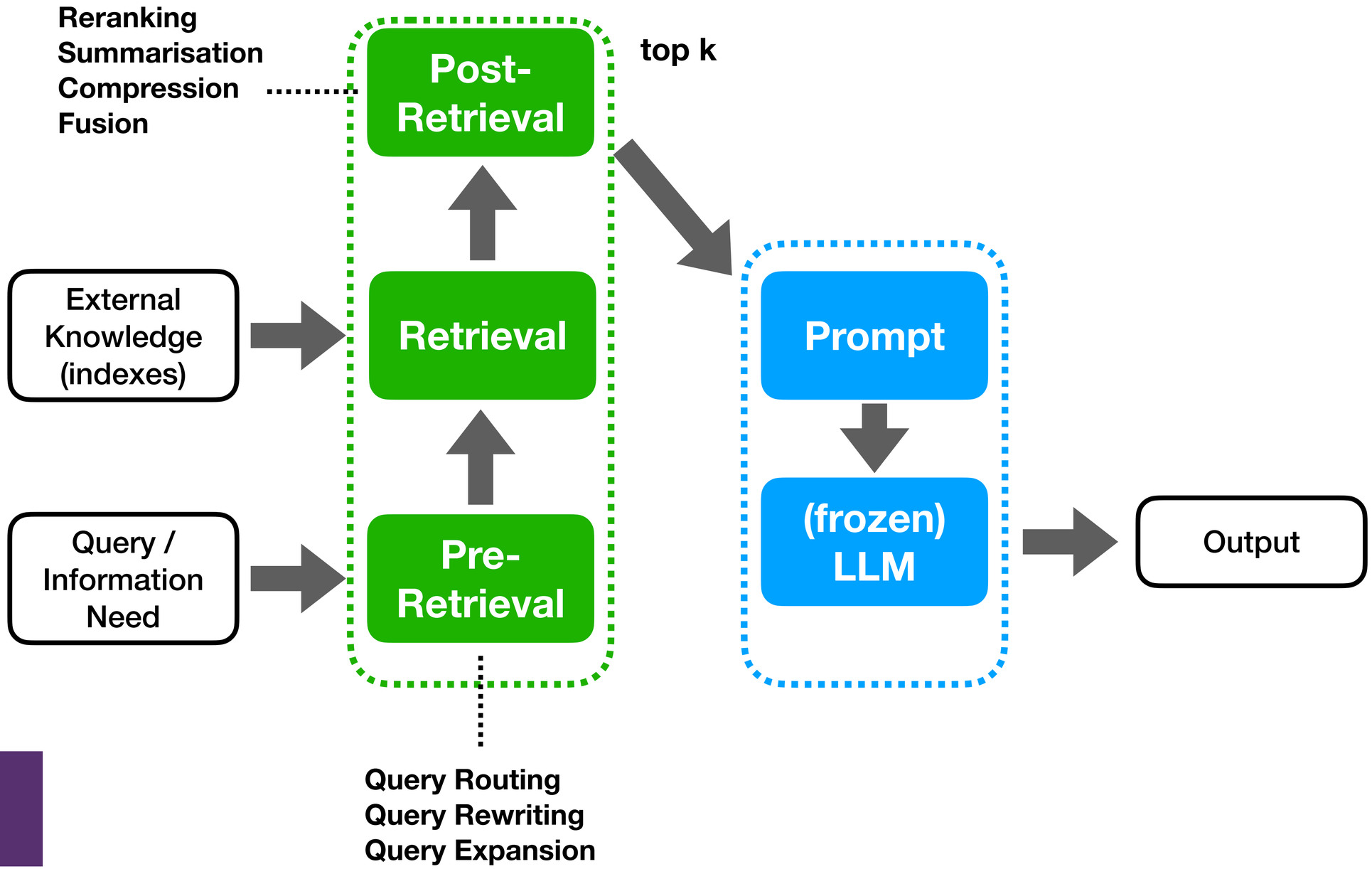
The RAG Pipeline



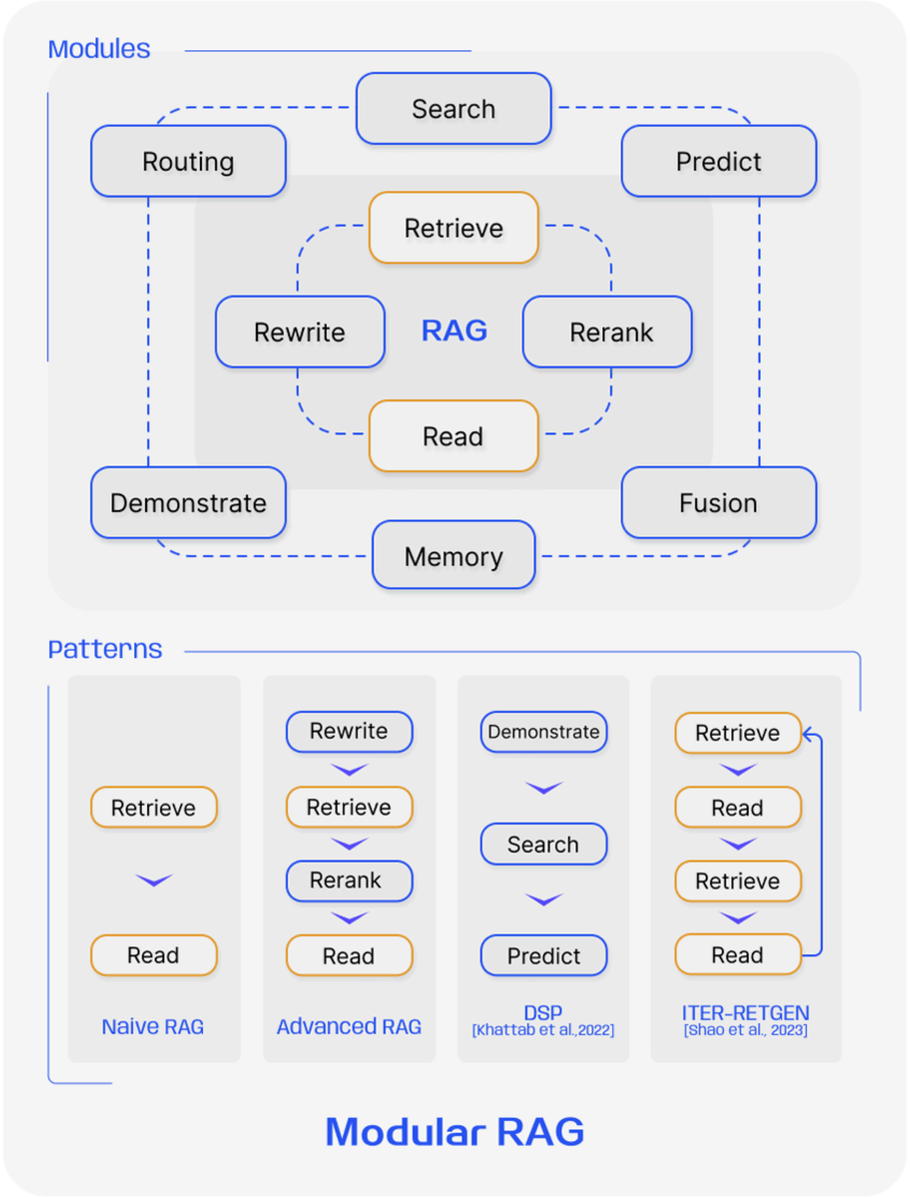
The RAG Pipeline



The RAG Pipeline



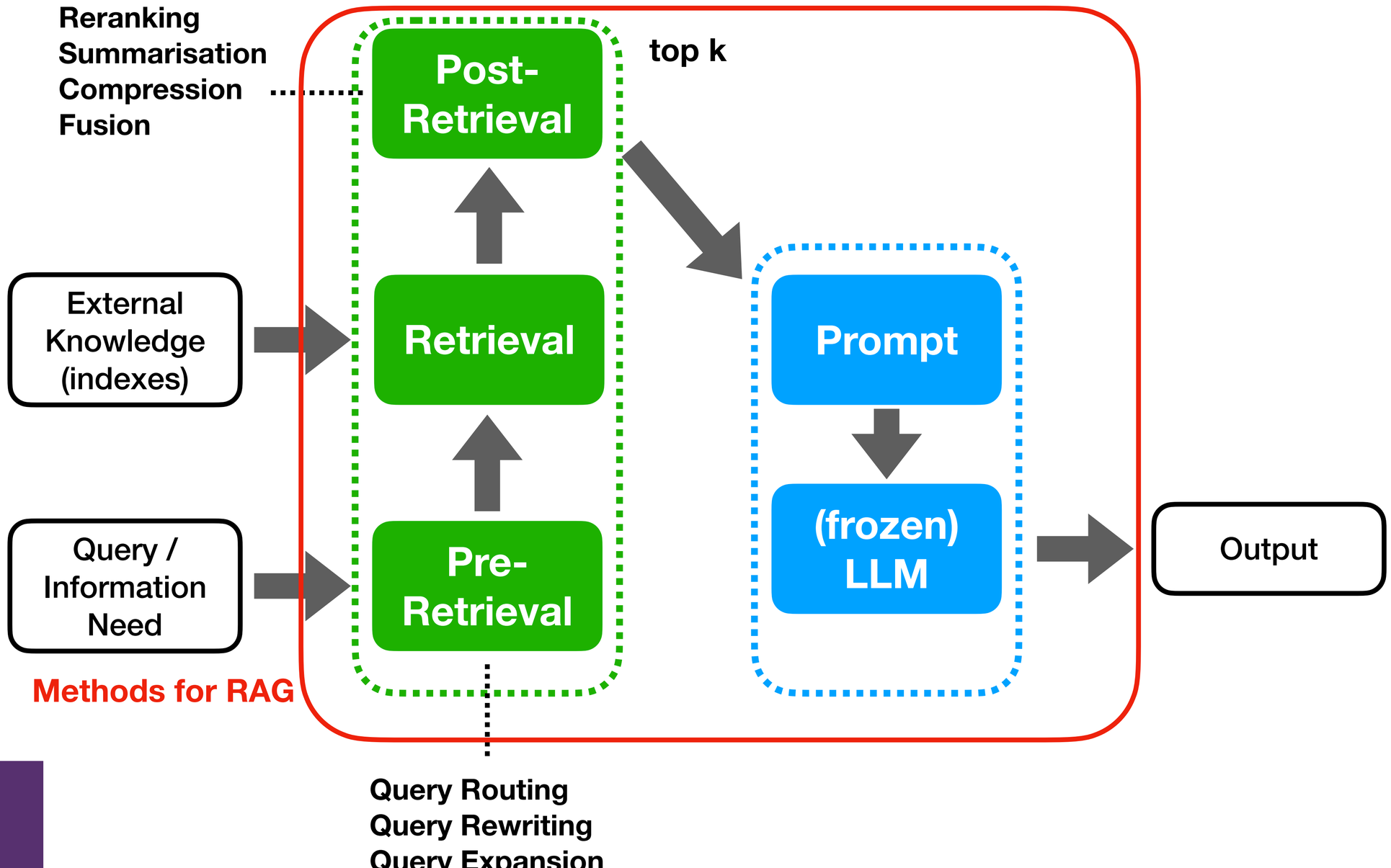
The Modular RAG pipeline



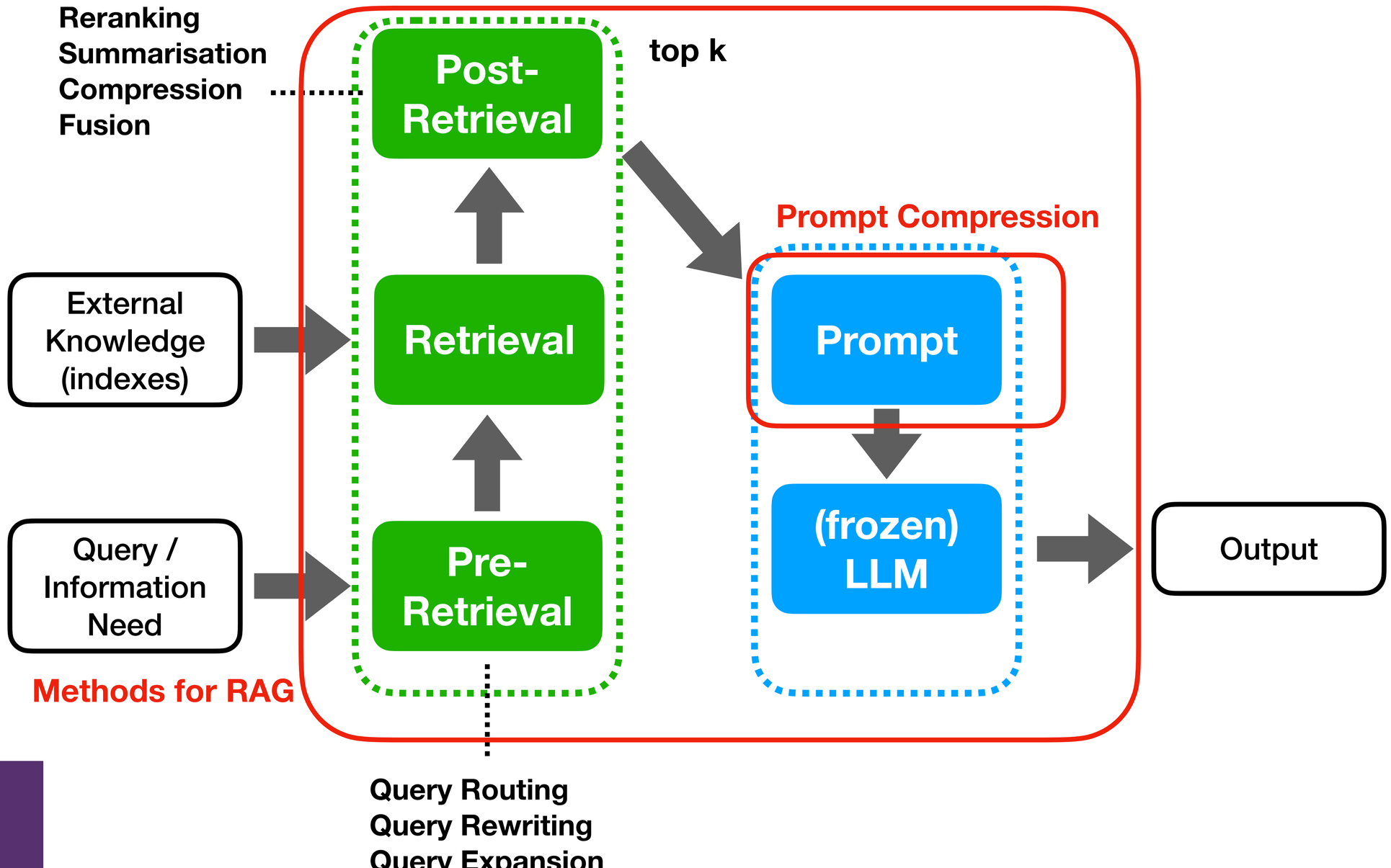
- restructured RAG modules; rearranged RAG pipelines
- Supports **sequential processing** and integrated **end-to-end training** across components
- New/Additional **modules**
- New **Patterns**: e.g. Rewrite-Retrieve-Read, Generate-Read, Recite-Read, Retrieve-Read-Retrieve-Read, Hypothetical Document Embeddings (HyDE), Demonstrate-Search-Predict (DSP)
- **Adaptive** RAG: flexible orchestration of RAG modules and flows, e.g. Self-RAG

Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J. and Wang, H., 2023. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*.

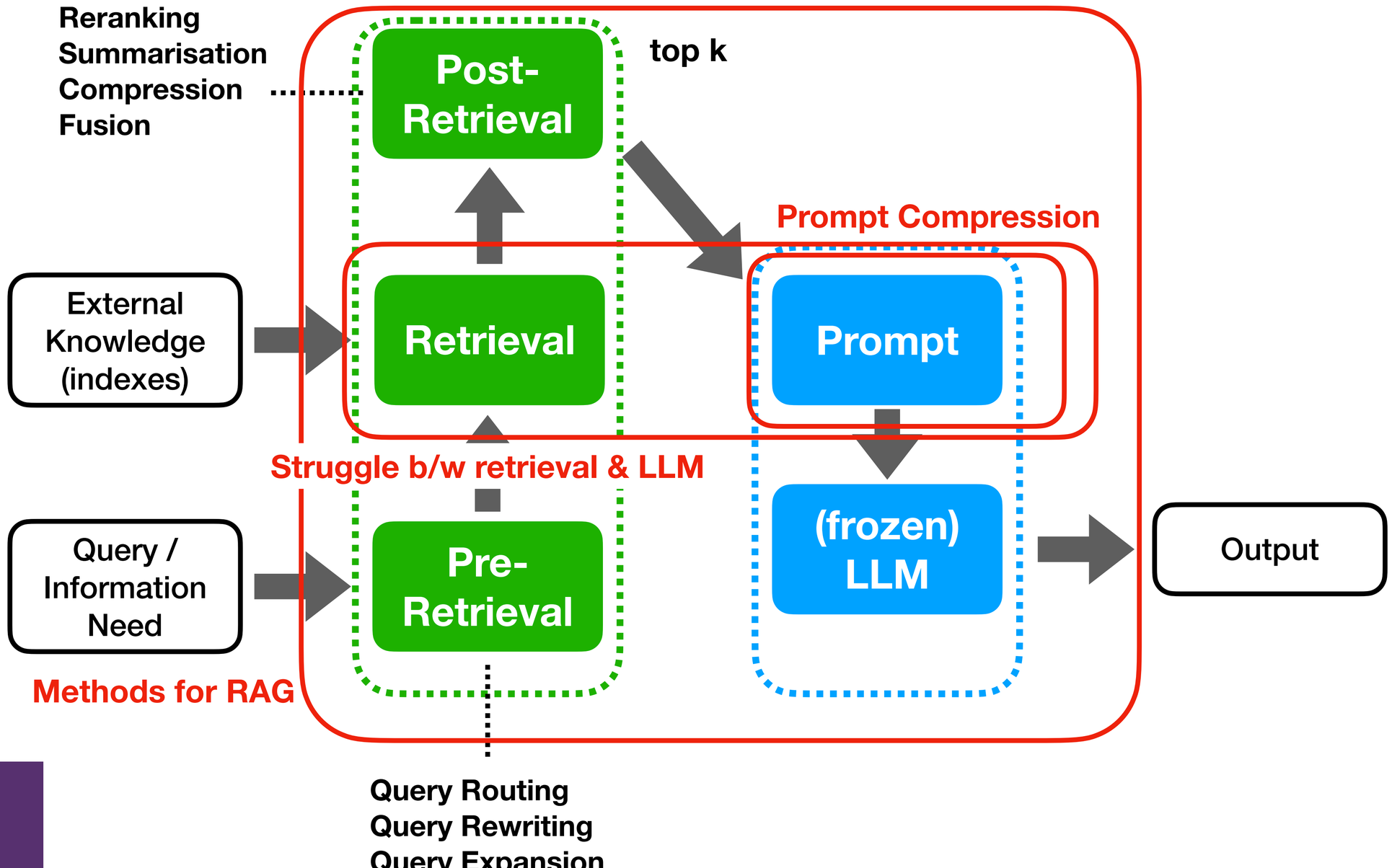
In the remaining of this part…



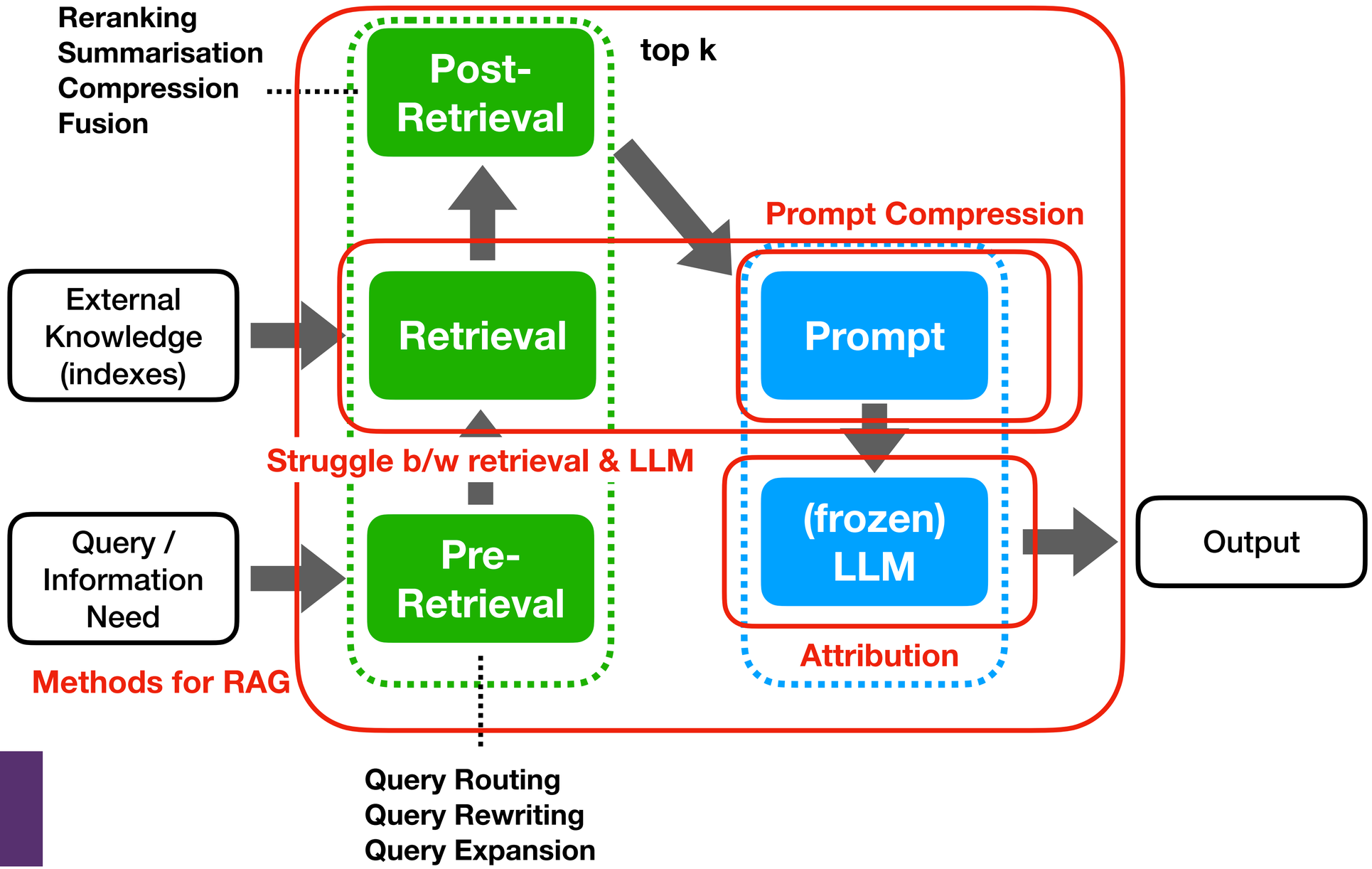
In the remaining of this part…



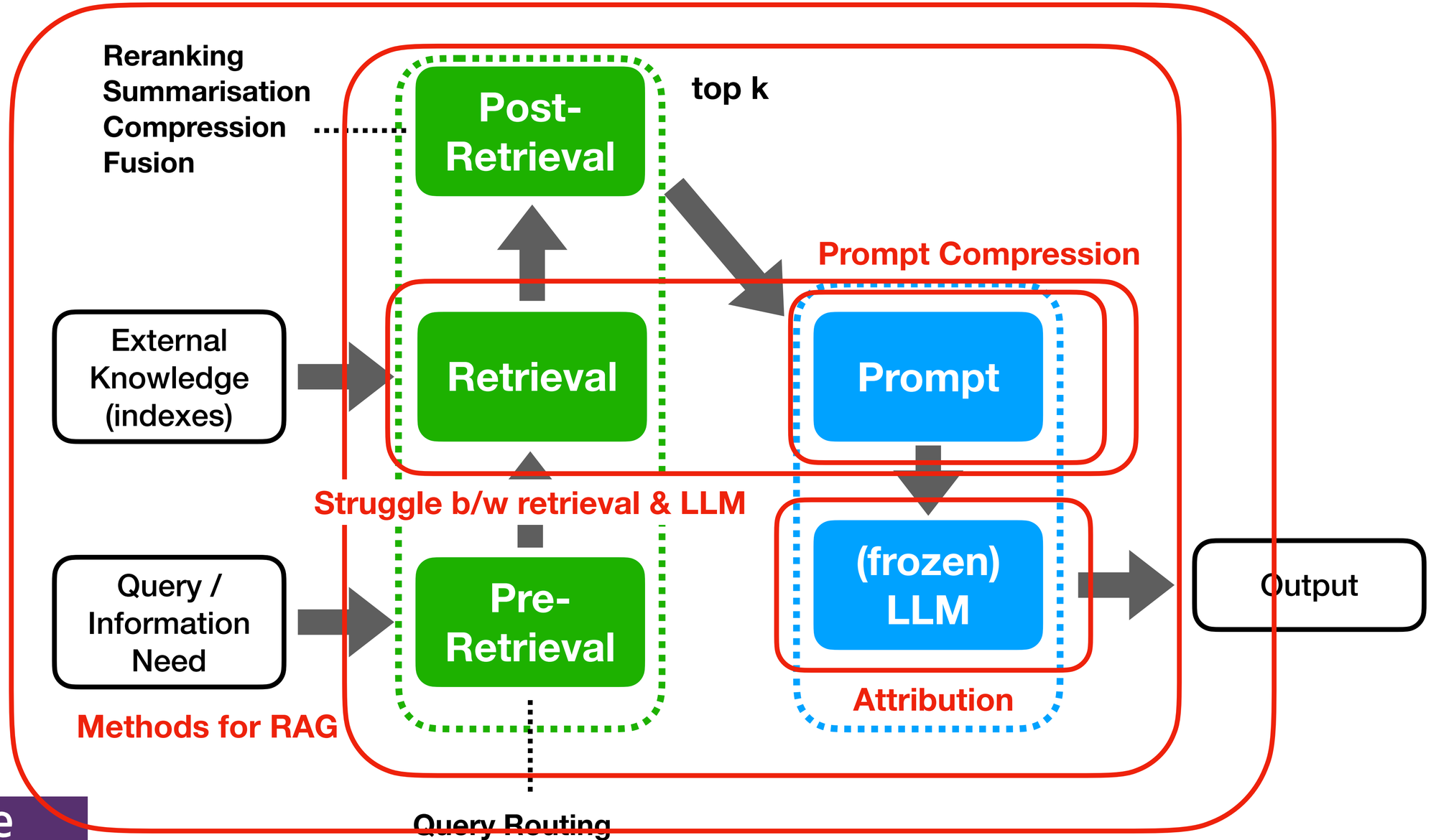
In the remaining of this part…



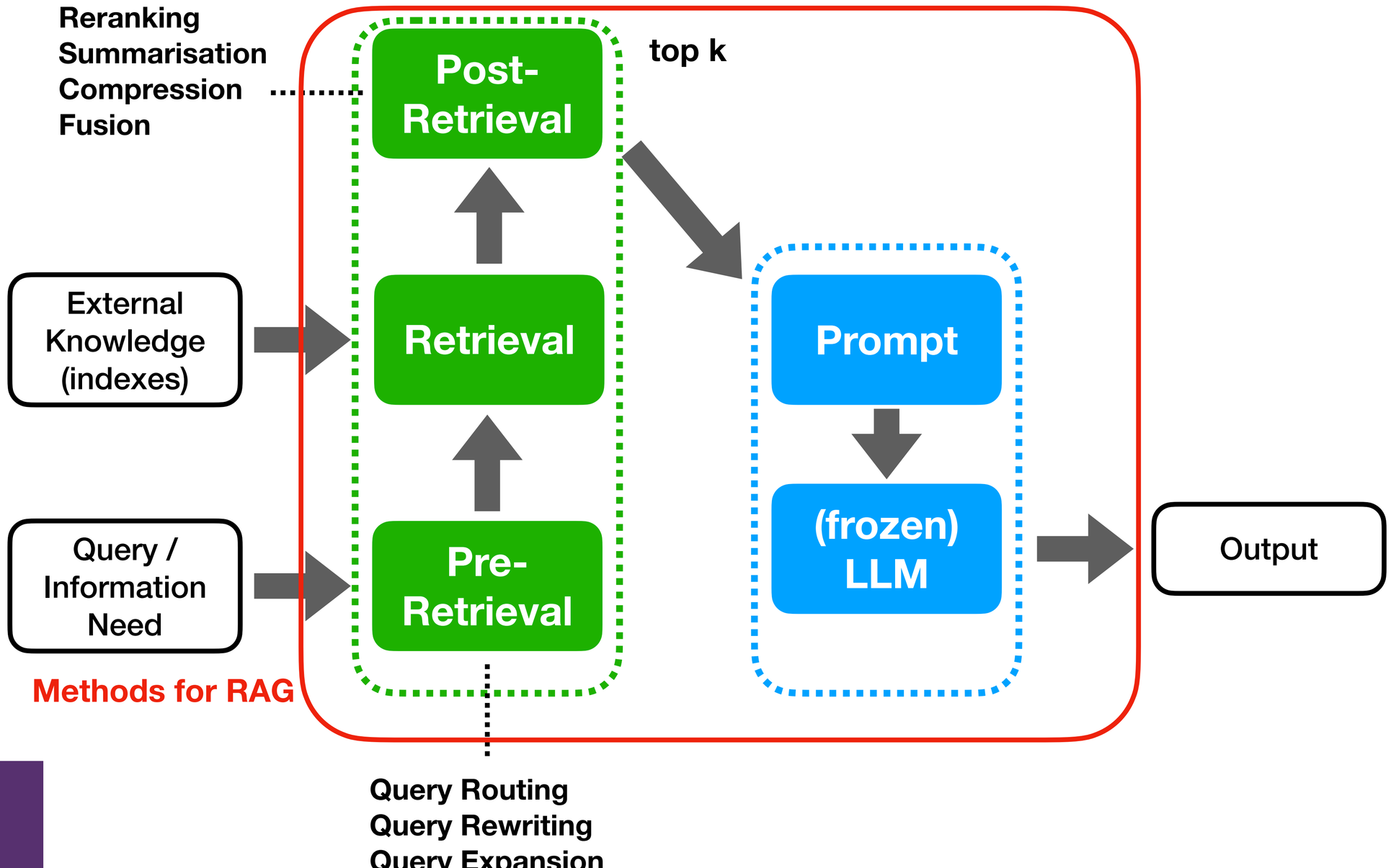
In the remaining of this part…



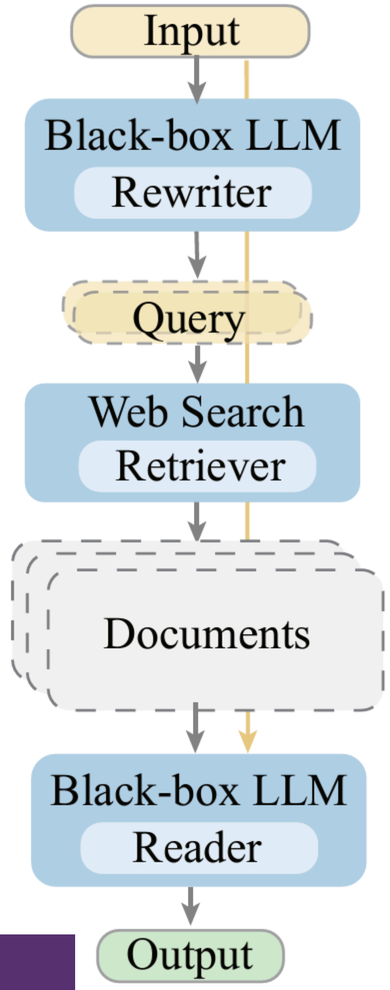
In the remaining of this part…



In the remaining of this part…



The Rewrite-Retrieve-Read Method



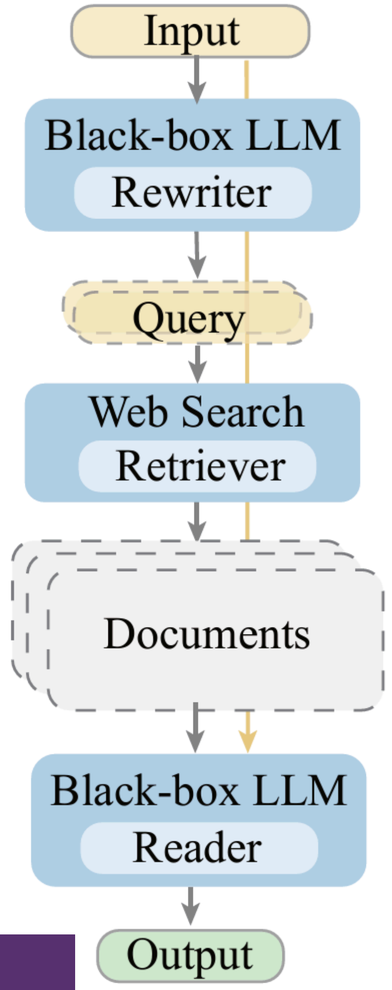
- Queries are often ambiguous and underspecified
- **Rewrite:** black-box LLM is prompted to **re-write query**
- **Retrieve:** search based on the re-written query
- **Read:** provided instruction, query, documents, generate answer

Few-shot (1-3) prompt in format [instruction, demonstrations, input]

Output can be none, one or more queries

Ma, X., Gong, Y., He, P., Zhao, H. and Duan, N., 2023, December. Query Rewriting in Retrieval-Augmented Large Language Models. EMNLP 2023

The Rewrite-Retrieve-Read Method



Open-domain QA

*Think step by step to answer this question, and provide search engine queries for knowledge that you need. Split the queries with ';' and end the queries with '**'. {demonstration} Question: {x} Answer:*

Multiple-choice QA

*Provide a better search query for web search engine to answer the given question, end the queries with '**'. {demonstration} Question: {x} Answer:*

Few-shot (1-3) prompt in format [instruction, demonstrations, input]

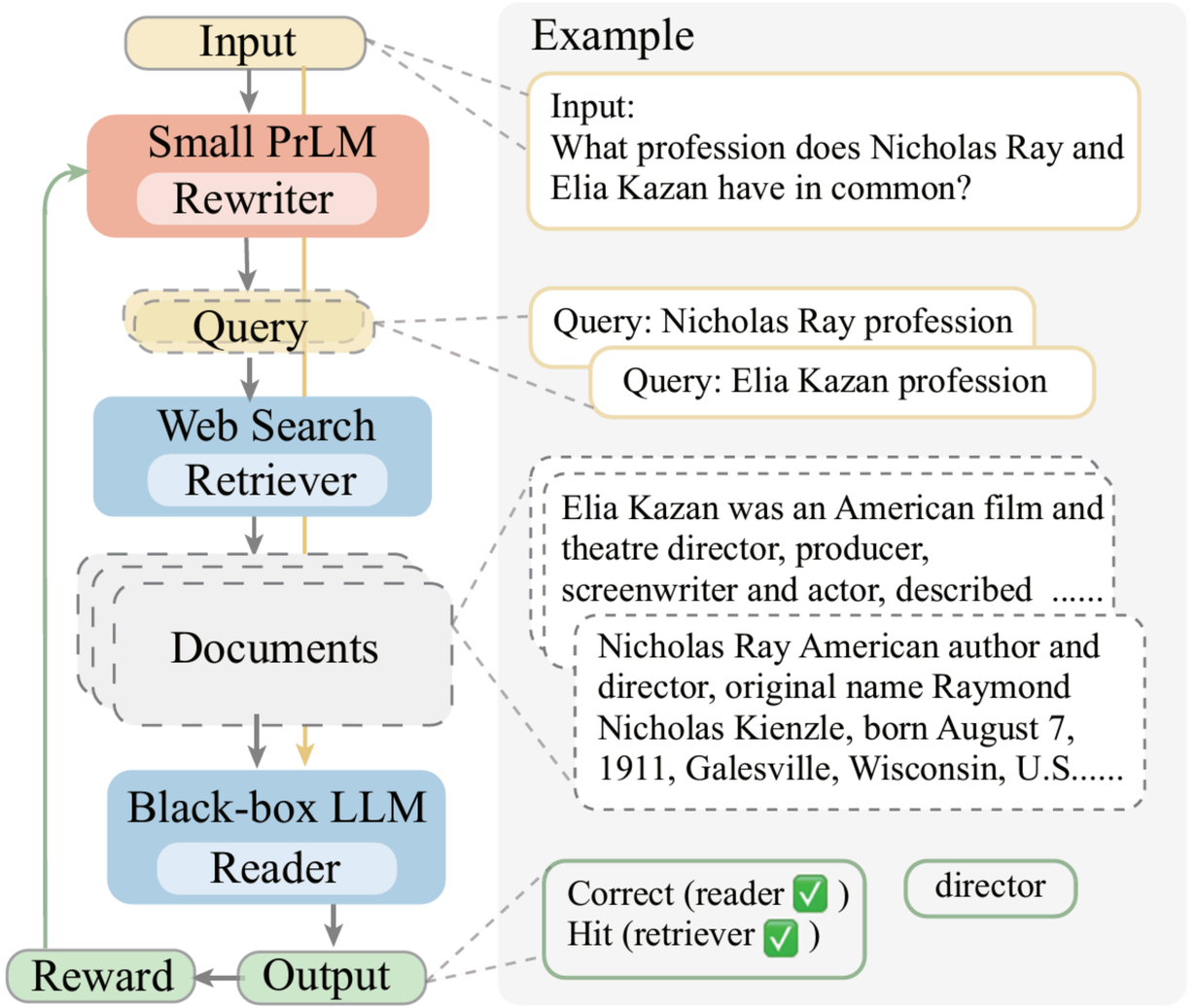
Output can be none, one or more queries

Ma, X., Gong, Y., He, P., Zhao, H. and Duan, N., 2023, December. Query Rewriting in Retrieval-Augmented Large Language Models. EMNLP 2023

The Rewrite-Retrieve-Read Method

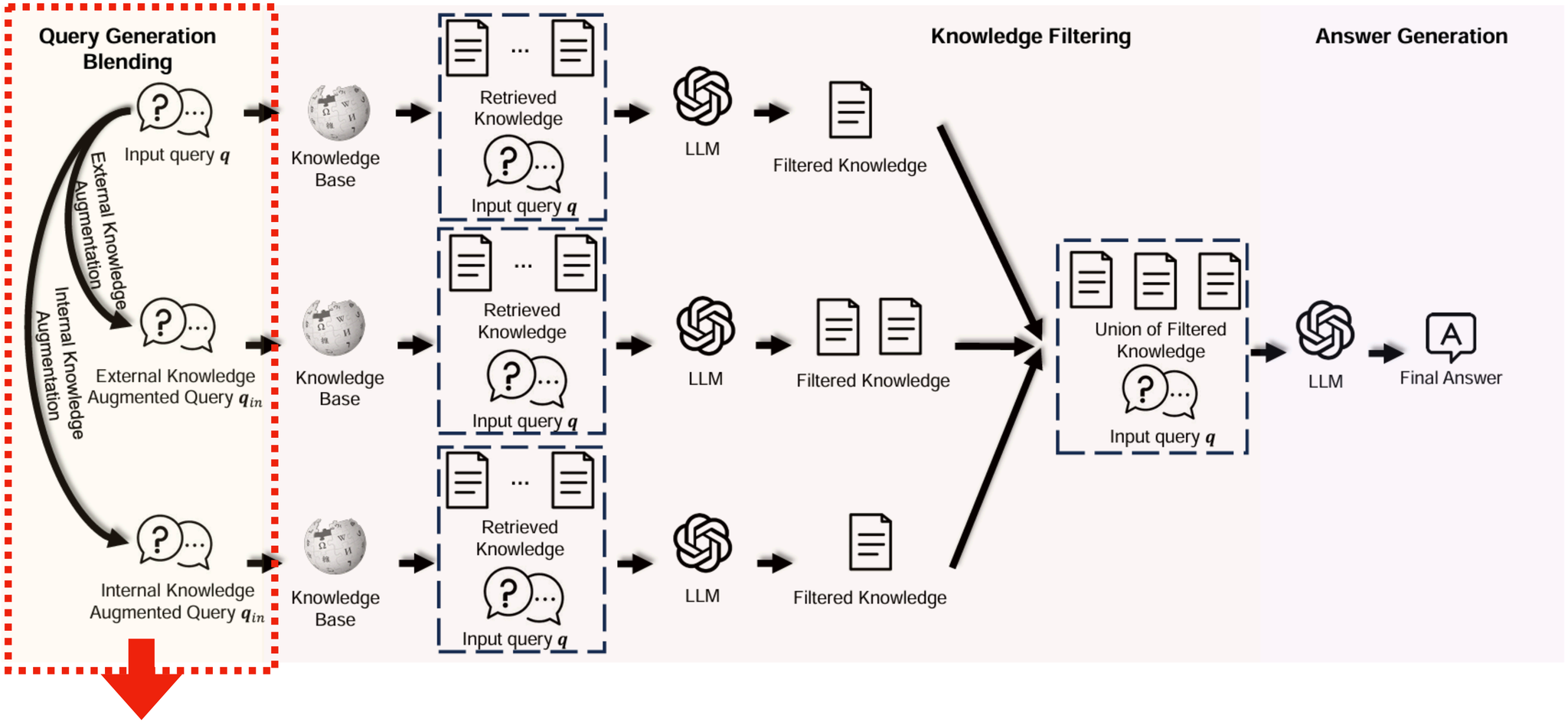
Trainable Rewriter

- Warm-up: training on pseudo-data
  - Collect rewritten queries x, keep only those that generate correct end-to-end answers: use to form warm-up dataset.
  - Fine-tune rewriter on warm-up dataset
- Then continually trained by Reinforcement Learning with PPO
- Reward obtained from end-to-end answers compared to gold + KL regularisation to prevent model doom deviating too far



Ma, X., Gong, Y., He, P., Zhao, H. and Duan, N., 2023, December. Query Rewriting in Retrieval-Augmented Large Language Models. EMNLP 2023

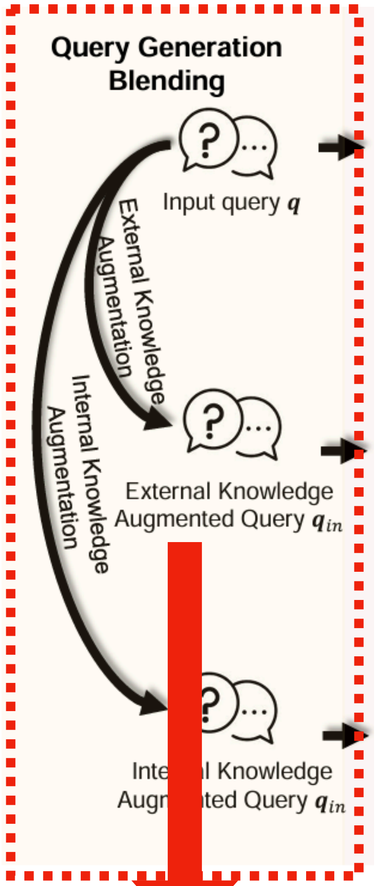
BlendFilter

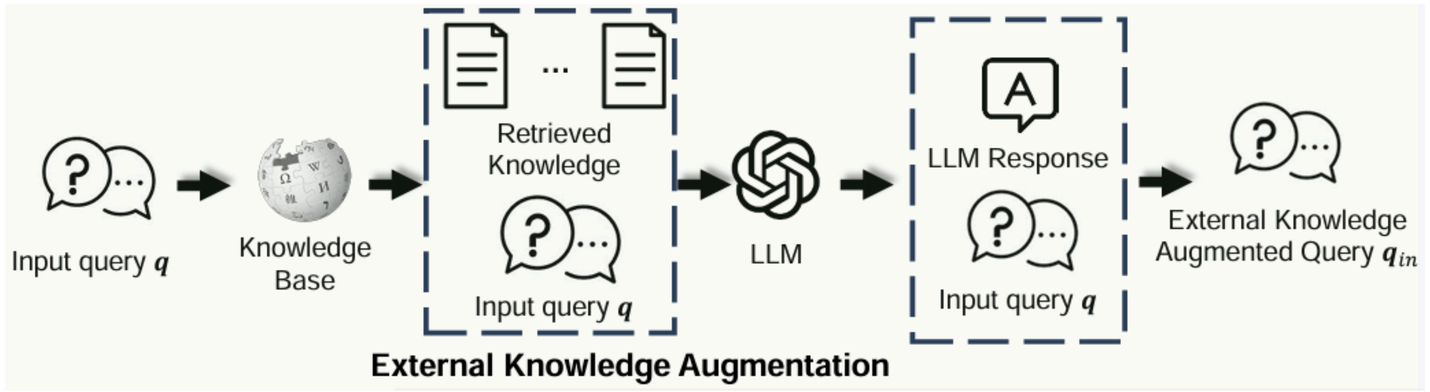


Enhance input queries through different augmentation strateties

Wang, H., Zhao, T. and Gao, J., 2024. BlendFilter: Advancing Retrieval-Augmented Large Language Models via Query Generation Blending and Knowledge Filtering. *arXiv preprint arXiv:2402.11129*.

BlendFilter





1

1. Use initial query to retrieve external knowledge
2. Pass this to LLM prompt with few-shot examples and Chain of Thought
3. Generate answer
4. Concatenate answer & original query

Enhance input queries through different augmentation strateties

Wang, H., Zhao, T. and Gao, J., 2024. BlendFilter: Advancing Retrieval-Augmented Large Language Models via Query Generation Blending and Knowledge Filtering. *arXiv preprint arXiv:2402.11129*.

Prompt for External Knowledge Augmentation on HotPotQA

Answer questions following the given format.

Knowledge:{`Example_Knowledge`}
Question:Are It Might Get Loud and Mr. Big both Canadian documentaries?
Let's think step by step.
Mr. Big is a 2007 documentary which examines the "Mr. Big" undercover methods used by the Royal Canadian Mounted Police. However, It Might Get Loud is a 2008 American documentary film.
So the answer is no.

Knowledge:{`Example_Knowledge`}
Question:Were László Benedek and Leslie H. Martinson both film directors?
Let's think step by step.
László Benedek was a Hungarian-born film director and Leslie H. Martinson was an American film director.
So the answer is yes.

Knowledge:{`Example_Knowledge`}
Question:Lucium was confimed to be an impure sample of yttrium by an English chemist who became the president of what?
Let's think step by step.
Lucium was confimed to be an impure sample of yttrium by William Crookes. William Crookes is Sir William Crookes. Sir William Crookes became the president of the Society for Psychical Research.
So the answer is Society for Psychical Research.

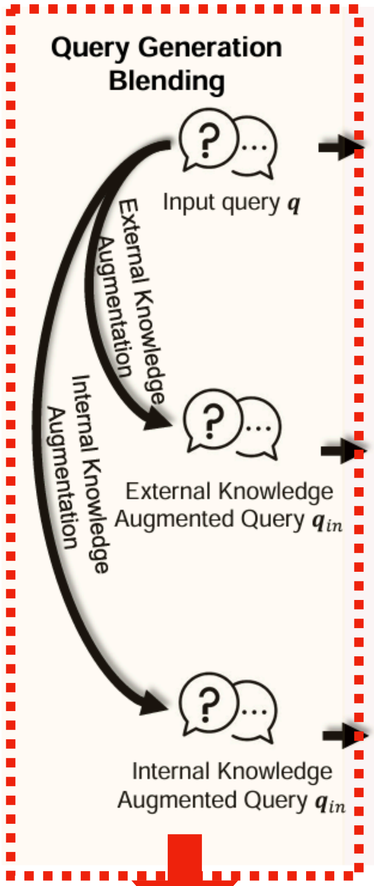
Knowledge:{`Knowledge`}
Question:{`question`}
Let's think step by step.

Prompt for External Knowledge Augmentation

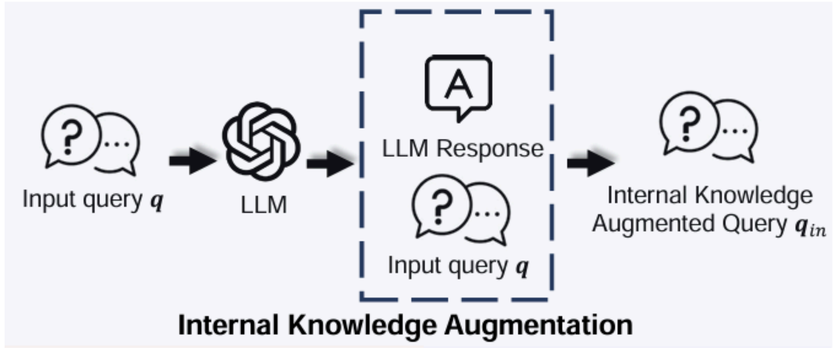
Few Shot Examples

Chain-of-Thought

BlendFilter



1. Prompt LLM to respond to query based on its own knowledge only
2. Concatenate answer & query



Prompt for Internal Knowledge Augmentation

Please write a passage to answer the question.

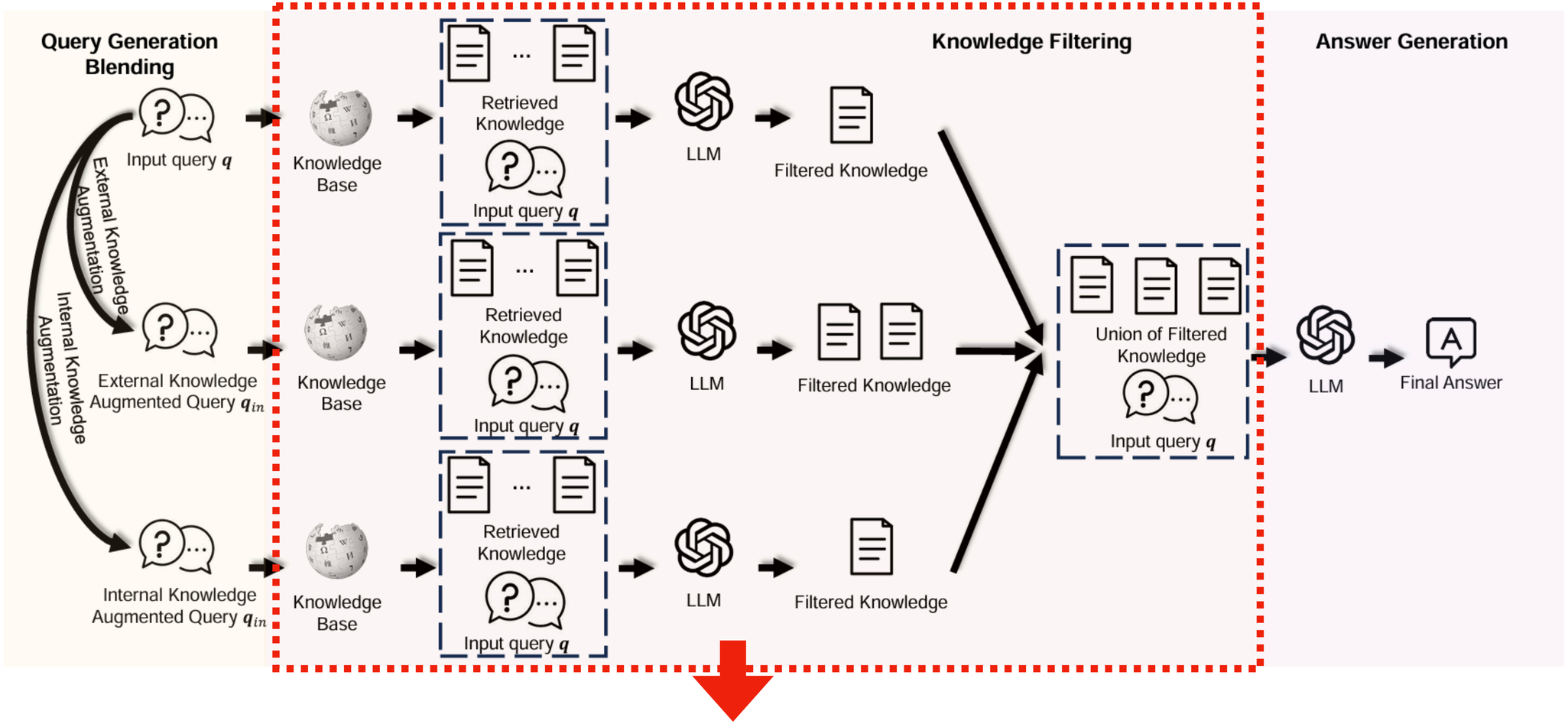
Question:{question}
Passage:

2

Enhance input queries through different augmentation strateties

Wang, H., Zhao, T. and Gao, J., 2024. BlendFilter: Advancing Retrieval-Augmented Large Language Models via Query Generation Blending and Knowledge Filtering. *arXiv preprint arXiv:2402.11129*.

BlendFilter



Eliminate irrelevant retrieved knowledge using prompted LLM

Knowledge Filtering Prompt

Prompt for Knowledge Filtering on HotPotQA and 2WikiMultihopQA

What general topic is Question {question} related to?
Answer:The topic is related to

— forget your knowledge about {topic}. Please only consider the knowledge below.
knowledge 0 : {Retrieved_knowledge0}
knowledge 1 : {Retrieved_knowledge1}
knowledge 2 : {Retrieved_knowledge2}
knowledge 3 : {Retrieved_knowledge3}
knowledge 4 : {Retrieved_knowledge4}
Please check the relevance between {question} and knowledges 0-4 one by one, remove the irrelevant ones and show me the relevant ones. There may be multiple relevent ones. Please take a deep breath and do it step by step.

—— Please check the relevance between the given question and knowledges 0-4 one by one based on the given context. ONLY output the relevant knowledge ids (0-4). There may be multiple relevent ones.

Context:{LLM_Last_Generated_Context}

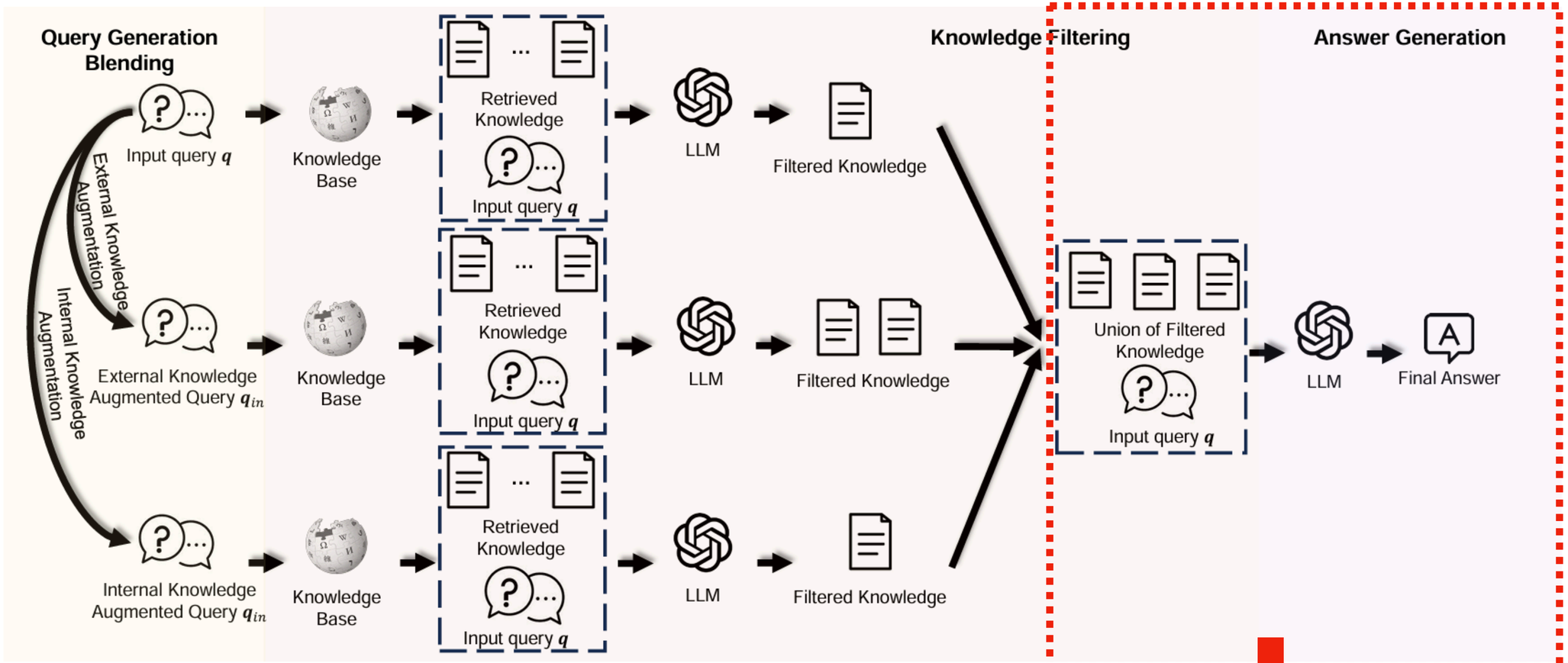
Question:{question}

knowledge 0 : {Retrieved_knowledge0}
knowledge 1 : {Retrieved_knowledge1}
knowledge 2 : {Retrieved_knowledge2}
knowledge 3 : {Retrieved_knowledge3}
knowledge 4 : {Retrieved_knowledge4}

Answer:

Wang, H., Zhao, T. and Gao, J., 2024. BlendFilter: Advancing Retrieval-Augmented Large Language Models via Query Generation Blending and Knowledge Filtering. *arXiv preprint arXiv:2402.11129*.

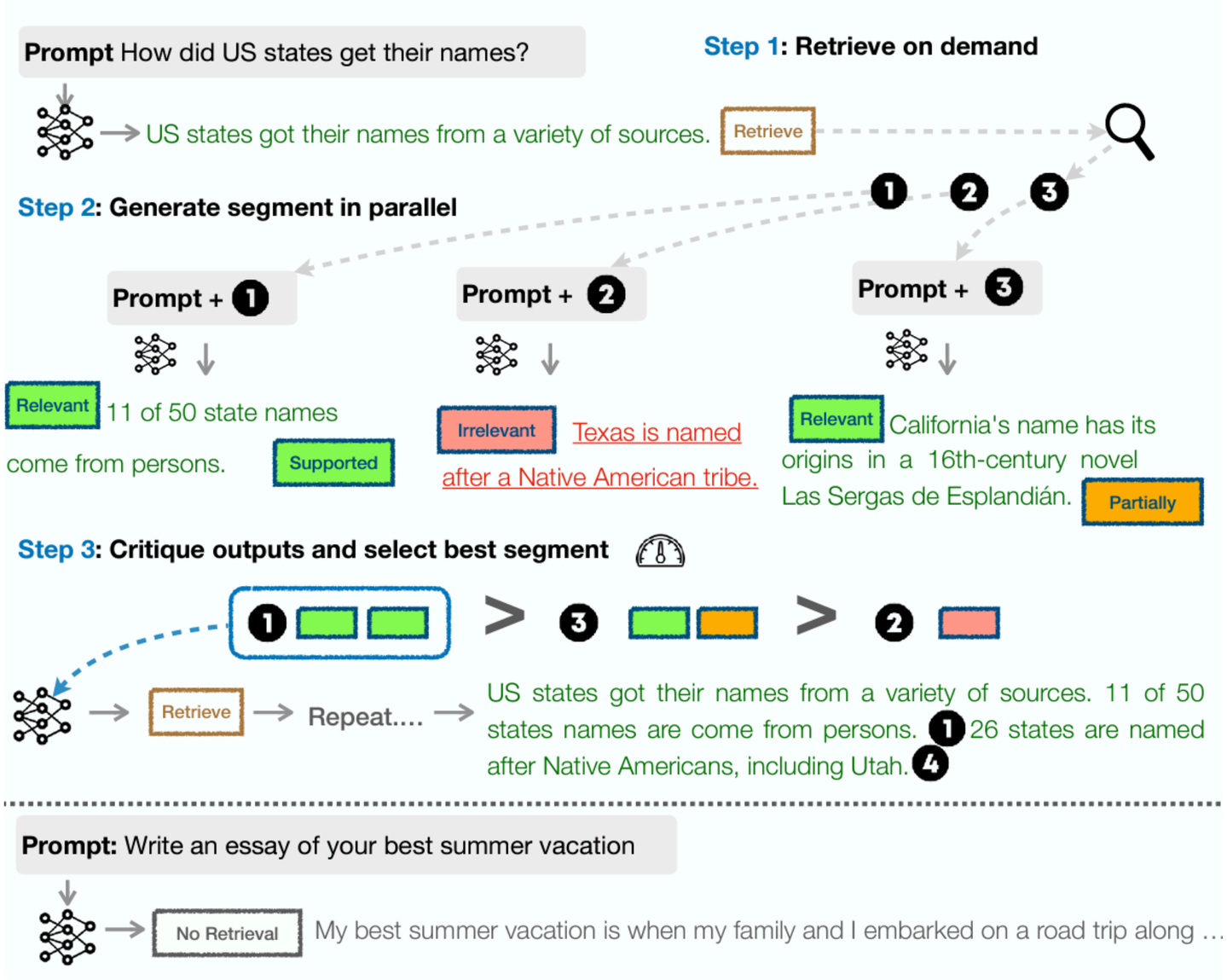
BlendFilter



LLM generates answer based on filtered knowledge and original query

Wang, H., Zhao, T. and Gao, J., 2024. BlendFilter: Advancing Retrieval-Augmented Large Language Models via Query Generation Blending and Knowledge Filtering. *arXiv preprint arXiv:2402.11129*.

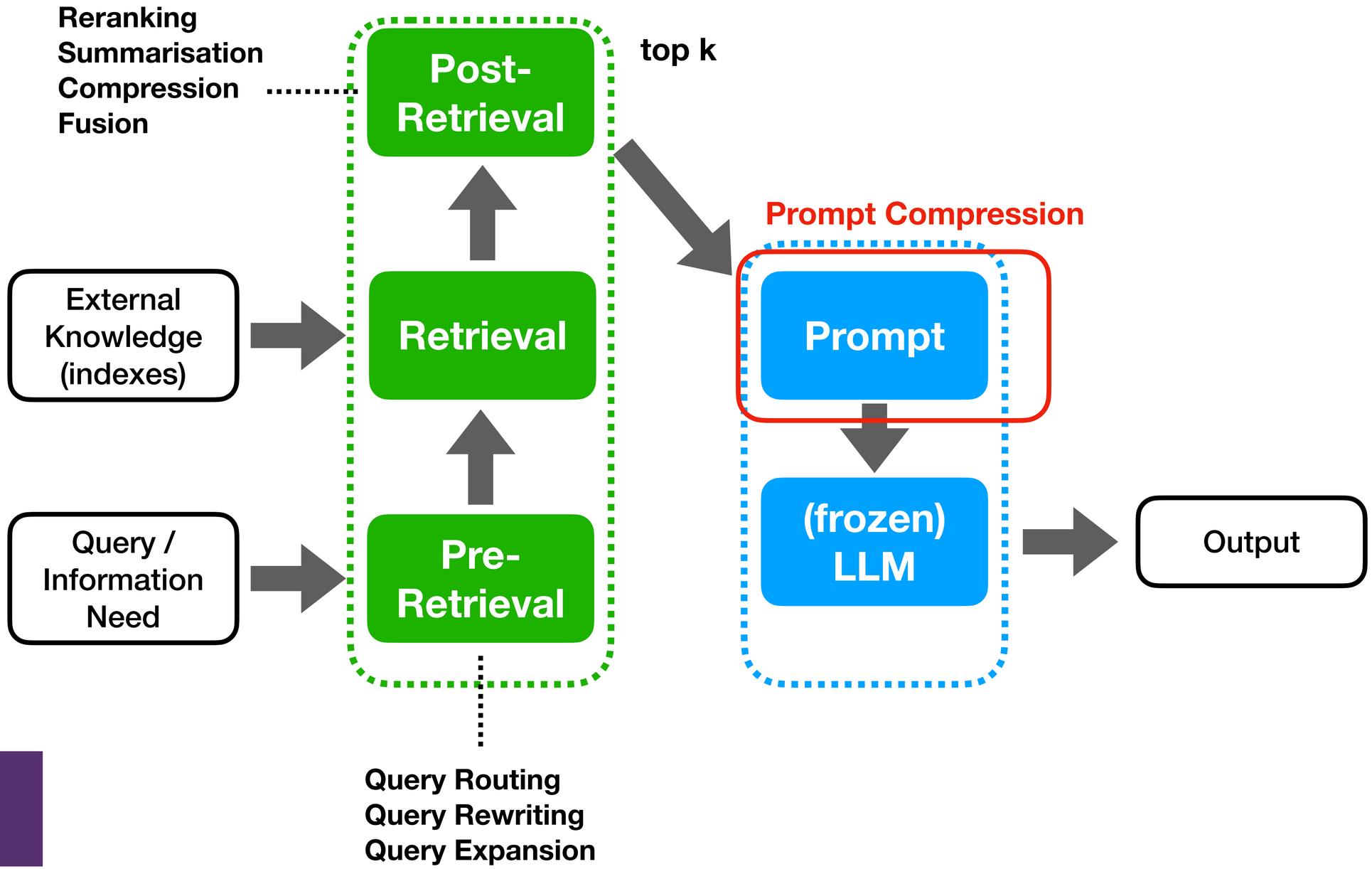
Retrieve-Generate-Critique: Self-RAG



- **Adaptively retrieve** passages on-demand
  - Can decide to (1) **retrieve**, or (2) **not retrieve**
- LLM **reflects on retrieved passages** and its generation using special tokens (reflection tokens)
  - **Retrieval reflection:** do I need to retrieve?
    YES->output retrieval token to call retriever on demand
  - **Critique reflection:** is the generated answer good?
    SELECT best based on factuality and overall quality

Asai, A., Wu, Z., Wang, Y., Sil, A. and Hajishirzi, H., Self-RAG: Learning to Retrieve, Generate, and Critique through Self-Reflection. ICLR 2024

In the remaining of this part…



Prompt Compression

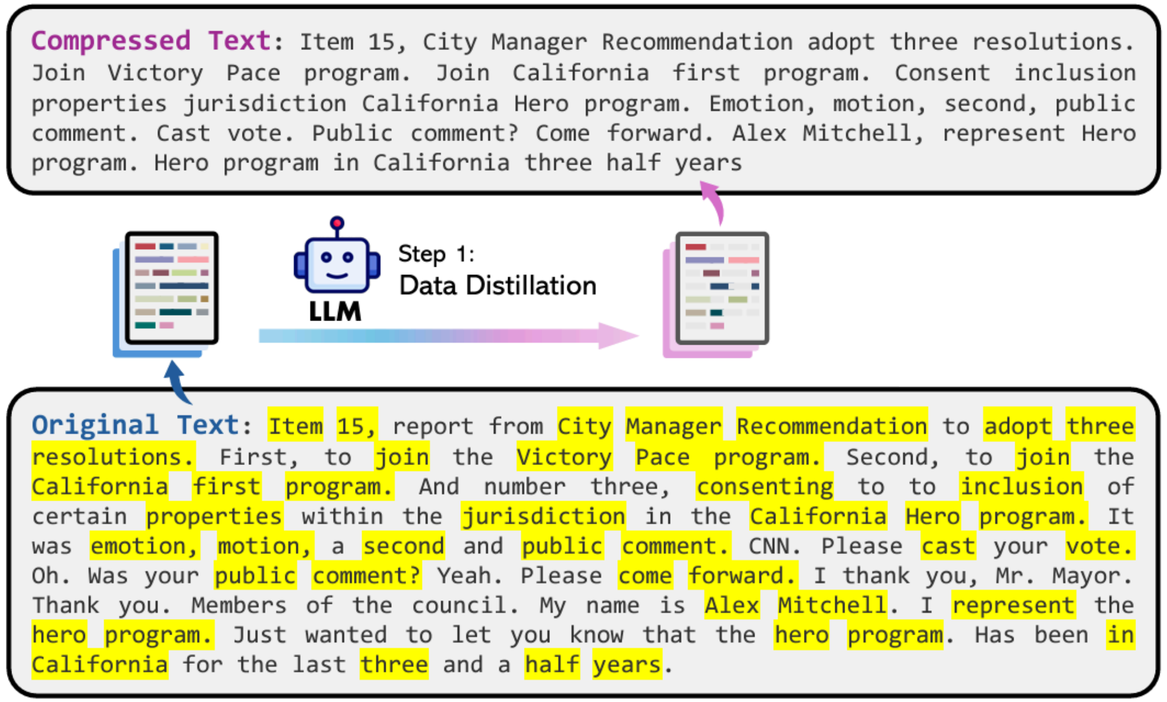
- Reduce the prompt (aka context) to consume less tokens than in its original form.
  Why prompt compression?
  - Long prompts often **confuse LLMs** and yield **lower effectiveness**
  - **Latency** of decoder-based LLMs is **quadratic** with respect to **prompt length**

- Two main **categories** of approaches to prompt compression:
  1. **Lexical based:** Compress prompts by **removing tokens** according to their information entropy obtained from LLM
  2. **Embedding based:** Compress prompt into **special tokens** (short compact *memory slots*) that can be directly conditioned on by the LLM

ie lab

The University Of Queensland AUSTRALIA
CREATE CHANGE

Lexical-based Compression via Distillation



Step 1: Data Distillation

Prompt GPT4 to generate compressed text from original text that meets criteria:

1. Token reduction
2. Informativeness
3. Faithfulness

Pan, Z., Wu, Q., Jiang, H., Xia, M., Luo, X., Zhang, J., Lin, Q., Rühle, V., Yang, Y., Lin, C.Y. and Zhao, H.V., 2024. Llmlingua-2: Data distillation for efficient and faithful task-agnostic prompt compression. *arXiv preprint arXiv:2403.12968*.

Lexical-based Compression via Distillation

Compressed Text: Item 15, City Manager Recommendation adopt three resolutions. Join Victory Pace program. Join California first program. Consent inclusion properties jurisdiction California Hero program. Emotion, motion, second, public comment. Cast vote. Public comment? Come forward. Alex Mitchell, represent Hero program. Hero program in California three half years

Original Text: Item 15, report from City Manager Recommendation to adopt three resolutions. First, to join the Victory Pace program. Second, to join the California first program. And number three, consenting to to inclusion of certain properties within the jurisdiction in the California Hero program. It was emotion, motion, a second and public comment. CNN. Please cast your vote. Oh. Was your public comment? Yeah. Please come forward. I thank you, Mr. Mayor. Thank you. Members of the council. My name is Alex Mitchell. I represent the hero program. Just wanted to let you know that the hero program. Has been in California for the last three and a half years.

Our Instruction for Compression:
Compress the given text to short expressions, and such that you (GPT-4) can reconstruct it as close as possible to the original. Unlike the usual text compression, I need you to comply with the 5 conditions below:
1. You can ONLY remove unimportant words.
2. Do not reorder the original words.
3. Do not change the original words.
4. Do not use abbreviations or emojis.
5. Do not add new words or symbols.
Compress the origin aggressively by removing words only. Compress the origin as short as you can, while retaining as much information as possible. If you understand, please compress the following text: {*text to compress*}
The compressed text is:

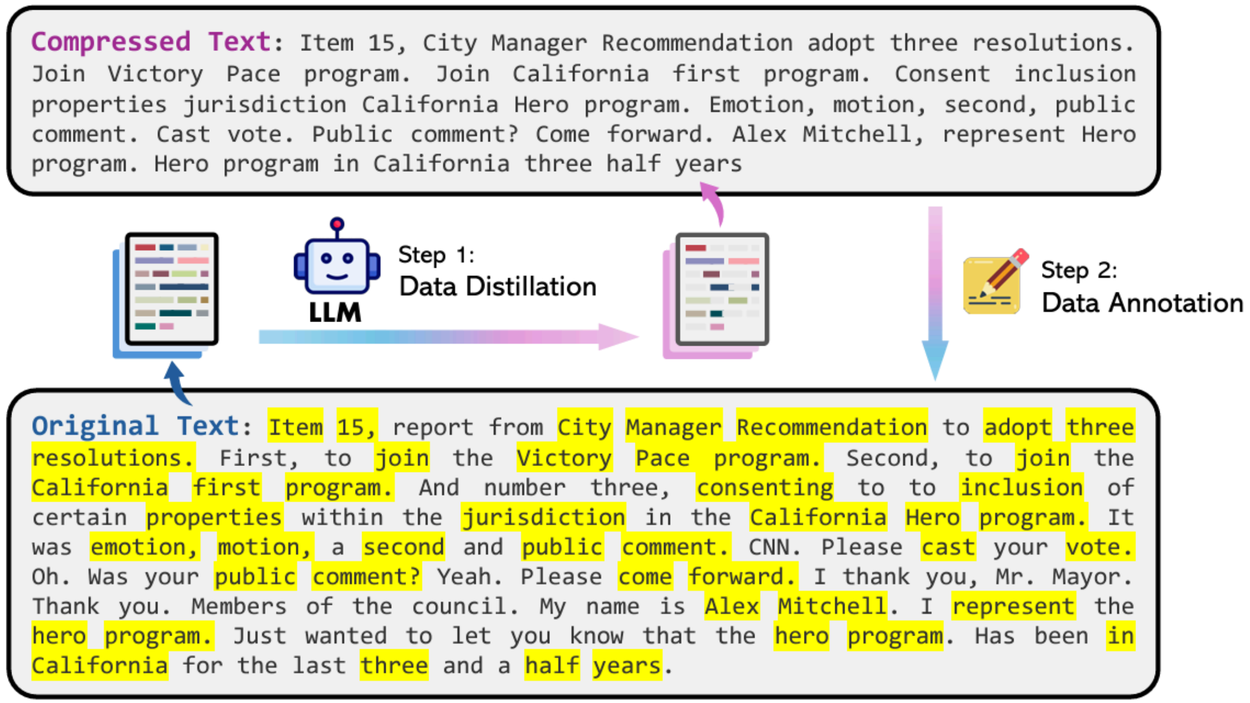
Step 1: Data Distillation

Prompt GPT4 to generate compressed text from original text that meets criteria:
1. Token reduction
2. Informativeness
3. Faithfulness

Challenge: GPT-4 does not consistently follow instructions. Remedies:
- Instruction design (i.e. tweaked the prompt)
- Chunk-wise compression (i.e. segment long context into multiple 512 tokens chunks)

Pan, Z., Wu, Q., Jiang, H., Xia, M., Luo, X., Zhang, J., Lin, Q., Rühle, V., Yang, Y., Lin, C.Y. and Zhao, H.V., 2024. LLmlingua-2: Data distillation for efficient and faithful task-agnostic prompt compression. *arXiv preprint arXiv:2403.12968*.

Lexical-based Compression via Distillation



Step 2: Data Annotation

Assign binary label to each token in original text to determine if it should be preserved or discarded after compression.

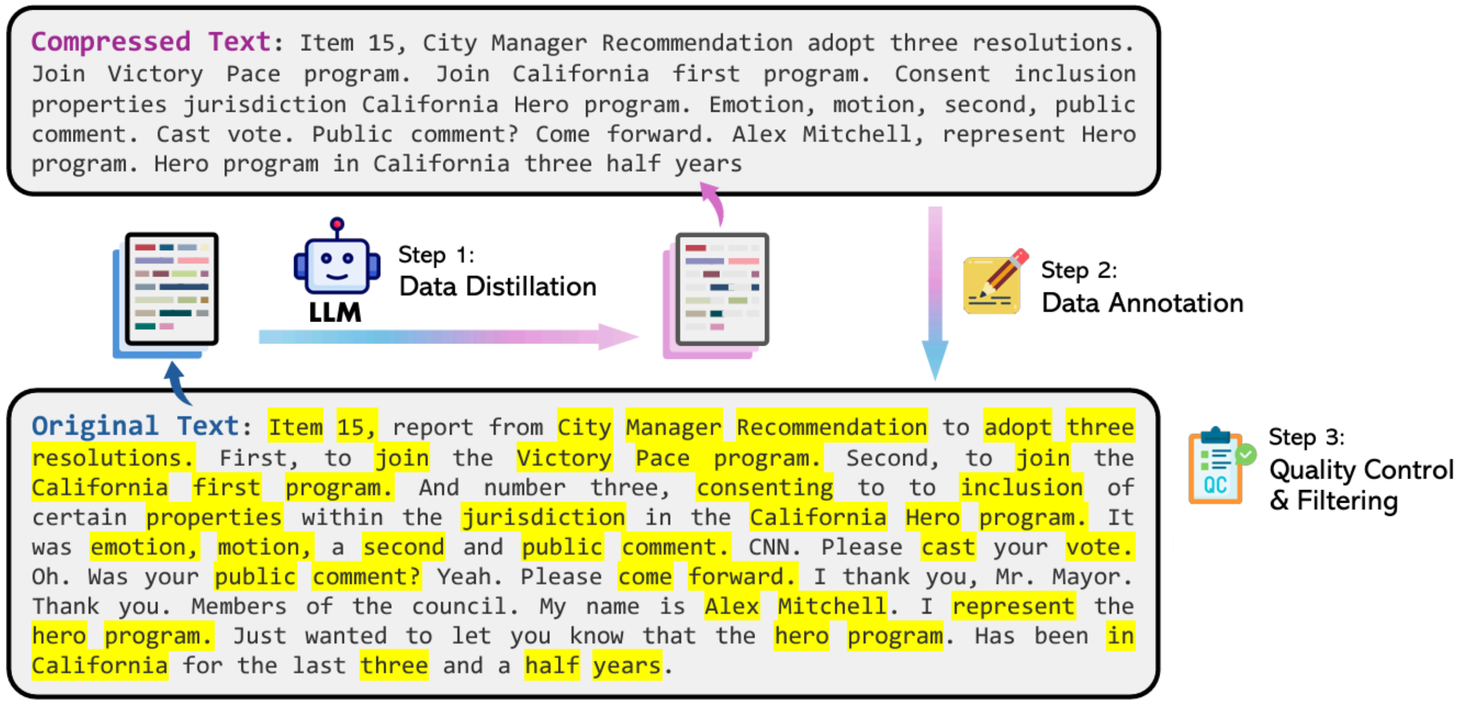
Challenges:

1. Ambiguity: a word in compressed text may appear multiple times in original text
2. Variation: GPT-4 may modify the original words in tense, plural form, etc. during compression
3. Reordering: order of words may be changed

ie lab

Pan, Z., Wu, Q., Jiang, H., Xia, M., Luo, X., Zhang, J., Lin, Q., Rühle, V., Yang, Y., Lin, C.Y. and Zhao, H.V., 2024. LImlingua-2: Data distillation for efficient and faithful task-agnostic prompt compression. *arXiv preprint arXiv:2403.12968*.

Lexical-based Compression via Distillation



Step 3: Quality Control & Filtering

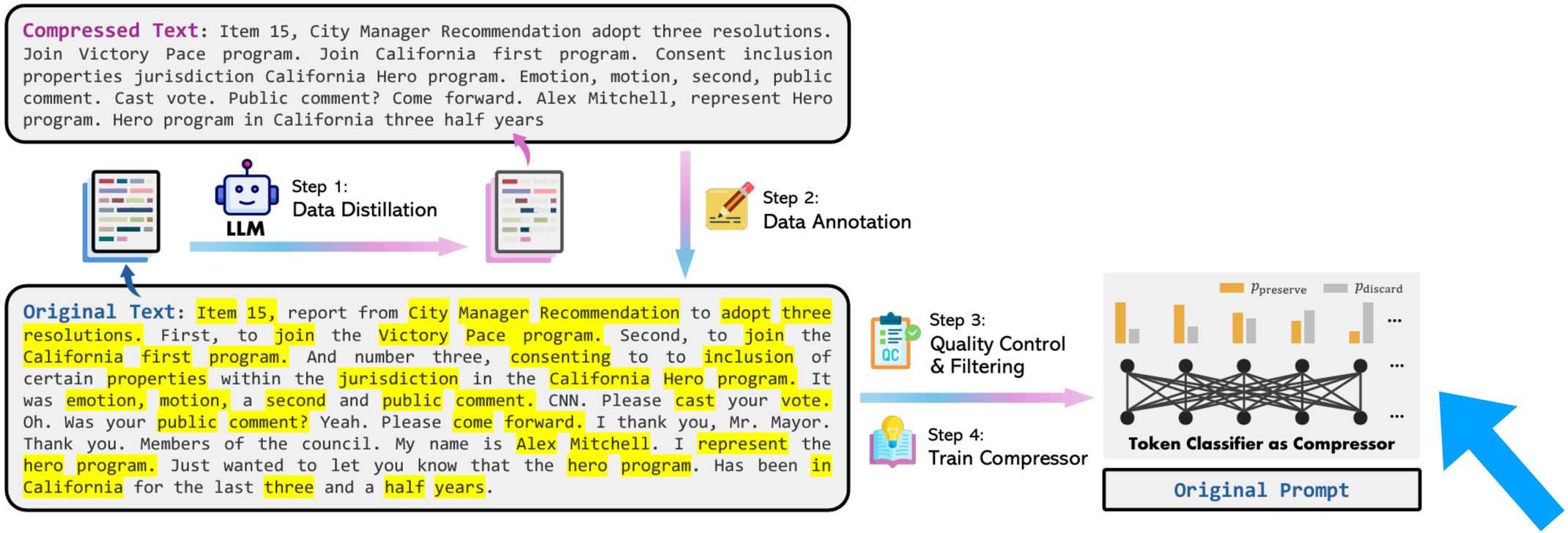
(i) Assess quality of compressed and of automatically annotated labels
(ii) Then filter examples by scores

Two quality control metrics:

1. Variation Rate: proportion of words in compressed text that are absent in original text
2. Alignment Gap: xxx

Pan, Z., Wu, Q., Jiang, H., Xia, M., Luo, X., Zhang, J., Lin, Q., Rühle, V., Yang, Y., Lin, C.Y. and Zhao, H.V., 2024. Llmlingua-2: Data distillation for efficient and faithful task-agnostic prompt compression. *arXiv preprint arXiv:2403.12968*.

Lexical-based Compression via Distillation

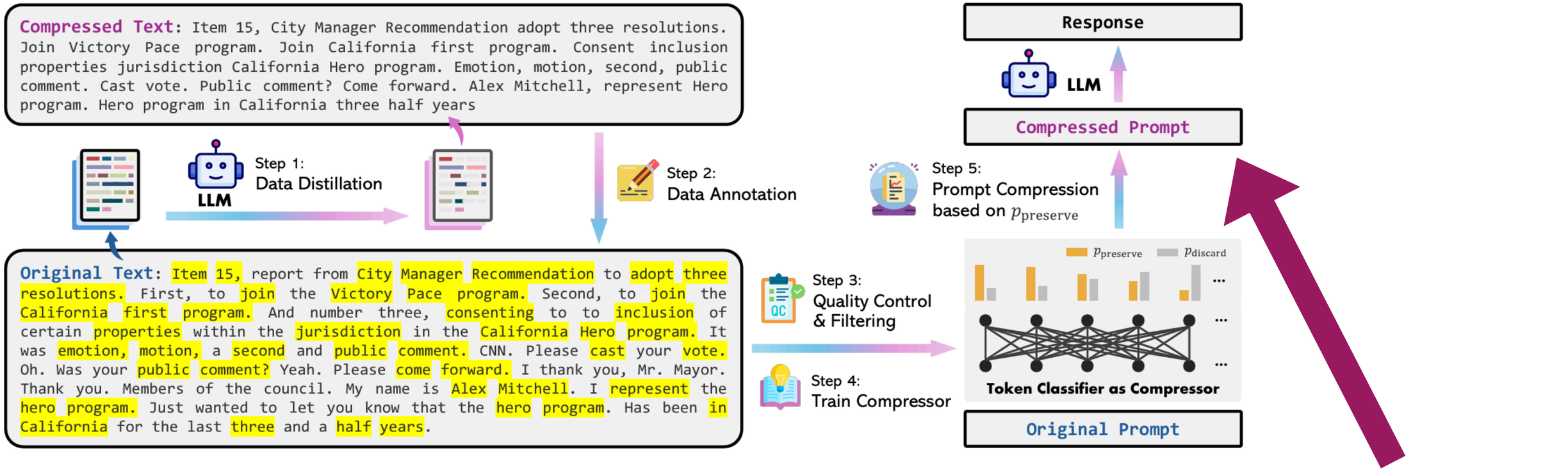


Step 4: Train Compressor (and then infer)

- Formulate prompt compression as binary token classification problem (i.e., preserve or discard)
- Smaller Transformer Encoder model (e.g. mBERT) allows for lower latency than using the larger LLM

Pan, Z., Wu, Q., Jiang, H., Xia, M., Luo, X., Zhang, J., Lin, Q., Rühle, V., Yang, Y., Lin, C.Y. and Zhao, H.V., 2024. Llmlingua-2: Data distillation for efficient and faithful task-agnostic prompt compression. *arXiv preprint arXiv:2403.12968*.

Lexical-based Compression via Distillation



Stage 5: Prompt Compression

1. use token classification model to predict probability of each word being labeled as preserve (p(word) = avg(p(tokens of word))
2. retain top words in original prompt with highest preservation probability, maintain their original order

Pan, Z., Wu, Q., Jiang, H., Xia, M., Luo, X., Zhang, J., Lin, Q., Rühle, V., Yang, Y., Lin, C.Y. and Zhao, H.V., 2024. Llmlingua-2: Data distillation for efficient and faithful task-agnostic prompt compression. *arXiv preprint arXiv:2403.12968*.

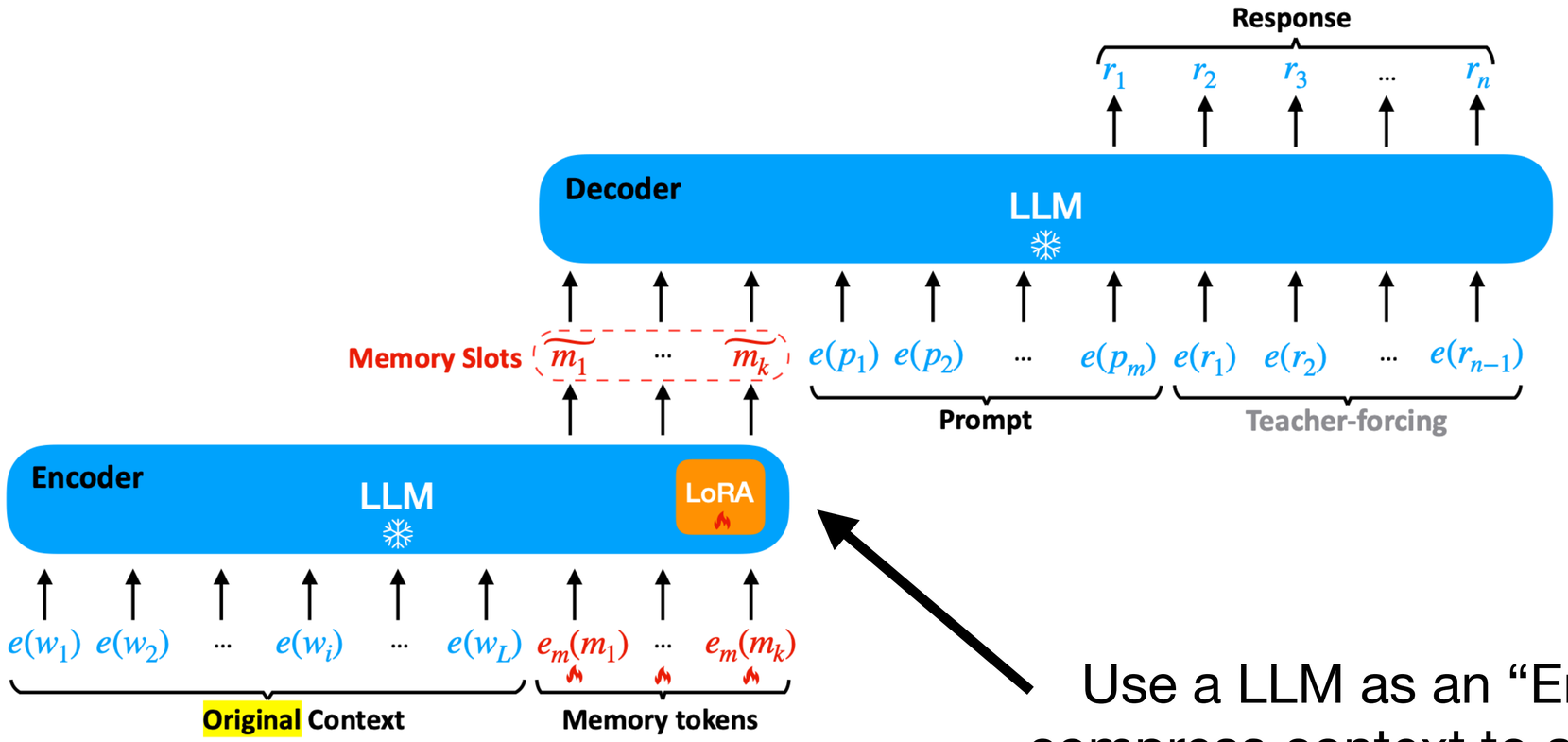
Embedding-based Compression: In-context Autoencoder (ICAE)

- Compress prompt into **special tokens** (short compact memory slots) that can be directly conditioned on by the LLM

- ~1% **additional parameters**, but 4× context compression

  ➡ Reduced latency and GPU memory cost during inference

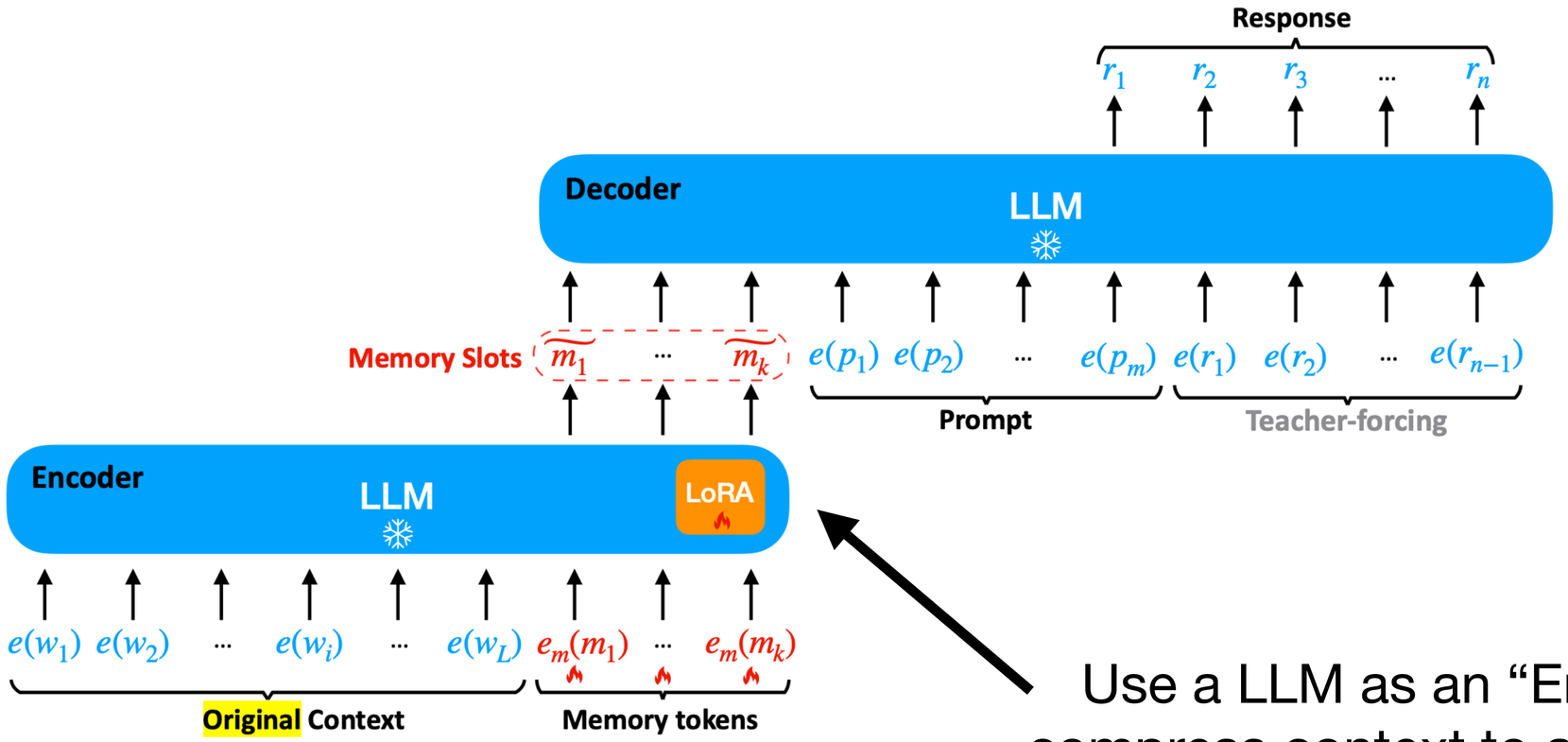
Embedding-based Compression: In-context Autoencoder (ICAE)



Use a LLM as an “Encoder” model to compress context to a few special tokens (memory slot tokens)

Ge, T., Jing, H., Wang, L., Wang, X., Chen, S.Q. and Wei, F., In-context Autoencoder for Context Compression in a Large Language Model. ICLR 2024

Embedding-based Compression: In-context Autoencoder (ICAE)

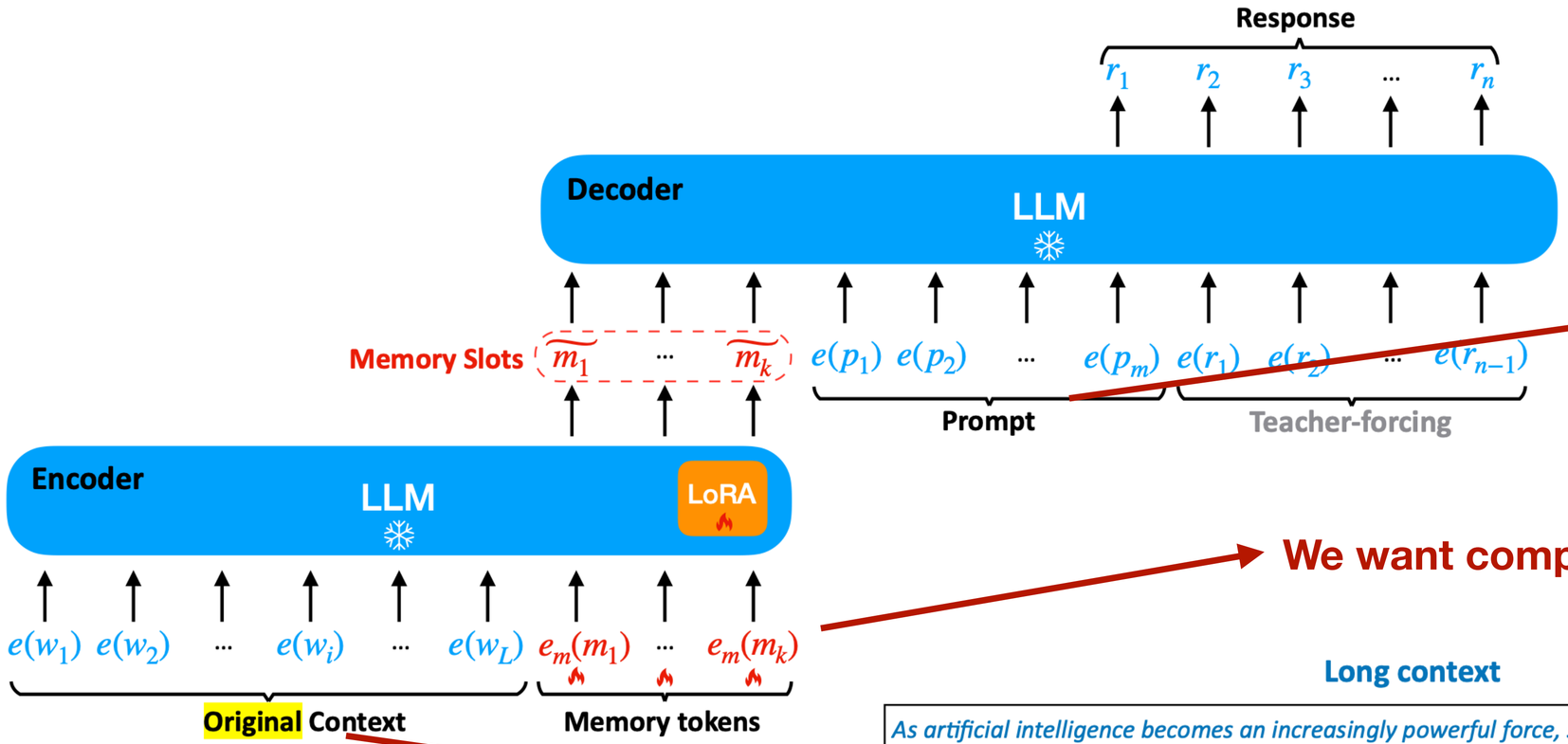


Use same LLM to "Decode" the memory slot and produce response

Use a LLM as an "Encoder" model to compress context to a few special tokens (memory slot tokens)

Ge, T., Jing, H., Wang, L., Wang, X., Chen, S.Q. and Wei, F., In-context Autoencoder for Context Compression in a Large Language Model. ICLR 2024

Embedding-based Compression: In-context Autoencoder (ICAE)



12 tokens

Prompt: List the companies that set up Partnership on AI research group

We want compression-rate = 32 -> 128 tokens

Long context

As artificial intelligence becomes an increasingly powerful force, some of the world's biggest companies are worrying about how the technology will be used ethically, and how the public will perceive its spread. To combat these problems (among others), five tech companies — Google, Amazon, Microsoft, Facebook, and IBM — set up a research group called the Partnership on AI.

...

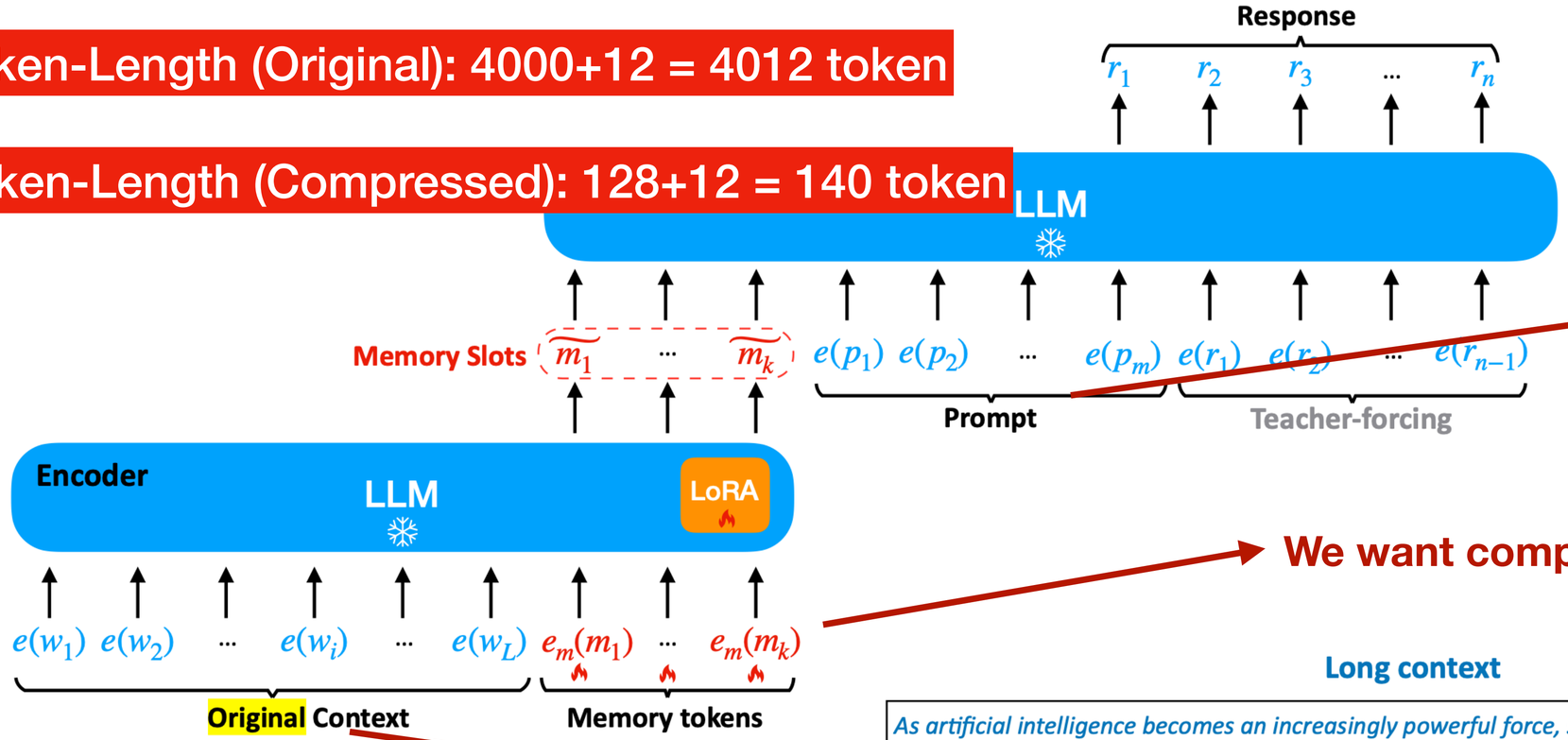
AI taking white collar jobs, eroding trust in public media, becoming embedded in public institutions like the courts and hospitals: these are the sorts of problems facing the industry in the future.

Ge, T., Jing, H., Wang, L., Wang, X., Chen, S.Q. and Wei, F., In-context Autoencoder for Context Compression in a Large Language Model. ICLR 2024

Embedding-based Compression: In-context Autoencoder (ICAE)

Token-Length (Original): 4000+12 = 4012 token

Token-Length (Compressed): 128+12 = 140 token



12 tokens

Prompt: List the companies that set up Partnership on AI research group

We want compression-rate = 32 -> 128 tokens

Long context

As artificial intelligence becomes an increasingly powerful force, some of the world's biggest companies are worrying about how the technology will be used ethically, and how the public will perceive its spread. To combat these problems (among others), five tech companies — Google, Amazon, Microsoft, Facebook, and IBM — set up a research group called the Partnership on AI.

...

AI taking white collar jobs, eroding trust in public media, becoming embedded in public institutions like the courts and hospitals: these are the sorts of problems facing the industry in the future.

Ge, T., Jing, H., Wang, L., Wang, X., Chen, S.Q. and Wei, F., In-context Autoencoder for Context Compression in a Large Language Model. ICLR 2024

Embedding-based Compression: In-context Autoencoder (ICAE)

- Compress prompt into **special tokens** (short compact memory slots) that can be directly conditioned on by the LLM

- ~1% **additional parameters**, but 4× context compression

  ➡ Reduced latency and GPU memory cost during inference

- Training:

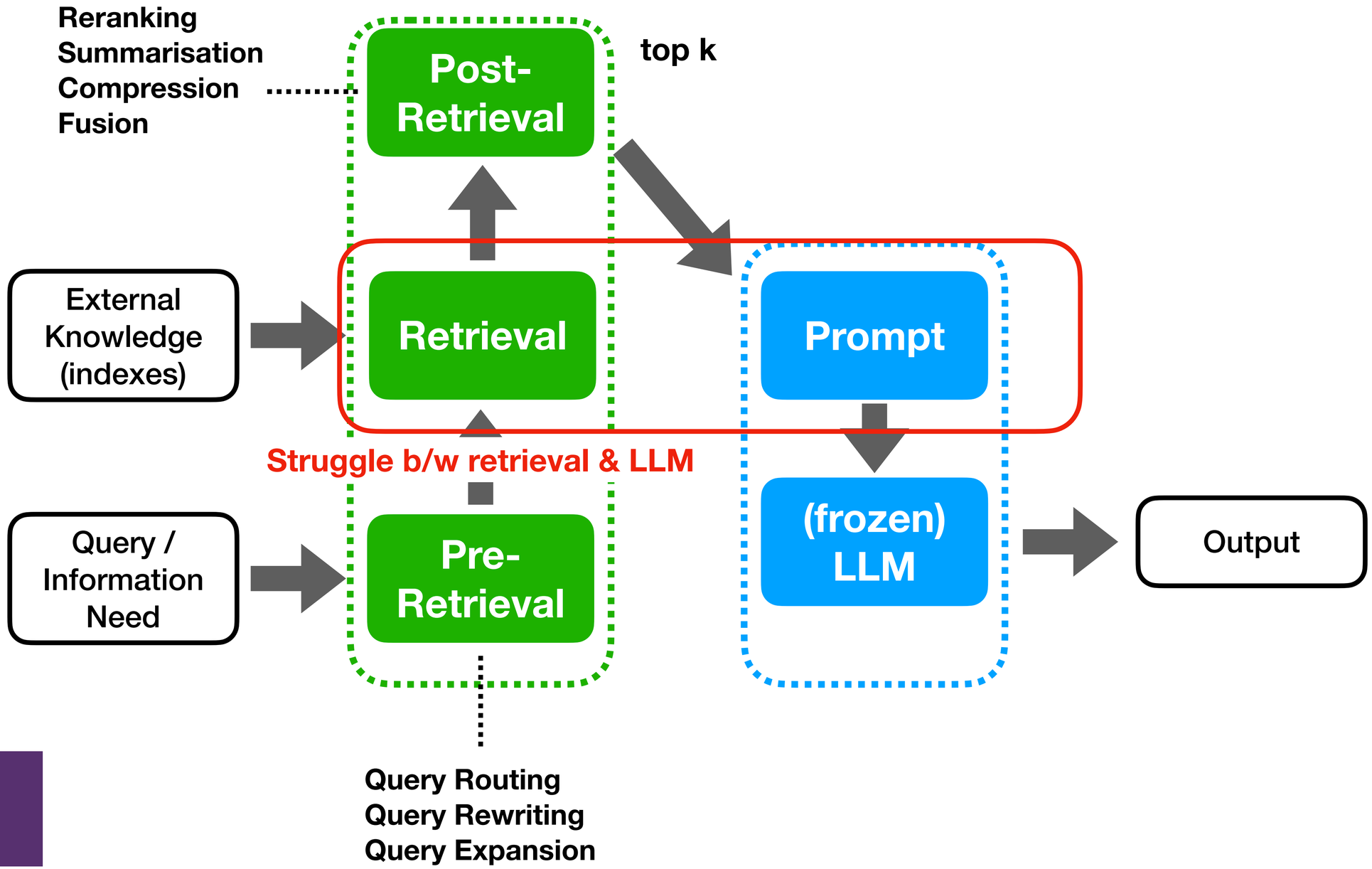
  1. pre-trained using auto-encoding and language modelling objectives on massive text data -> enables to generate memory slots that accurately and comprehensively represent original context.

  2. fine-tuned on instruction data -> produces desirable responses to various prompts.

ie lab

Ge, T., Jing, H., Wang, L., Wang, X., Chen, S.Q. and Wei, F., In-context Autoencoder for Context Compression in a Large Language Model. ICLR 2024

In the remaining of this part…



The struggle between model knowledge and retrieval knowledge

- The output of RAG is not guaranteed to be consistent with retrieved relevant passages
- Because the models are not explicitly trained to leverage and follow facts from provided passages.

ie lab

Gao, T., Yen, H., Yu, J. and Chen, D., 2023. Enabling Large Language Models to Generate Text with Citations. EMNLP 2023

Answers Inconsistent with RAG Evidence in Prompt

Question ····· *"Will drinking vinegar dissolve a stuck fish bone?"*

↓

```
[question_text]

Answer <Yes>, <No>, and provide an
explanation afterwards.
```

↓

↓

Answer

Koopman, B. and Zuccon, G., 2023. Dr ChatGPT tell me what I want to hear: How different prompts impact health answer correctness. EMNLP 2023

Answers Inconsistent with RAG Evidence in Prompt

Question ····· *"Will drinking vinegar dissolve a stuck fish bone?"*

```
[question_text]

Answer <Yes>, <No>, and provide an
explanation afterwards.
```

Answer

Ground truth answer

Koopman, B. and Zuccon, G., 2023. Dr ChatGPT tell me what I want to hear: How different prompts impact health answer correctness. EMNLP 2023

Answers Inconsistent with RAG Evidence in Prompt

Question ····· *"Will drinking vinegar dissolve a stuck fish bone?"*

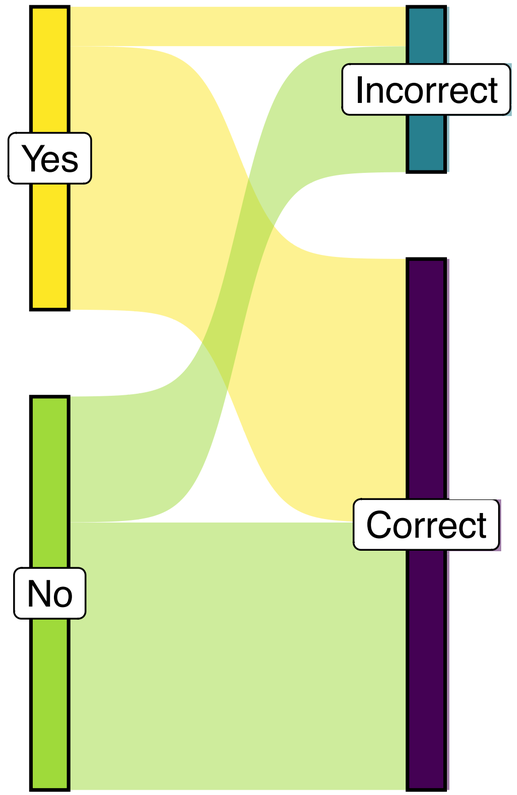
↓

```
[question_text]

Answer <Yes>, <No>, and provide an
explanation afterwards.
```

↓

↓

Answer



Ground truth answer

Question–only correctness

Koopman, B. and Zuccon, G., 2023. Dr ChatGPT tell me what I want to hear: How different prompts impact health answer correctness. EMNLP 2023

Answers Inconsistent with RAG Evidence in Prompt

Question ····· *"Will drinking vinegar dissolve a stuck fish bone?"*

↓

Retriever: top-1

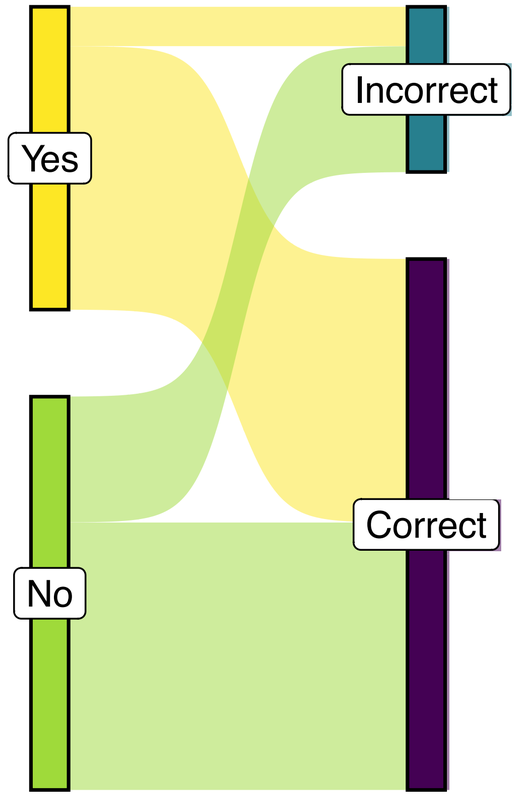
↓

```
[question_text]
A web search for this question has returned the
following evidence, which I provide to you in
quotes:
"[document_text]"
You MUST answer to my question with one of the
following options ONLY: <Yes>, <No>. Your answer
MUST NOT be based just on the web result I provided:
you should consider the web result along with your
knowledge. Please also provide an explanation for
your answer.
```

↓

↓

Answer



Ground truth answer

Question–only correctness

Answers Inconsistent with RAG Evidence in Prompt

Question ····· *"Will drinking vinegar dissolve a stuck fish bone?"*

↓

Retriever: top-1 ····· *We control this*

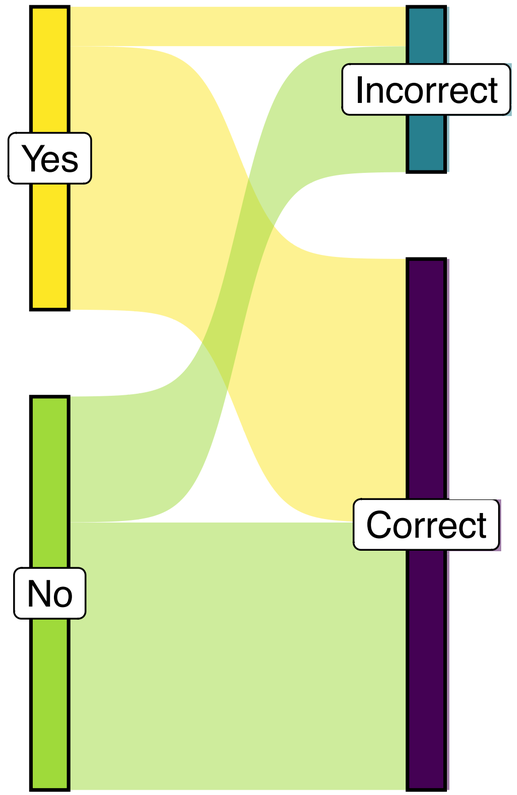
↓

```
[question_text]

A web search for this question has returned the
following evidence, which I provide to you in
quotes:

"[document_text]"

You MUST answer to my question with one of the
following options ONLY: <Yes>, <No>. Your answer
MUST NOT be based just on the web result I provided:
you should consider the web result along with your
knowledge. Please also provide an explanation for
your answer.
```

↓

↓

Answer



Ground truth answer

Question–only correctness

Answers Inconsistent with RAG Evidence in Prompt

Question ····· *"Will drinking vinegar dissolve a stuck fish bone?"*

Retriever: top-1 ····· *We control this*

```
[question_text]

A web search for this question has returned the
following evidence, which I provide to you in
quotes:

"[document_text]"

You MUST answer to my question with one of the
following options ONLY: <Yes>, <No>. Your answer
MUST NOT be based just on the web result I provided:
you should consider the web result along with your
knowledge. Please also provide an explanation for
your answer.
```

Answer

Supporting Evidence: provides the same answer as the ground-truth

Contrary Evidence: provides answer contrary to the ground-truth

Answers Inconsistent with RAG Evidence in Prompt

Question ····· *"Will drinking vinegar dissolve a stuck fish bone?"*

↓

Retriever: top-1 ····· *We control this*

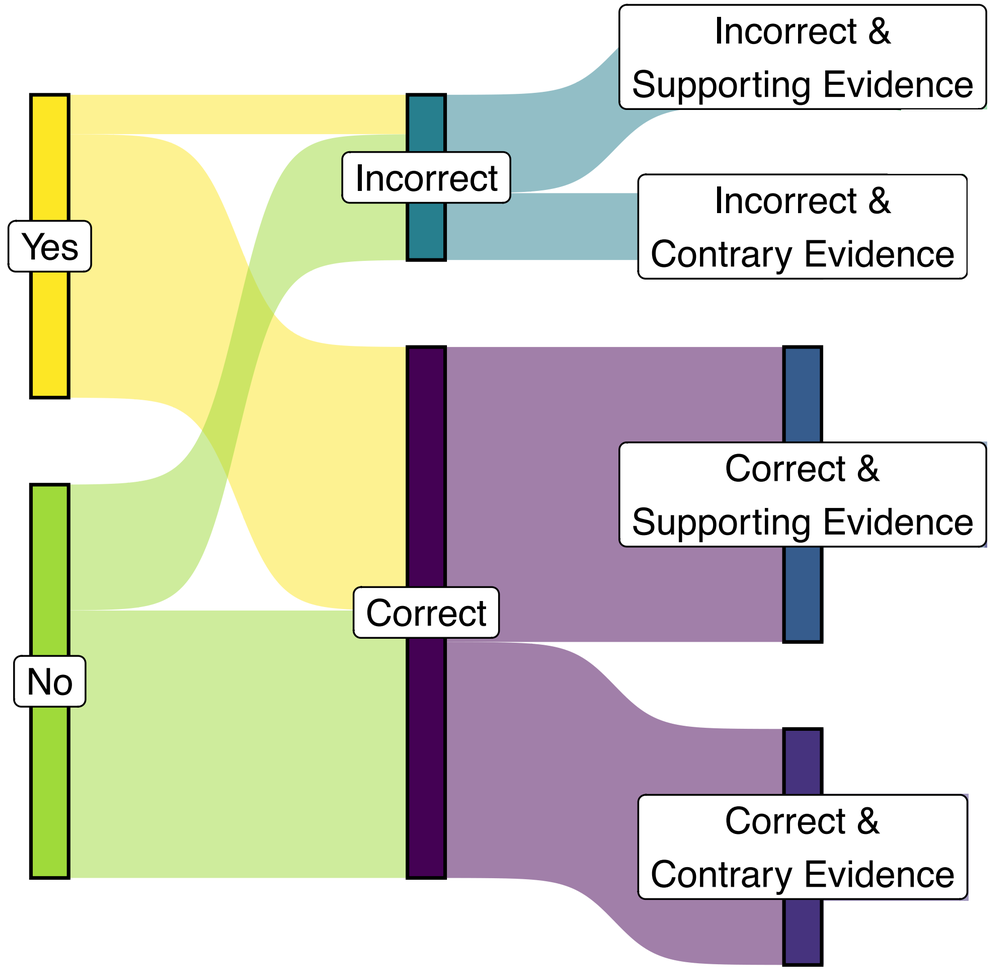
↓

```
[question_text]
A web search for this question has returned the following evidence, which I provide to you in quotes:
"[document_text]"
You MUST answer to my question with one of the following options ONLY: <Yes>, <No>. Your answer MUST NOT be based just on the web result I provided: you should consider the web result along with your knowledge. Please also provide an explanation for your answer.
```

↓

↓

Answer



Answers Inconsistent with RAG Evidence in Prompt

Question ····· *"Will drinking vinegar dissolve a stuck fish bone?"*

↓

Retriever: top-1 ····· *We control this*

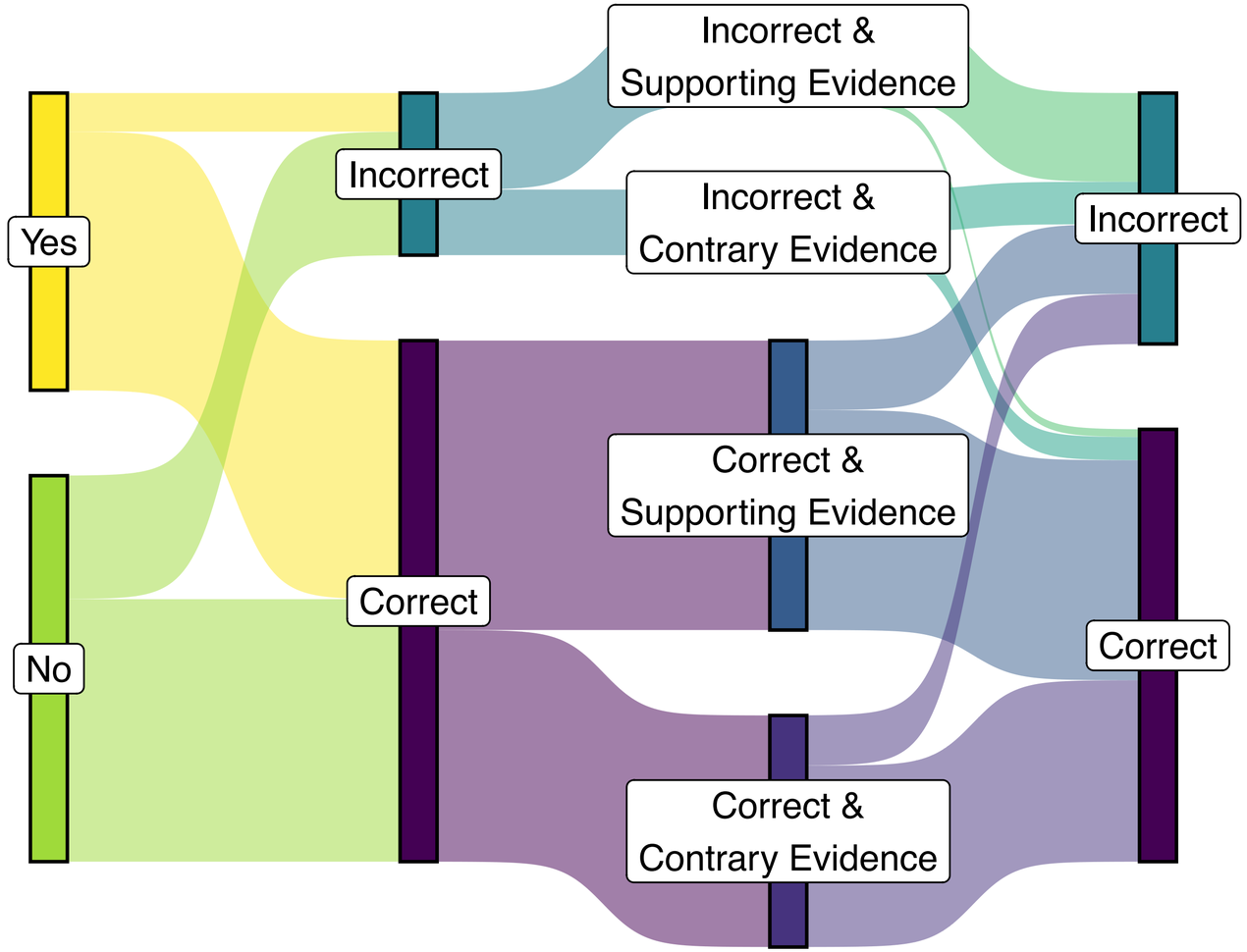
↓

```
[question_text]
A web search for this question has returned the
following evidence, which I provide to you in
quotes:
"[document_text]"
You MUST answer to my question with one of the
following options ONLY: <Yes>, <No>. Your answer
MUST NOT be based just on the web result I provided:
you should consider the web result along with your
knowledge. Please also provide an explanation for
your answer.
```

↓

↓

Answer



Koopman, B. and Zuccon, G., 2023. Dr ChatGPT tell me what I want to hear: How different prompts impact health answer correctness. EMNLP 2023

Answers Inconsistent with RAG Evidence in Prompt

Question ····· *"Will drinking vinegar dissolve a stuck fish bone?"*

↓

Retriever: top-1 ····· *We control this*

↓

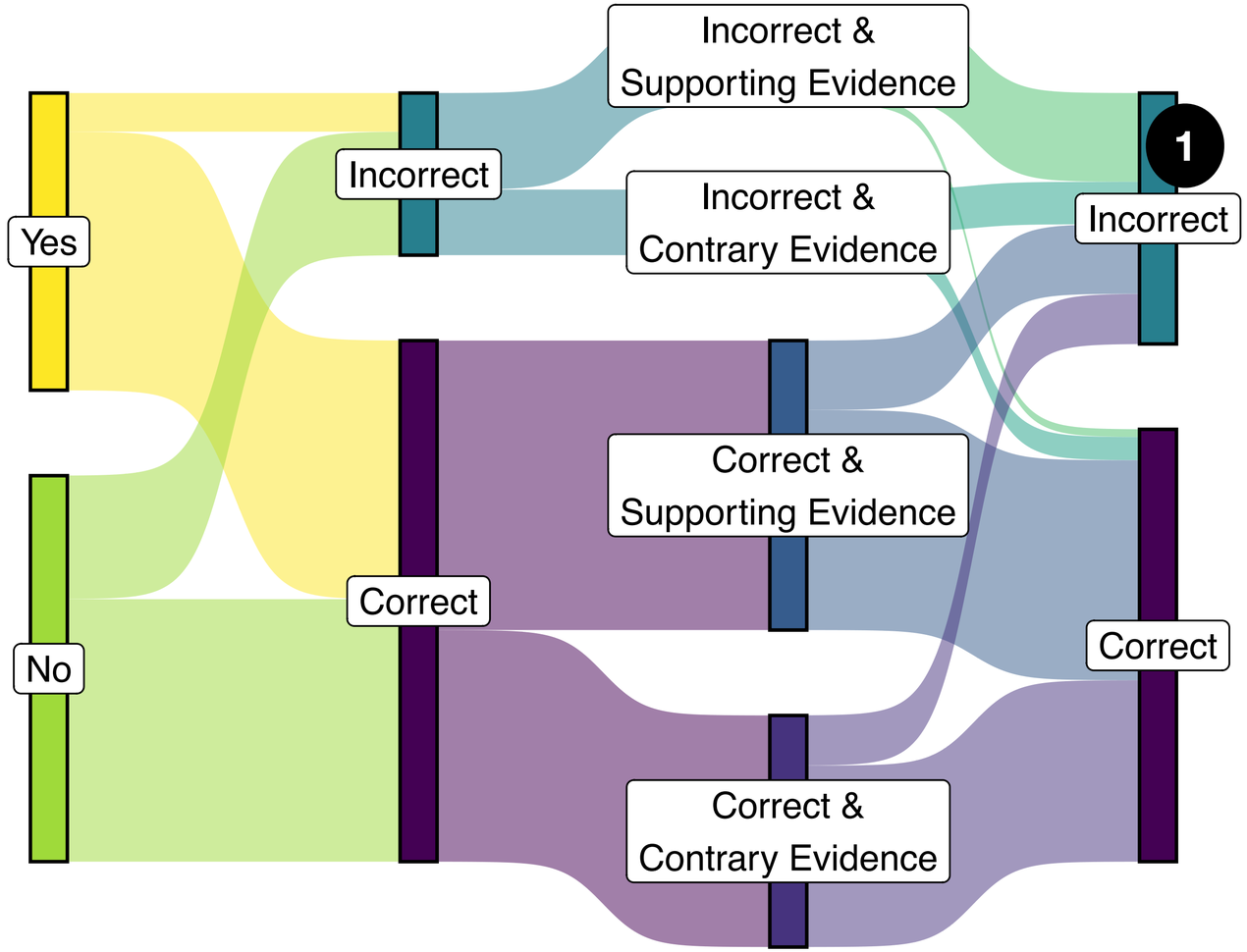
```
[question_text]
A web search for this question has returned the
follow
quotes
"[docu
You MU
follow
MUST N
you sh
knowledge. Please also provide an explanation for
your answer.
```

1) providing supporting evidence to try to overturn an incorrect answer does not flip it to a correct answer.

↓

↓

Answer



Koopman, B. and Zuccon, G., 2023. Dr ChatGPT tell me what I want to hear: How different prompts impact health answer correctness. EMNLP 2023

Answers Inconsistent with RAG Evidence in Prompt

Question ····· *"Will drinking vinegar dissolve a stuck fish bone?"*

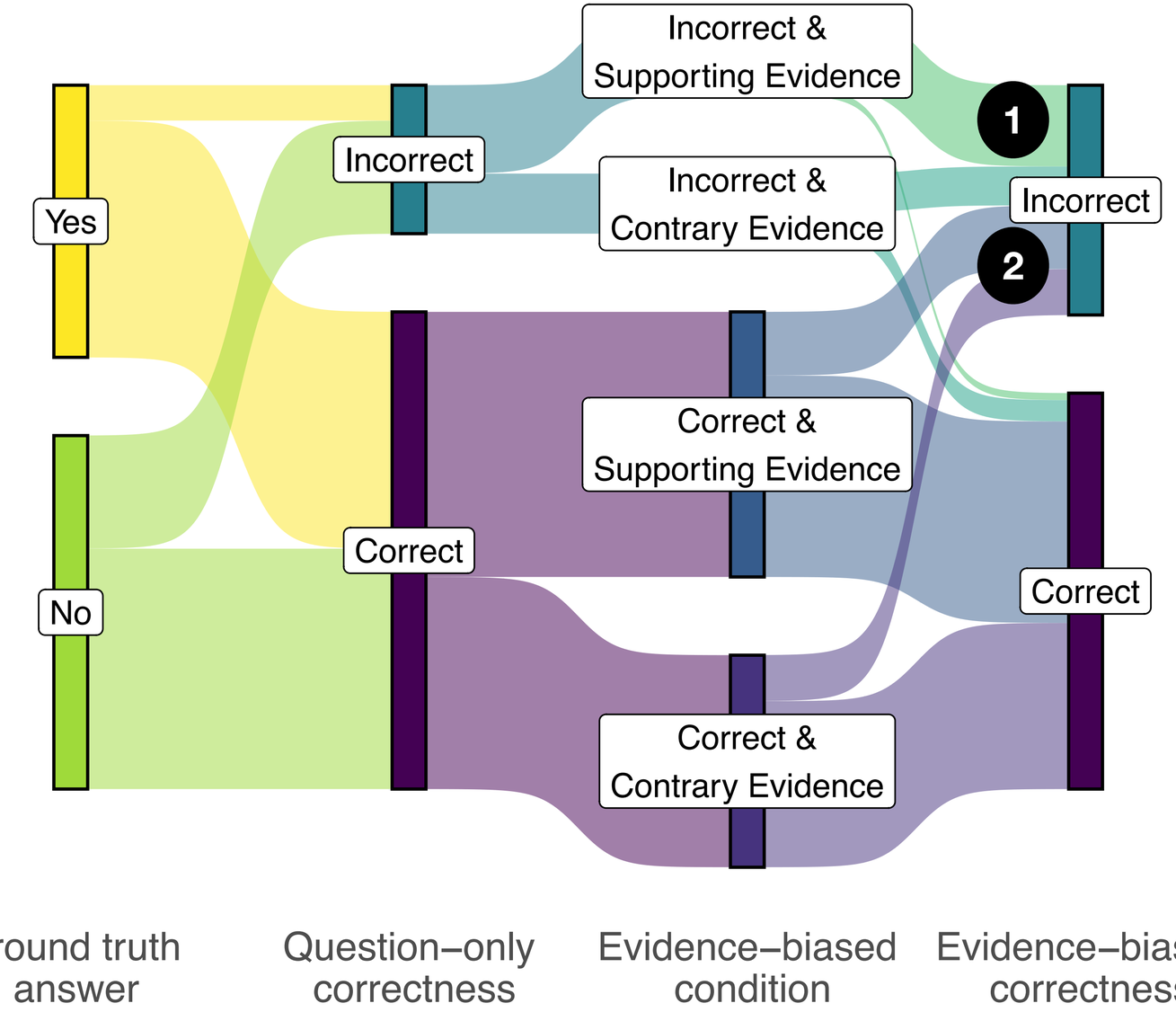
Retriever: top-1 ····· *We control this*

```
[question_text]
A web search for this question has returned the
follow
quotes
"[docu
You MU
follow
MUST N
you sh
knowledge. Please also provide an explanation for
your answer.
```

1) providing supporting evidence to try to overturn an incorrect answer does not flip it to a correct answer.

2) providing evidence caused the answer to flip from correct to incorrect

Answer



Koopman, B. and Zuccon, G., 2023. Dr ChatGPT tell me what I want to hear: How different prompts impact health answer correctness. EMNLP 2023

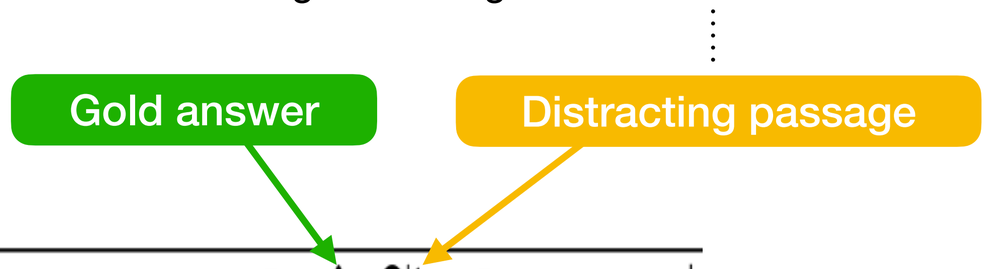
The Power of Noise in RAG

- What the type of evidence an IR systems in RAG should retrieve?
- Cuconasu et al. analysed effect of **relevance**, **position**, and **amount** of evidence in prompts on RAG effectiveness.

Cuconasu, F., Trappolini, G., Siciliano, F., Filice, S., Campagnano, C., Maarek, Y., Tonellotto, N. and Silvestri, F., 2024. The power of noise: Redefining retrieval for rag systems. *arXiv preprint arXiv:2401.14887*.

The Power of Noise in RAG

Passages with high retrieval scores but not containing the answer



	Far - [I, ★, 🌟, Q]			
# 🌟	Llama2	MPT	Phi-2	Falcon
0	**0.5642**	0.2148	**0.4438**	**0.4330**
1	0.4586	0.1976	0.3585	0.3469
2	0.3455	0.1913	0.3430	0.3246
4	0.2745	**0.2209***	0.3019	0.2670
6	0.2898	0.2171*	0.2943	0.2392
8	0.2643	0.2077*	0.2513	0.1878
10	0.2537	-	-	-
12	0.2688	-	-	-
14	0.2583	-	-	-
16	0.2413	-	-	-
18	0.2348	-	-	-

Cuconasu, F., Trappolini, G., Siciliano, F., Filice, S., Campagnano, C., Maarek, Y., Tonellotto, N. and Silvestri, F., 2024. The power of noise: Redefining retrieval for rag systems. *arXiv preprint arXiv:2401.14887*.

The Power of Noise in RAG

Passages with high retrieval scores but not containing the answer

Gold answer

Distracting passage

	Far - [I, ★, 🌀, Q]			
# 🌀	Llama2	MPT	Phi-2	Falcon
0	**0.5642**	0.2148	**0.4438**	**0.4330**
1	0.4586	0.1976	0.3585	0.3469
2	0.3455	0.1913	0.3430	0.3246
4	0.2745	**0.2209***	0.3019	0.2670
6	0.2898	0.2171*	0.2943	0.2392
8	0.2643	0.2077*	0.2513	0.1878
10	0.2537	-	-	-
12	0.2688	-	-	-
14	0.2583	-	-	-
16	0.2413	-	-	-
18	0.2348	-	-	-

1. progressive accuracy degradation as the number of distracting documents included in the context increases

Cuconasu, F., Trappolini, G., Siciliano, F., Filice, S., Campagnano, C., Maarek, Y., Tonellotto, N. and Silvestri, F., 2024. The power of noise: Redefining retrieval for rag systems. *arXiv preprint arXiv:2401.14887*.

The Power of Noise in RAG

Passages with high retrieval scores but not containing the answer

Gold answer

Distracting passage

	Far - [I, ★, 🌀, Q]			
# 🌀	Llama2	MPT	Phi-2	Falcon
0	**0.5642**	0.2148	**0.4438**	**0.4330**
1	0.4586	0.1976	0.3585	0.3469
2	0.3455	0.1913	0.3430	0.3246
4	0.2745	**0.2209***	0.3019	0.2670
6	0.2898	0.2171*	0.2943	0.2392
8	0.2643	0.2077*	0.2513	0.1878
10	0.2537	-	-	-
12	0.2688	-	-	-
14	0.2583	-	-	-
16	0.2413	-	-	-
18	0.2348	-	-	-

1. progressive accuracy degradation as the number of distracting documents included in the context increases

2. adding just one distracting document causes sharp reduction

Cuconasu, F., Trappolini, G., Siciliano, F., Filice, S., Campagnano, C., Maarek, Y., Tonellotto, N. and Silvestri, F., 2024. The power of noise: Redefining retrieval for rag systems. *arXiv preprint arXiv:2401.14887*.

The Power of Noise in RAG

Passages with high retrieval scores but not containing the answer

Gold answer

Distracting passage

	Far - [I, ★, 🌀, Q]			
# 🌀	Llama2	MPT	Phi-2	Falcon
0	**0.5642**	0.2148	**0.4438**	**0.4330**
1	0.4586	0.1976	0.3585	0.3469
2	0.3455	0.1913	0.3430	0.3246
4	0.2745	**0.2209***	0.3019	0.2670
6	0.2898	0.2171*	0.2943	0.2392
8	0.2643	0.2077*	0.2513	0.1878
10	0.2537	-	-	-
12	0.2688	-	-	-
14	0.2583	-	-	-
16	0.2413	-	-	-
18	0.2348	-	-	-

1. progressive accuracy degradation as the number of distracting documents included in the context increases

2. adding just one distracting document causes sharp reduction

Takeaway: introducing semantically aligned yet non-relevant documents potentially misguides LLMs

The Power of Noise in RAG

Passages with high retrieval scores but not containing the answer

Gold answer

Distracting passage

Gold answer first, in middle, last

	Far - [I, ★, 🌀, Q]				Mid - [I, 🌀, ★, 🌀, Q]				Near - [I, 🌀, ★, Q]			
# 🌀	Llama2	MPT	Phi-2	Falcon	Llama2	MPT	Phi-2	Falcon	Llama2	MPT	Phi-2	Falcon
0	**0.5642**	0.2148	**0.4438**	**0.4330**	**0.5642**	**0.2148**	**0.4438**	**0.4330**	**0.5642**	**0.2148**	**0.4438**	**0.4330**
1	0.4586	0.1976	0.3585	0.3469	no-mid	no-mid	no-mid	no-mid	0.4283	0.1791	0.4227	0.3602
2	0.3455	0.1913	0.3430	0.3246	0.3322	0.1802	0.3375	0.2823	0.3974	0.2002	0.3975	0.3111
4	0.2745	**0.2209***	0.3019	0.2670	0.2857	0.1775	0.2885	0.2378	0.3795	0.2059*	0.3701	0.2736
6	0.2898	0.2171*	0.2943	0.2392	0.2698	0.1424	0.2625	0.2103	0.3880	0.1892	0.3623	0.2656
8	0.2643	0.2077*	0.2513	0.1878	0.2268	0.1002	0.2360	0.1745	0.3748	0.1944	0.3423	0.2424
10	0.2537	-	-	-	0.2180	-	-	-	0.3716	-	-	-
12	0.2688	-	-	-	0.2382	-	-	-	0.3991	-	-	-
14	0.2583	-	-	-	0.2280	-	-	-	0.4118	-	-	-
16	0.2413	-	-	-	0.2024	-	-	-	0.3889	-	-	-
18	0.2348	-	-	-	0.1795	-	-	-	0.3781	-	-	-

The Power of Noise in RAG

Passages with high retrieval scores but not containing the answer

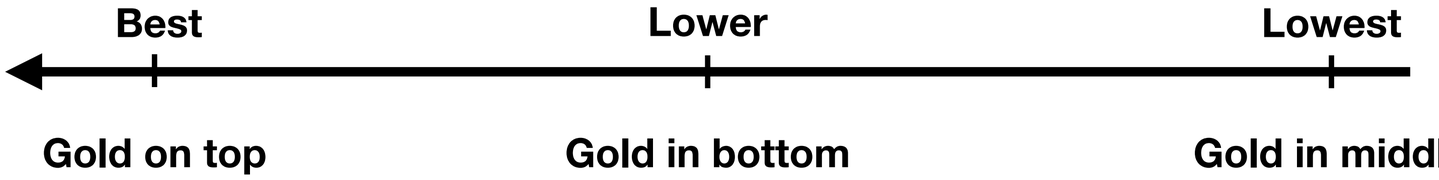
Gold answer

Distracting passage

Gold answer first, in middle, last

	Far - [I, ★, 🌀, Q]				Mid - [I, 🌀, ★, 🌀, Q]				Near - [I, 🌀, ★, Q]			
# 🌀	Llama2	MPT	Phi-2	Falcon	Llama2	MPT	Phi-2	Falcon	Llama2	MPT	Phi-2	Falcon
0	**0.5642**	0.2148	**0.4438**	**0.4330**	**0.5642**	**0.2148**	**0.4438**	**0.4330**	**0.5642**	**0.2148**	**0.4438**	**0.4330**
1	0.4586	0.1976	0.3585	0.3469	no-mid	no-mid	no-mid	no-mid	0.4283	0.1791	0.4227	0.3602
2	0.3455	0.1913	0.3430	0.3246	0.3322	0.1802	0.3375	0.2823	0.3974	0.2002	0.3975	0.3111
4	0.2745	**0.2209***	0.3019	0.2670	0.2857	0.1775	0.2885	0.2378	0.3795	0.2059*	0.3701	0.2736
6	0.2898	0.2171*	0.2943	0.2392	0.2698	0.1424	0.2625	0.2103	0.3880	0.1892	0.3623	0.2656
8	0.2643											0.2424
10	0.2537											-
12	0.2688											-
14	0.2583											-
16	0.2413											-
18	0.2348											-

Lost in the Middle Effect



Liu, N.F., Lin, K., Hewitt, J., Paranjape, A., Bevilacqua, M., Petroni, F. and Liang, P., 2024. Lost in the middle: How language models use long contexts. *TACL*

The Power of Noise in RAG

Gold answer

Random passage

	Far - [I, ★, 🎲, Q]				Mid - [I, 🎲, ★, 🎲, Q]				Near - [I, 🎲, ★, Q]			
# 🎲	Llama2	MPT	Phi-2	Falcon	Llama2	MPT	Phi-2	Falcon	Llama2	MPT	Phi-2	Falcon
0	**0.5642**	0.2148	**0.4438**	**0.4330**	**0.5642**	0.2148	**0.4438**	**0.4330**	0.5642	0.2148	0.4438	**0.4330**
1	0.4733	0.2447	0.4329	0.4035	no-mid	no-mid	no-mid	no-mid	0.4862	0.2125*	0.4587	0.4091
2	0.3776	0.2639	0.4249	0.3805	0.3928	**0.2584**	0.4293	0.3612	0.5032	0.2660	**0.4614**	0.3912
4	0.3109	0.2933	0.4091	0.3468	0.3998	0.2577	0.3985	0.3462	0.5221	**0.2930**	0.4311	0.3949
6	0.3547	0.3036	0.4130	0.3250	0.4138	0.2265	0.3891	0.3196	0.5681*	0.2890	0.4388	0.3908
8	0.3106	**0.3039**	0.3812	0.2543	0.3734	0.1566	0.3596	0.2767	0.5609*	0.2911	0.4258	0.3704
10	0.3390	-	-	-	0.3675	-	-	-	0.5579*	-	-	-
12	0.3736	-	-	-	0.3641	-	-	-	0.5836	-	-	-
14	0.3527	-	-	-	0.3372	-	-	-	**0.5859**	-	-	-
16	0.3401	-	-	-	0.3159	-	-	-	0.5722	-	-	-
18	0.3466	-	-	-	0.2982	-	-	-	0.5588*	-	-	-

The Power of Noise in RAG

Gold answer

Random passage

Can observe improvements in effectiveness

	Far - [I, ★, 🎲, Q]				Mid - [I, 🎲, ★, 🎲, Q]				Near - [I, 🎲, ★, Q]			
# 🎲	Llama2	MPT	Phi-2	Falcon	Llama2	MPT	Phi-2	Falcon	Llama2	MPT	Phi-2	Falcon
0	**0.5642**	0.2148	**0.4438**	**0.4330**	**0.5642**	0.2148	**0.4438**	**0.4330**	0.5642	0.2148	0.4438	**0.4330**
1	0.4733	0.2447	0.4329	0.4035	no-mid	no-mid	no-mid	no-mid	0.4862	0.2125*	0.4587	0.4091
2	0.3776	0.2639	0.4249	0.3805	0.3928	**0.2584**	0.4293	0.3612	0.5032	0.2660	**0.4614**	0.3912
4	0.3109	0.2933	0.4091	0.3468	0.3998	0.2577	0.3985	0.3462	0.5221	**0.2930**	0.4311	0.3949
6	0.3547	0.3036	0.4130	0.3250	0.4138	0.2265	0.3891	0.3196	0.5681*	0.2890	0.4388	0.3908
8	0.3106	**0.3039**	0.3812	0.2543	0.3734	0.1566	0.3596	0.2767	0.5609*	0.2911	0.4258	0.3704
10	0.3390	-	-	-	0.3675	-	-	-	0.5579*	-	-	-
12	0.3736	-	-	-	0.3641	-	-	-	0.5836	-	-	-
14	0.3527	-	-	-	0.3372	-	-	-	**0.5859**	-	-	-
16	0.3401	-	-	-	0.3159	-	-	-	0.5722	-	-	-
18	0.3466	-	-	-	0.2982	-	-	-	0.5588*	-	-	-

The Power of Noise in RAG

Gold answer

Random passage

Can observe improvements in effectiveness

	Far - [I, ★, 🎲, Q]				Mid - [I, 🎲, ★, 🎲, Q]				Near - [I, 🎲, ★, Q]			
# 🎲	Llama2	MPT	Phi-2	Falcon	Llama2	MPT	Phi-2	Falcon	Llama2	MPT	Phi-2	Falcon
0	**0.5642**	0.2148	**0.4438**	**0.4330**	**0.5642**	0.2148	**0.4438**	**0.4330**	0.5642	0.2148	0.4438	**0.4330**
1	0.4733	0.2447	0.4329	0.4035	no-mid	no-mid	no-mid	no-mid	0.4862	0.2125*	0.4587	0.4091
2	0.3776	0.2639	0.4249	0.3805	0.3928	**0.2584**	0.4293	0.3612	0.5032	0.2660	**0.4614**	0.3912
4	0.3109	0.2933	0.4091	0.3468	0.3998	0.2577	0.3985	0.3462	0.5221	**0.2930**	0.4311	0.3949
6	0.3547	0.3036	0.4130	0.3250	0.4138	0.2265	0.3891	0.3196	0.5681*	0.2890	0.4388	0.3908
8	0.3106	**0.3039**	0.3812	0.2543	0.3734	0.1566	0.3596	0.2767	0.5609*	0.2911	0.4258	0.3704
10	0.3390	-	-	-	0.3675	-	-	-	0.5579*	-	-	-
12	0.3736	-	-	-	0.3641	-	-	-	0.5836	-	-	-
14	0.3527	-	-	-	0.3372	-	-	-	**0.5859**	-	-	-
16	0.3401	-	-	-	0.3159	-	-	-	0.5722	-	-	-
18	0.3466	-	-	-	0.2982	-	-	-	0.5588*	-	-	-

different LLMs
=
different behaviours

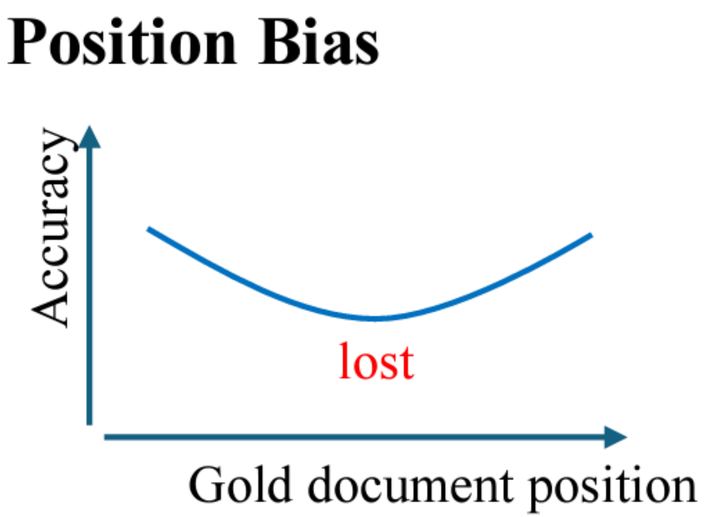
The Power of Noise in RAG

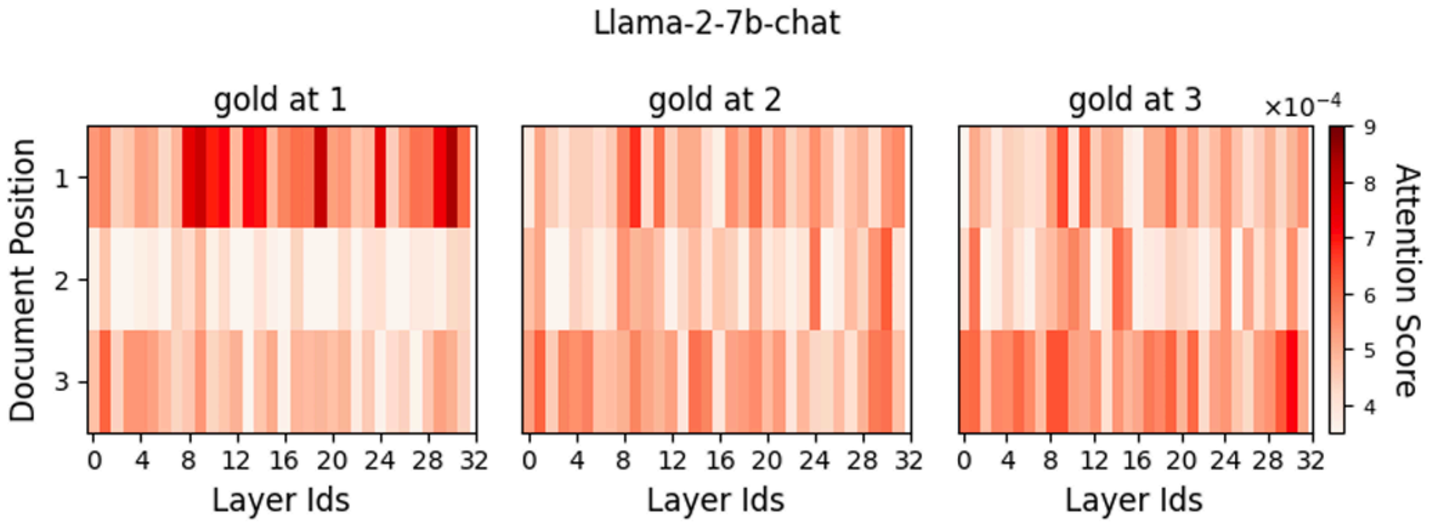
- What the type of evidence an IR systems in RAG should retrieve?
- Cuconasu et al. analysed effect of **relevance**, **position**, and **amount** of evidence in prompts on RAG effectiveness.
- Key findings:
    1. retriever's **highest scoring** documents **not directly relevant** to query (i.e. do not contain answer) **negatively** impact LLM effectiveness
    2. lost in the **middle** effect
    3. adding **random** documents in prompt **improves** LLM effectiveness by up to 35%

ie lab

Cuconasu, F., Trappolini, G., Siciliano, F., Filice, S., Campagnano, C., Maarek, Y., Tonellotto, N. and Silvestri, F., 2024. The power of noise: Redefining retrieval for rag systems. *arXiv preprint arXiv:2401.14887*.

Attention Instruction: Pay Attention to the Middle





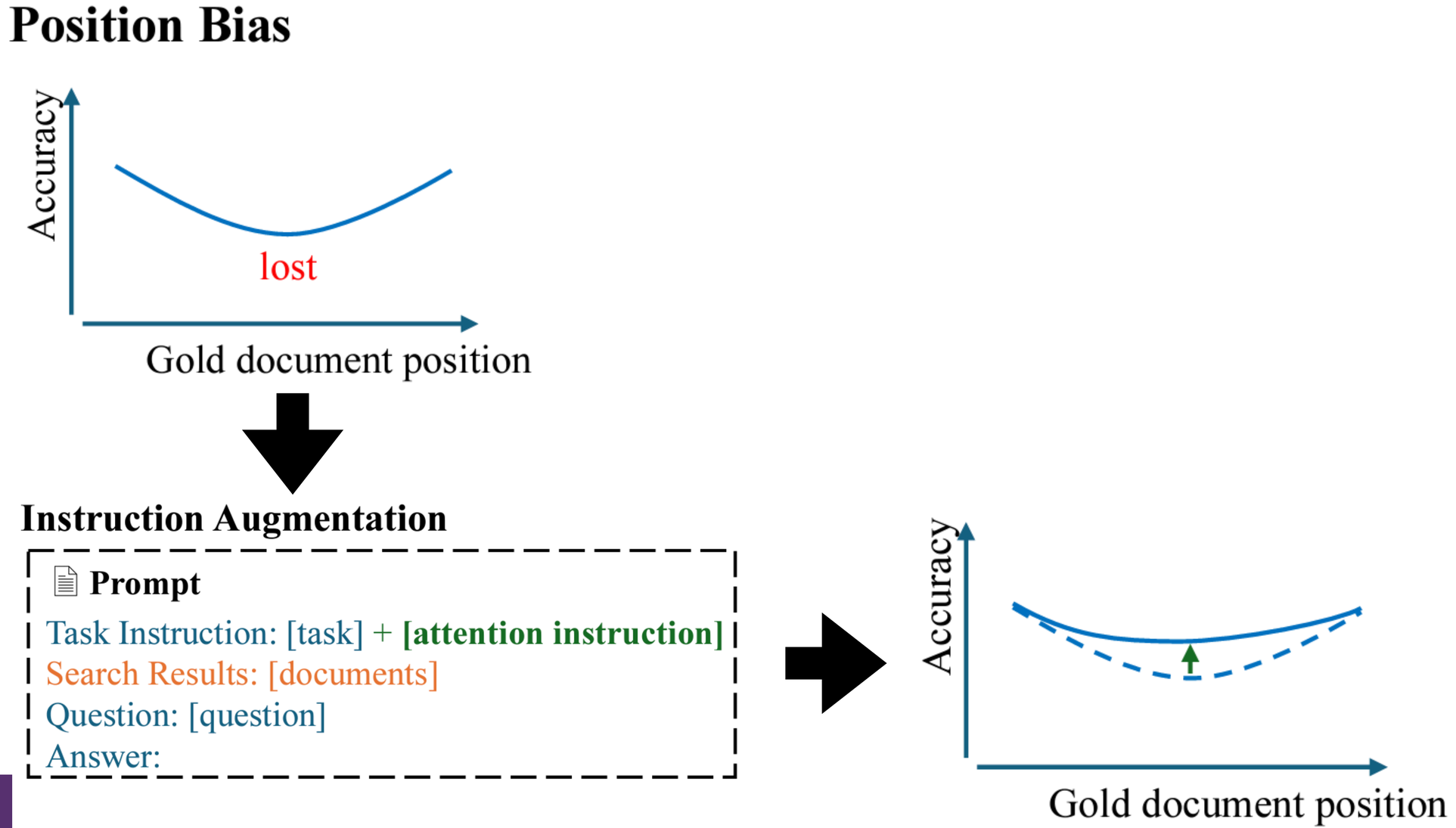
- LLMs suffer of position bias — for RAG, this means position bias among search results
- Position bias is confirmed by attention scores: significant drop in the middle
- Difficulty in accessing and using middle part of context due to lack of attention

ie lab

Zhang, M., Meng, Z. and Collier, N., 2024. Attention Instruction: Amplifying Attention in the Middle via Prompting. *arXiv preprint arXiv:2406.17095*.

The University Of Queensland AUSTRALIA
CREATE CHANGE

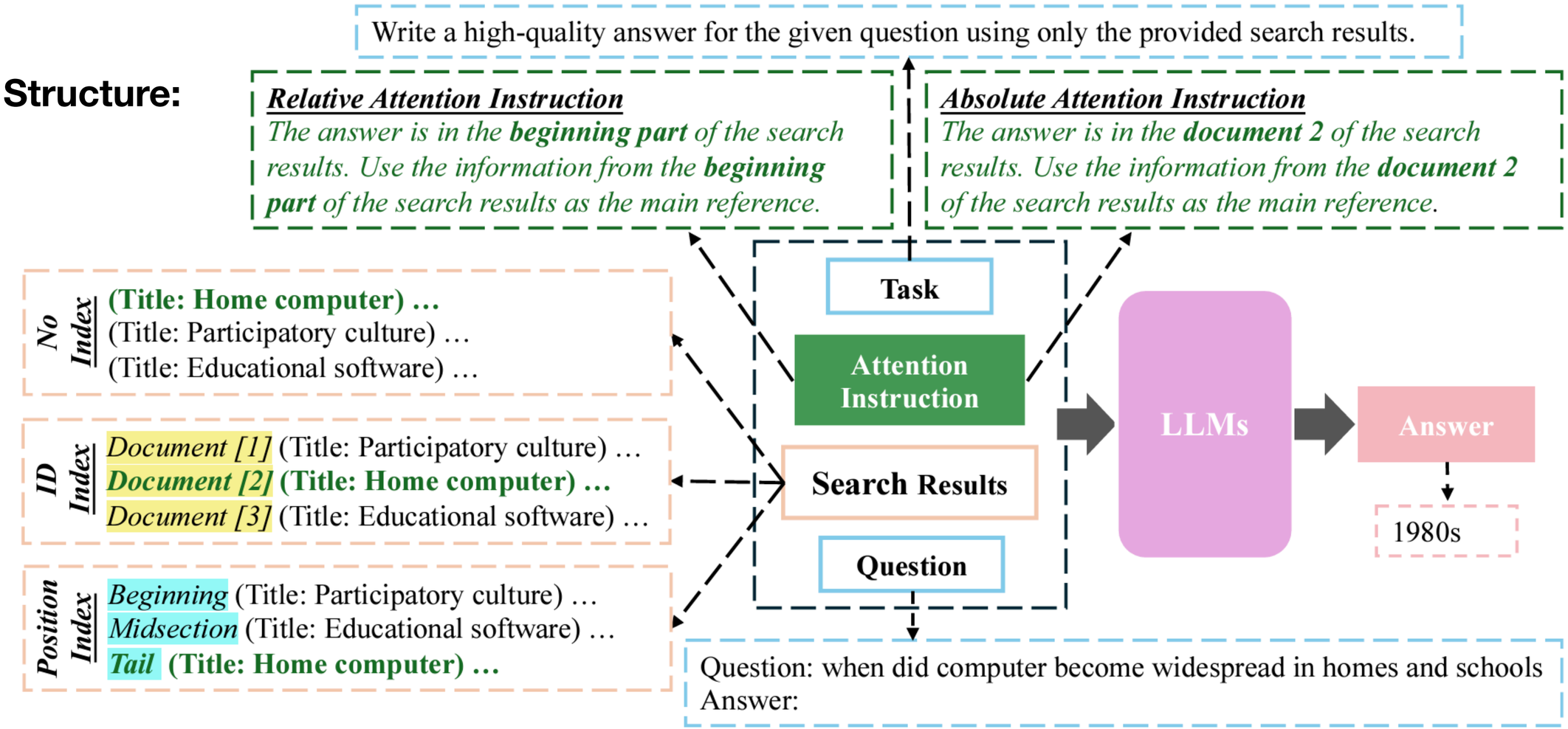
Attention Instruction: Pay Attention to the Middle



Zhang, M., Meng, Z. and Collier, N., 2024. Attention Instruction: Amplifying Attention in the Middle via Prompting. *arXiv preprint arXiv:2406.17095*.

Attention Instruction: Pay Attention to the Middle

Prompt Structure:



Zhang, M., Meng, Z. and Collier, N., 2024. Attention Instruction: Amplifying Attention in the Middle via Prompting. *arXiv preprint arXiv:2406.17095*.

Attention Instruction: Pay Attention to the Middle

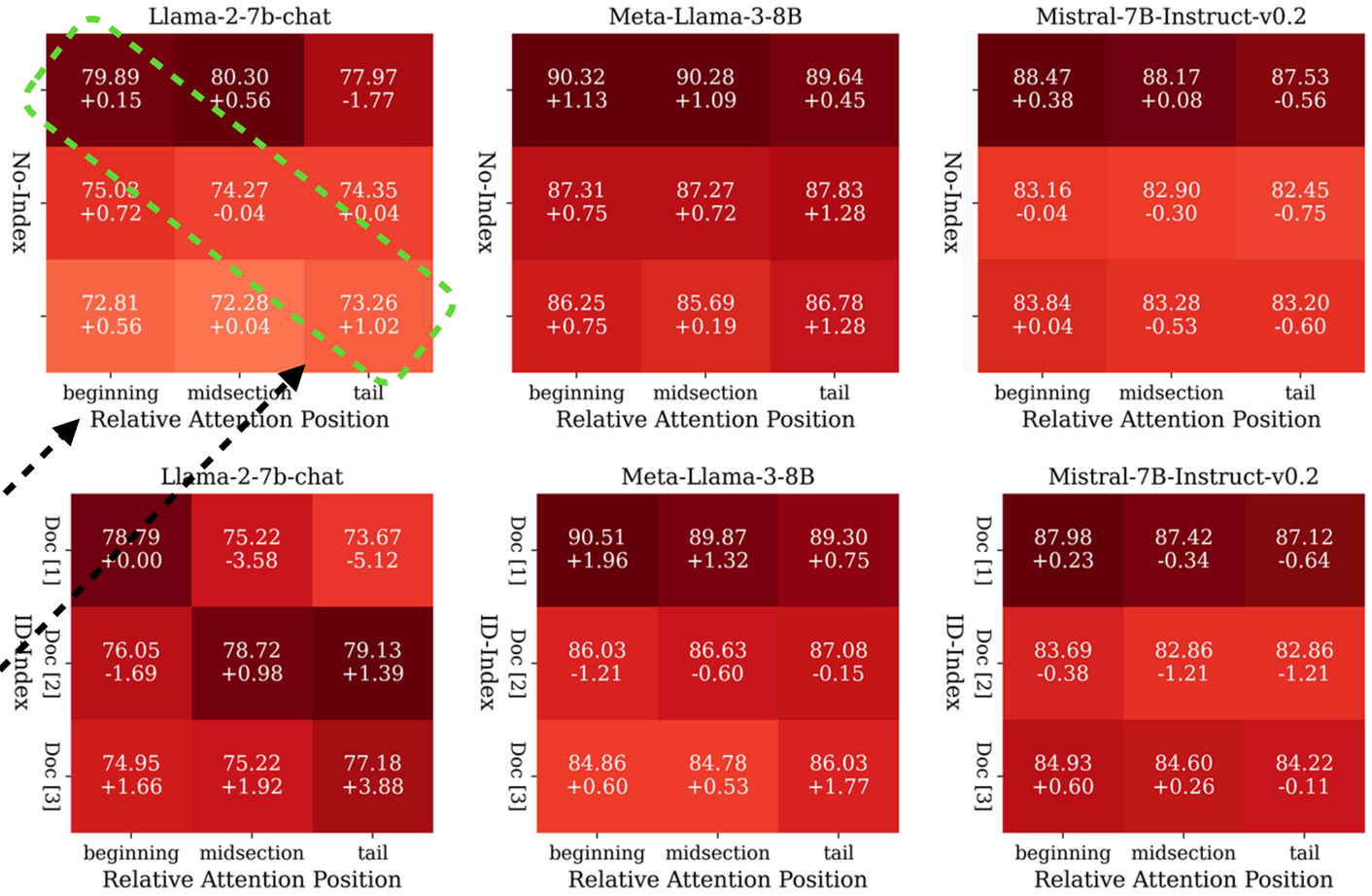
Can LLMs follow relative attention instructions?

Where the gold evidence is

Position at which we measure attention

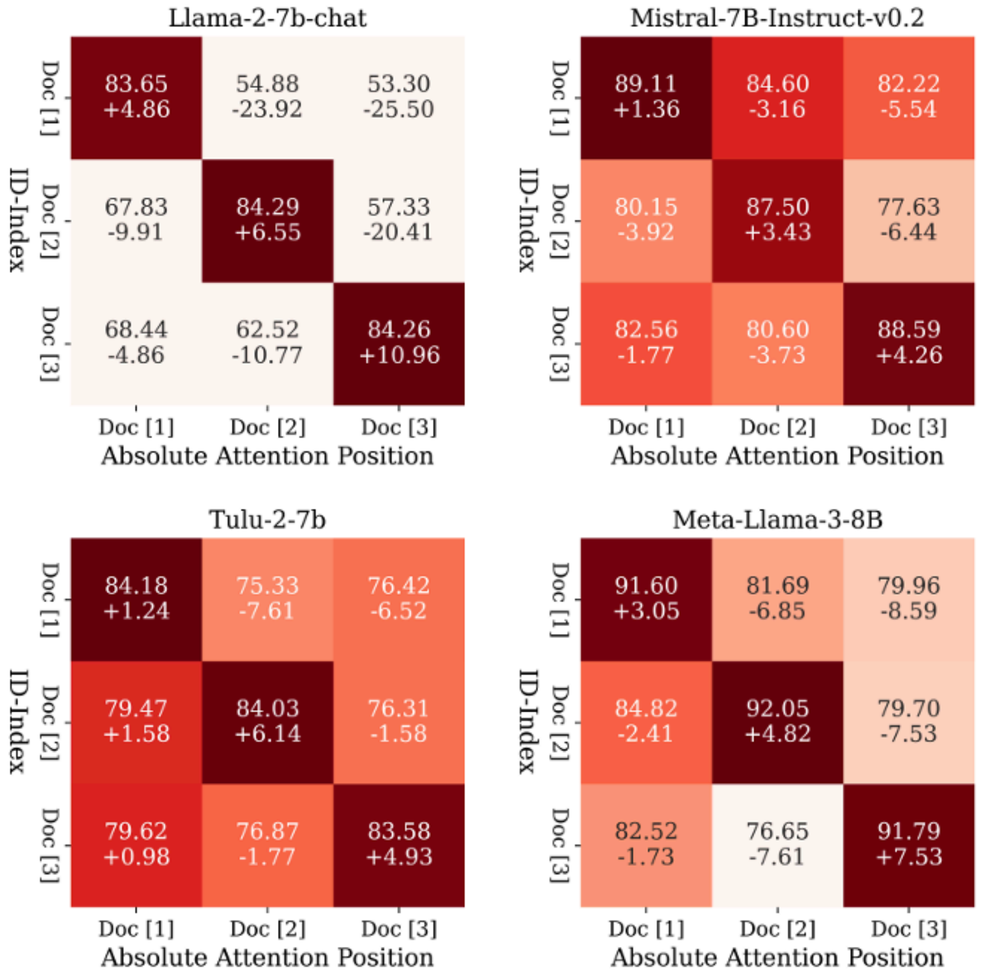
Improvement in diagonal: the LLM is following the instruction regarding where to put attention

Darker red in diagonal means the LLM follows instructions



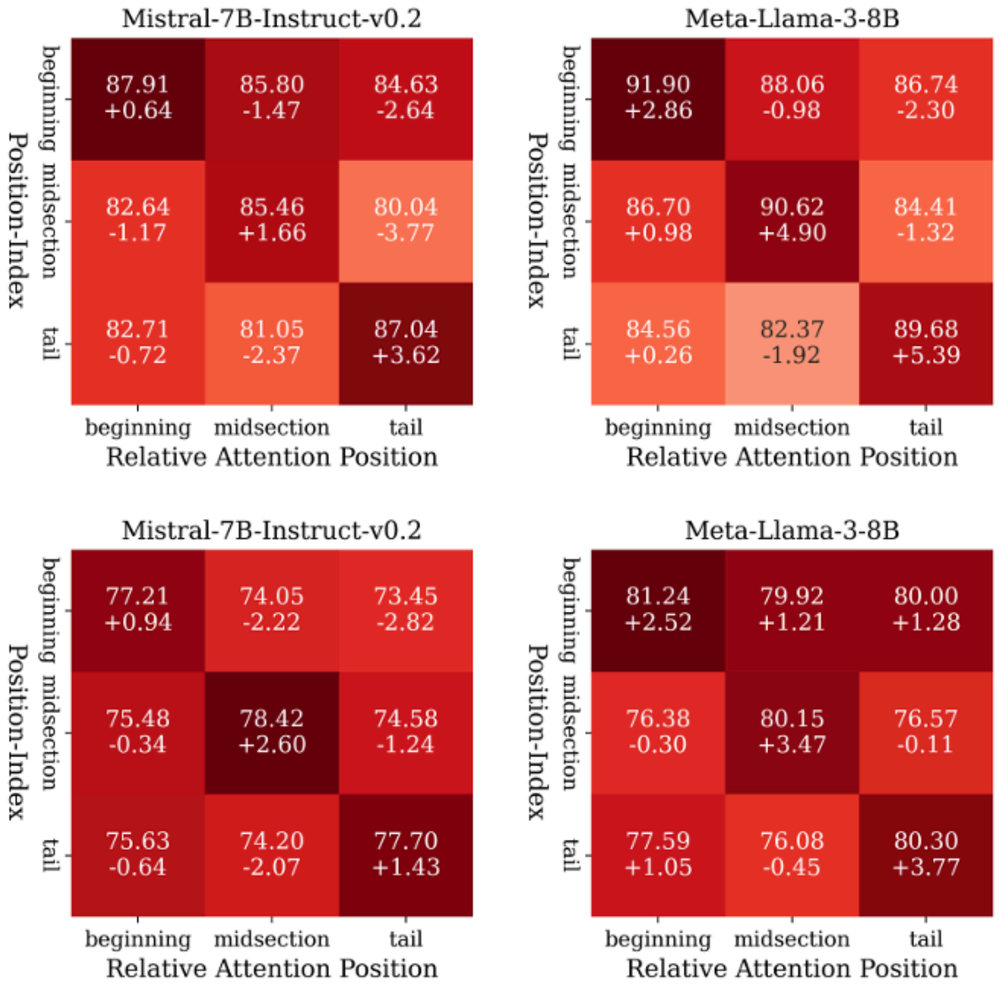
Attention Instruction: Pay Attention to the Middle

Can we instruct LLMs to attend to a document using absolute attention instruction?



Attention Instruction: Pay Attention to the Middle

Can LLMs follow absolute attention instruction with position-index and achieve regional control?



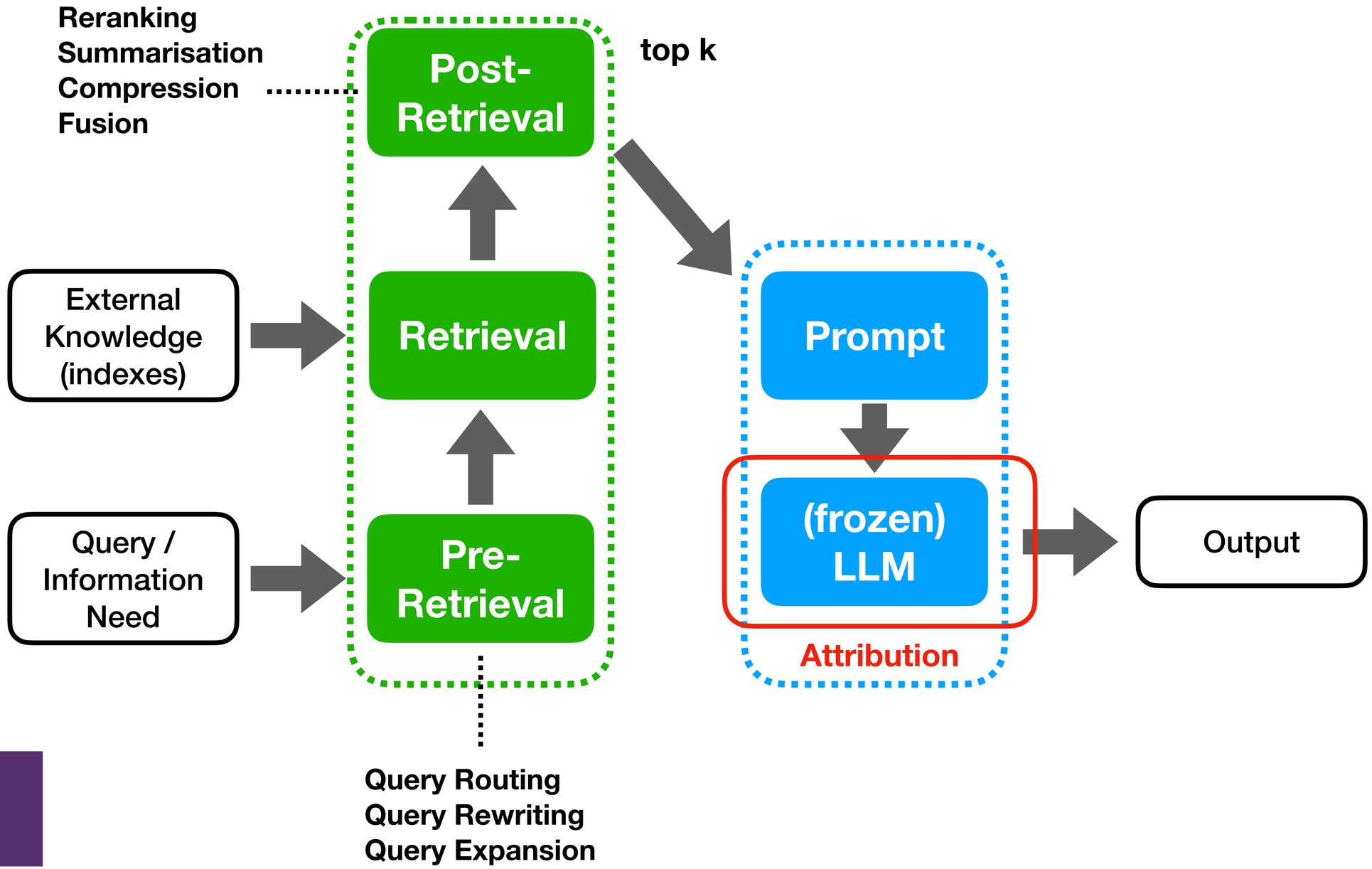
Attention Instruction: Pay Attention to the Middle

Key Findings:

- LLMs can be guided to pay more attention to a document or region through direct indexing (absolute attention instruction)
    - But LLMs do not have positional awareness of where each document is relative to the context (i.e. relative attention instruction)

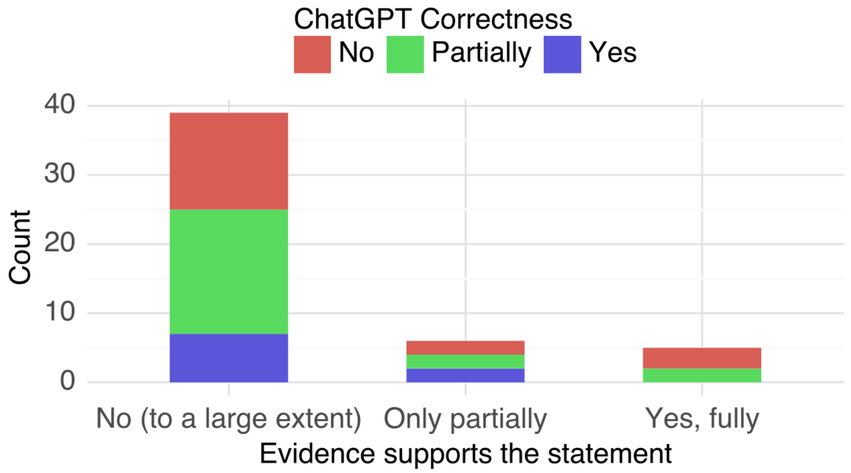
Zhang, M., Meng, Z. and Collier, N., 2024. Attention Instruction: Amplifying Attention in the Middle via Prompting. *arXiv preprint arXiv:2406.17095*.

In the remaining of this part…



Attribution

- Attribution: the ability to generate evidence (in the form of references or citations) that supports claims the LLM makes in an answer.
- LLMs fail to correctly attribute



- Two directions: (1) direct generation attribution, (2) retrieval-based attribution

Question: "What type of herbicides are effective against sowthistle?"

Ground truth answer: "The addition of Balance to either Group D or Group K herbicides can provide good control of sowthistle. The addition of Flame, Group D, Balance or Group K to broadleaf active herbicides (Group C and Valour) are also effective."

ChatGPT Output:

Herbicides that are effective against sowthistle include postemergent selective herbicides such as 2,4-D, dicamba, and glyphosate.

1. 2,4-D: A study conducted by R.L. Richardson and published in the journal "Weed Science" in 1983 (Vol. 31, Issue 4, pp. 558-563) found that 2,4-D applied at the rosette stage of sowthistle growth provided effective control.

2. Dicamba: A study by K.A. Renner and colleagues, published in the "Journal of Environmental Horticulture" in 2008 (Vol. 26, Issue 2, pp. 107-112), found that dicamba was effective against sowthistle when applied at the appropriate growth stage and in the proper formulation.

3. Glyphosate: According to a study by P.J. Tranel and J.D. Nalewaja, published in the journal "Weed Science" in 1998 (Vol. 46, Issue 4, pp. 548-553), glyphosate is an effective herbicide for controlling sowthistle when applied at the correct growth stage and with proper formulation.

Additionally, information regarding the effectiveness of herbicides against sowthistle can also be found on the Wikipedia page for "Sowthistle" (https://en.wikipedia.org/wiki/Sowthistle). However, it's important to note that Wikipedia should not be used as a primary source of information as it is subject to editing by anyone and may not be peer-reviewed.

The four references provided do not exist

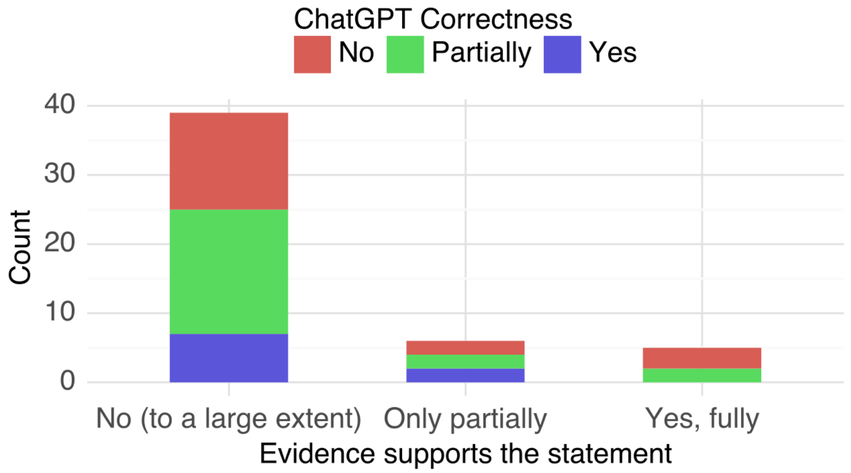
Statement about the reference

Invalid URL

Zuccon, G., Koopman, B. and Shaik, R., 2023. ChatGPT hallucinates when attributing answers. SIGIR-AP 2023

Attribution

- Attribution: the ability to generate evidence (in the form of references or citations) that supports claims the LLM makes in an answer.
- LLMs fail to correctly attribute



- Two directions: (1) direct generation attribution, (2) retrieval-based attribution

Question: "What type of herbicides are effective against sowthistle?"

Ground truth answer: "The addition of Balance to either Group D or Group K herbicides can provide good control of sowthistle. The addition of Flame, Group D, Balance or Group K to broadleaf active herbicides (Group C and Valour) are also effective."

ChatGPT Output:

Herbicides that are effective against sowthistle include postemergent selective herbicides such as 2,4-D, dicamba, and glyphosate.

1. 2,4-D: A study conducted by R.L. Richardson and published in the journal "Weed Science" in 1983 (Vol. 31, Issue 4, pp. 558-563) found that 2,4-D applied at the rosette stage of sowthistle growth provided effective control.

2. Dicamba: A study by K.A. Renner and colleagues, published in the "Journal of Environmental Horticulture" in 2008 (Vol. 26, Issue 2, pp. 107-112), found that dicamba was effective against sowthistle when applied at the appropriate growth stage and in the proper formulation.

3. Glyphosate: According to a study by P.J. Tranel and J.D. Nalewaja, published in the journal "Weed Science" in 1998 (Vol. 46, Issue 4, pp. 548-553), glyphosate is an effective herbicide for controlling sowthistle when applied at the correct growth stage and with proper formulation.

Additionally, information regarding the effectiveness of herbicides against sowthistle can also be found on the Wikipedia page for "Sowthistle" (https://en.wikipedia.org/wiki/Sowthistle). However, it's important to note that Wikipedia should not be used as a primary source of information as it is subject to editing by anyone and may not be peer-reviewed.

The four references provided do not exist

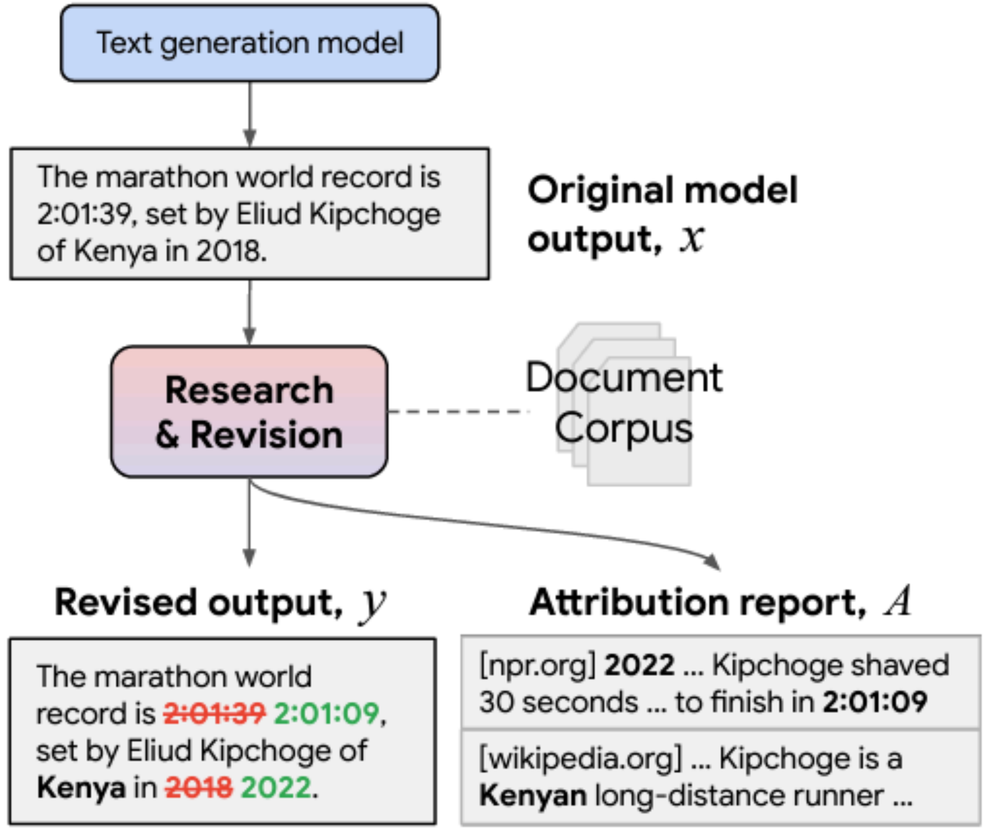
Statement about the reference

Invalid URL

Zuccon, G., Koopman, B. and Shaik, R., 2023. ChatGPT hallucinates when attributing answers. SIGIR-AP 2023

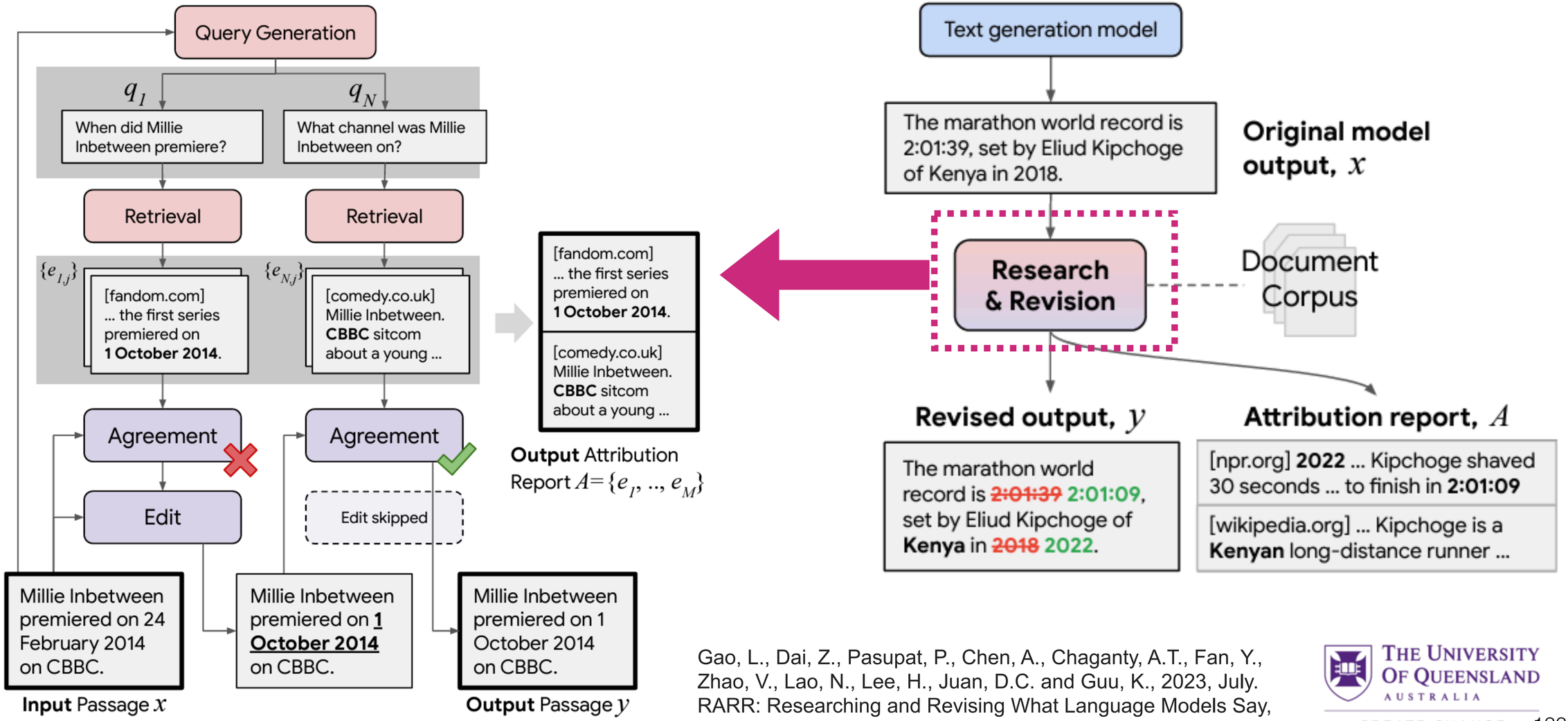
RARR: Retrofit Attribution using Research and Revision

- automatically finds attribution for LLM output
- post-edits the output to fix unsupported content while preserving original output as much as possible.



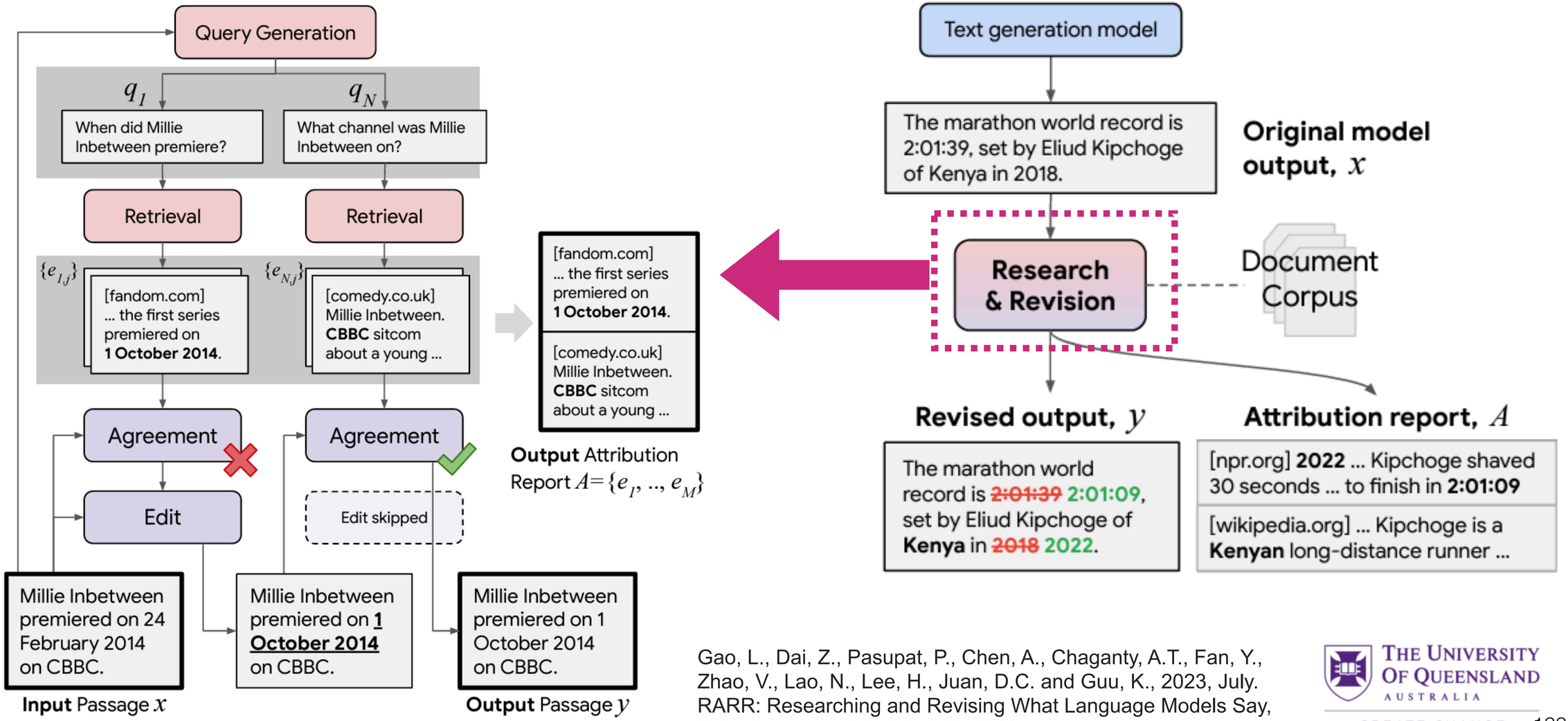
Gao, L., Dai, Z., Pasupat, P., Chen, A., Chaganty, A.T., Fan, Y., Zhao, V., Lao, N., Lee, H., Juan, D.C. and Guu, K., 2023, July. RARR: Researching and Revising What Language Models Say, Using Language Models. ACL 2023

RARR: Retrofit Attribution using Research and Revision



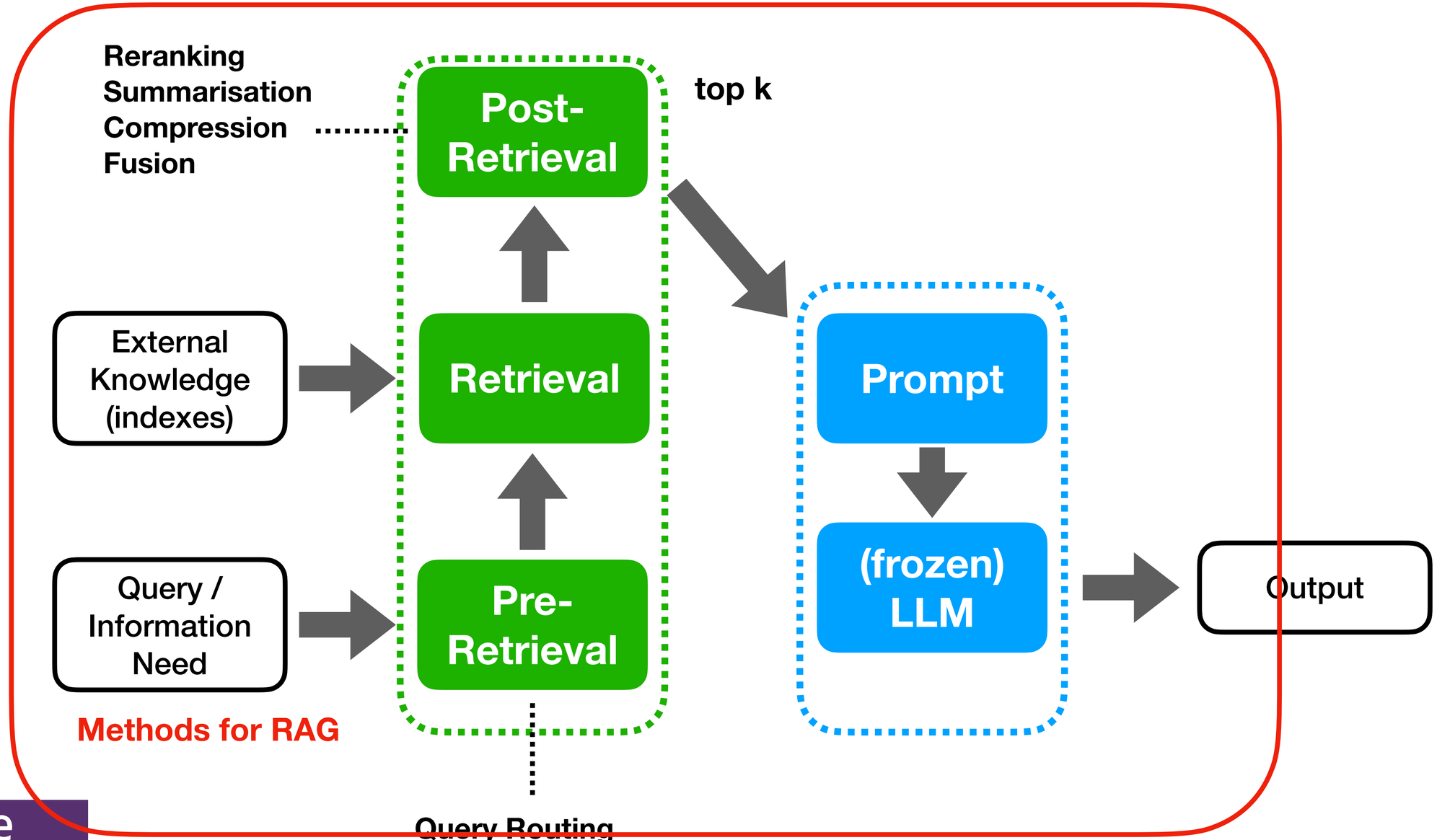
Gao, L., Dai, Z., Pasupat, P., Chen, A., Chaganty, A.T., Fan, Y., Zhao, V., Lao, N., Lee, H., Juan, D.C. and Guu, K., 2023, July. RARR: Researching and Revising What Language Models Say, Using Language Models. ACL 2023

RARR: Retrofit Attribution using Research and Revision



Gao, L., Dai, Z., Pasupat, P., Chen, A., Chaganty, A.T., Fan, Y., Zhao, V., Lao, N., Lee, H., Juan, D.C. and Guu, K., 2023, July. RARR: Researching and Revising What Language Models Say, Using Language Models. ACL 2023

In the remaining of this part…



Resources & Platforms for RAG

- Many **platforms/frameworks** for RAG
  - Industrial-oriented, e.g. Llama-Index, LangChain
    - Some provided only/also as a service, e.g. Nuclia AI
  - Research-oriented: **Ragnarök**, FlashRAG, **BERGEN**
- **Datasets**
  - Many datasets, not necessarily focusing on RAG
  - Few recent datasets specifically designed for RAG, e.g.
    - **TREC RAG 2024**: currently on-going [Topics released: August 4th; Submission deadline: August 11th]; Feb4Rag

BERGEN

- end-to-end library for reproducible research standardizing RAG experiments.
- focus on QA
- Implements different state-of-the-art retrievers, rerankers, and LLMs.

```
python bergen.py dataset=kilt_nq generator=gemma-7b
retriever=splade-v3 reranker=minilm6
```

Natural Questions, Trivia QA, HotpotQA, WoW, ELI5, WikiQA, TruthfulQA, PopQA, ASQA, SCIQ

BM25, RetroMAE v2, RepLlama, SPLADE, DeBERTa-v3, MiniLM, …

Llama-2-7B,13B,70B SOLAR-10B, Mixtral-8x7B, Gemma-2B, 7B, …

F1, (Exact) Match, Rouge-*, BEM, LLMEval, …

NAVER LABS Europe

July, 2024

BERGEN: A Benchmarking Library for Retrieval-Augmented Generation

David Rau* Hervé Déjean Nadezhda Chirkova Thibault Formal Shuai Wang* Vassilina Nikoulina Stéphane Clinchant
NAVER LABS Europe
* Work performed while at Naver Labs Europe.
https://github.com/naver/bergen

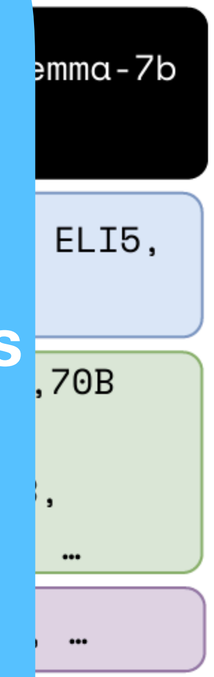
Rau D., Déjean H., Chirkova N., Formal T., Wang S., Nikoulina V., Clinchant, S.. 2024. BERGEN: A Benchmarking Library for Retrieval-Augmented Generation, arXiv:2407.01102v1

BERGEN

- end
  rese
- focu
- Imp
  retr

Also large scale analysis of RAG components (state-of-the-art retrievers, rerankers, LLMs -> 500+ experiments).
Key findings:

1. need to go **beyond commonly used surface-matching metrics** (e.g. exact match, F1, Rouge-L, etc.)
2. **retrieval** quality **matters** for RAG response generation
3. need to **improve** current knowledge-intensive **benchmarks** to use them in RAG:
   - datasets evaluating general knowledge might not be suitable for RAG, as LLMs have acquired most such knowledge from Web/Wikipedia
4. **LLMs** of any size can **benefit from retrieval**

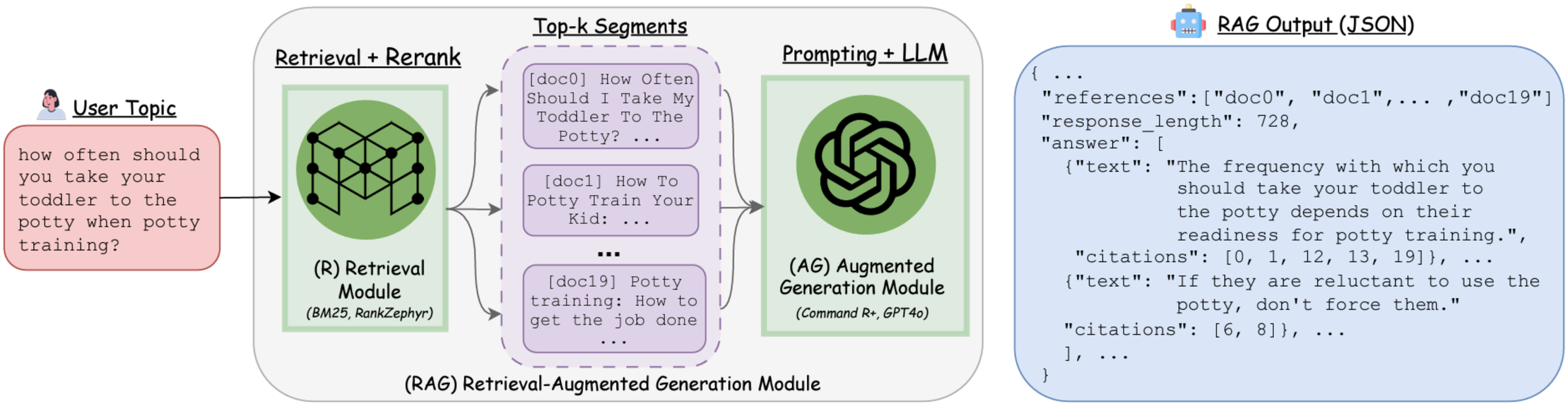


NAVER

BERGEN
Genera

David Rau* [illegible]
Stéphane Clin
NAVER LABS Eur
* Work performed w
https://github.com/naver/bergen

TREC RAG 2024 & Ragnarök



- TREC RAG 2024: dataset for RAG evaluation. Three tasks: Retrieve (R), Augmented Generation (AG), Retrieval Augmented Generation (RAG)

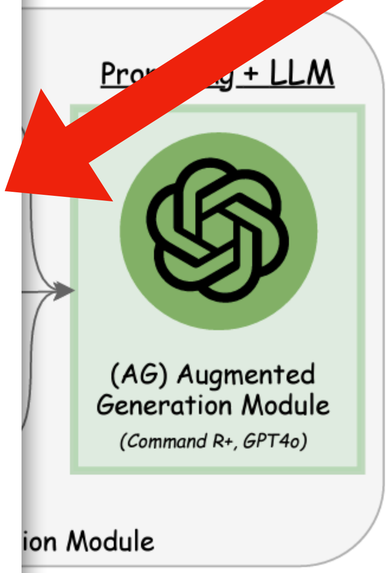
- Ragnarök: open-source, reproducible, reusable framework implementing RAG pipeline, with 2 sequential modules: (1) R, (2) AG

Pradeep, R., Thakur, N., Sharifymoghaddam, S., Zhang, E., Nguyen, R., Campos, D., Craswell, N. and Lin, J., 2024. Ragnarok: A Reusable RAG Framework and Baselines for TREC 2024 Retrieval-Augmented Generation Track. *arXiv preprint arXiv:2406.16828*.

024 & Ragnarök

Ragnarök

In Norse mythology, Ragnarök is a foretold series of impending events, including a great battle in which numerous great Norse mythological figures will perish; it will entail a catastrophic series of natural disasters, including the burning of the world, and culminate in the submersion of the world underwater. Wikipedia



RAG Output (JSON)

```
{ ...
 "references":["doc0", "doc1",... ,"doc19"]
 "response_length": 728,
 "answer": [
   {"text": "The frequency with which you
             should take your toddler to
             the potty depends on their
             readiness for potty training.",
    "citations": [0, 1, 12, 13, 19]}, ...
   {"text": "If they are reluctant to use the
             potty, don't force them."
   "citations": [6, 8]}, ...
   ], ...
 }
```

valuation. Three tasks: Retrieve (R),
al Augmented Generation (RAG)

mework implementing RAG

Really?

Many More Datasets

Task	Sub Task	Dataset
QA	Single-hop	Natural Qustion(NQ) [111]
		TriviaQA(TQA) [113]
		SQuAD [114]
		Web Questions(WebQ) [115]
		PopQA [116]
		MS MARCO [117]
	Multi-hop	HotpotQA [118]
		2WikiMultiHopQA [119]
		MuSiQue [120]
	Long-form QA	ELI5 [121]
		NarrativeQA(NQA) [122]
		ASQA [124]
		QMSum(QM) [125]
	Domain QA	Qasper [126]
		COVID-QA [127]
		CMB [128],MMCU_Medical [129]
	Multi-Choice QA	QuALITY [130]
		ARC [131]
		CommonsenseQA [132]
	Graph QA	GraphQA [84]

Task	Sub Task	Dataset
Dialog	Dialog Generation	Wizard of Wikipedia (WoW) [133]
	Personal Dialog	KBP [134]
		DuleMon [136]
	Task-oriented Dialog	CamRest [137]
	Recommendation	Amazon(Toys,Sport,Beauty) [138]
IE	Event Argument Extraction	WikiEvent [139]
		RAMS [140]
	Relation Extraction	T-REx [141],ZsRE [142]
Reasoning	Commonsense Reasoning	HellaSwag [143]
	CoT Reasoning	CoT Reasoning [144]
	Complex Reasoning	CSQA [145]
Others	Language Understanding	MMLU [146]
	Language Modeling	WikiText-103 [147]
		StrategyQA [148]
	Fact Checking/Verification	FEVER [149]
		PubHealth [150]
	Text Generation	Biography [151]
	Text Summarization	WikiASP [152]
		XSum [153]
	Text Classification	VioLens [154]
		TREC [155]
	Sentiment	SST-2 [156]
	Code Search	CodeSearchNet [157]
	Robustness Evaluation	NoMIRACL [56]
	Math	GSM8K [158]
	Machine Translation	JRC-Acquis [159]

From: Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J. and Wang, H., 2023. Retrieval-augmented generation for large language models: A survey. arXiv preprint arXiv:2312.10997.

Many Open Directions

RAG vs. Long Context

RAG Evaluation

Scaling Laws in RAG

RAG Robustness

End-to-End Optimisation

Hybrid Approaches

Production-Ready RAGS

And many more: temporality, trade-offs

Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J. and Wang, H., 2023. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*.

Part 3: Model-based IR

Model-based IR

Rethinking Search: Making Domain Experts out of Dilettantes*

Donald Metzler
Google Research
metzler@google.com

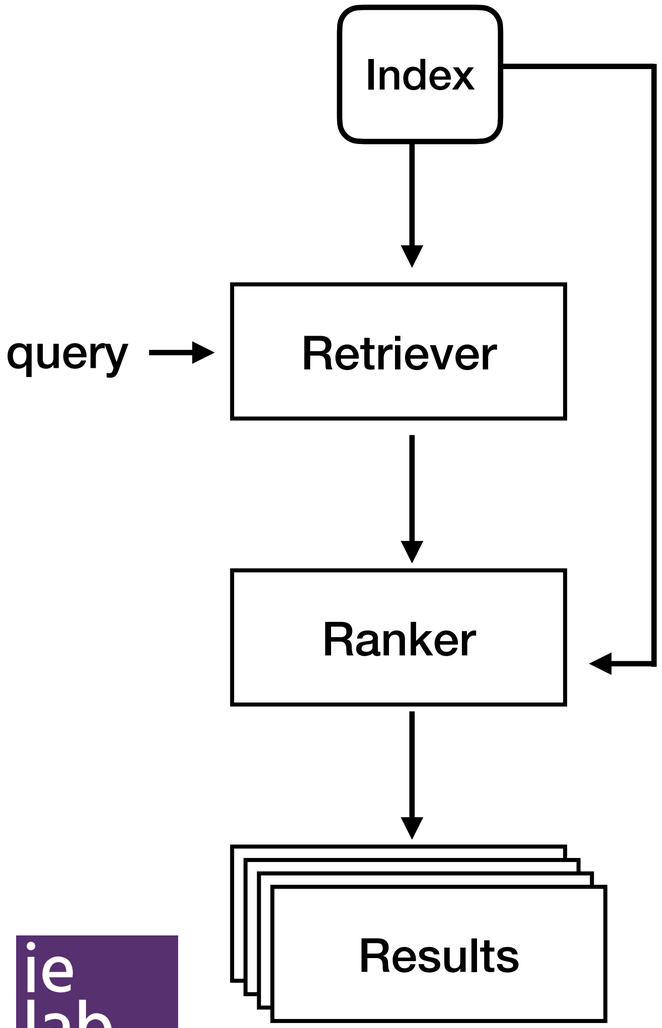
Yi Tay
Google Research
yitay@google.com

Dara Bahri
Google Research
dbahri@google.com

Marc Najork
Google Research
najork@google.com

ACM SIGIR Forum, Volume 55, Issue 1 (2021)

Model-based IR



Rethinking Search: Making Domain Experts out of Dilettantes*

Donald Metzler
Google Research
metzler@google.com

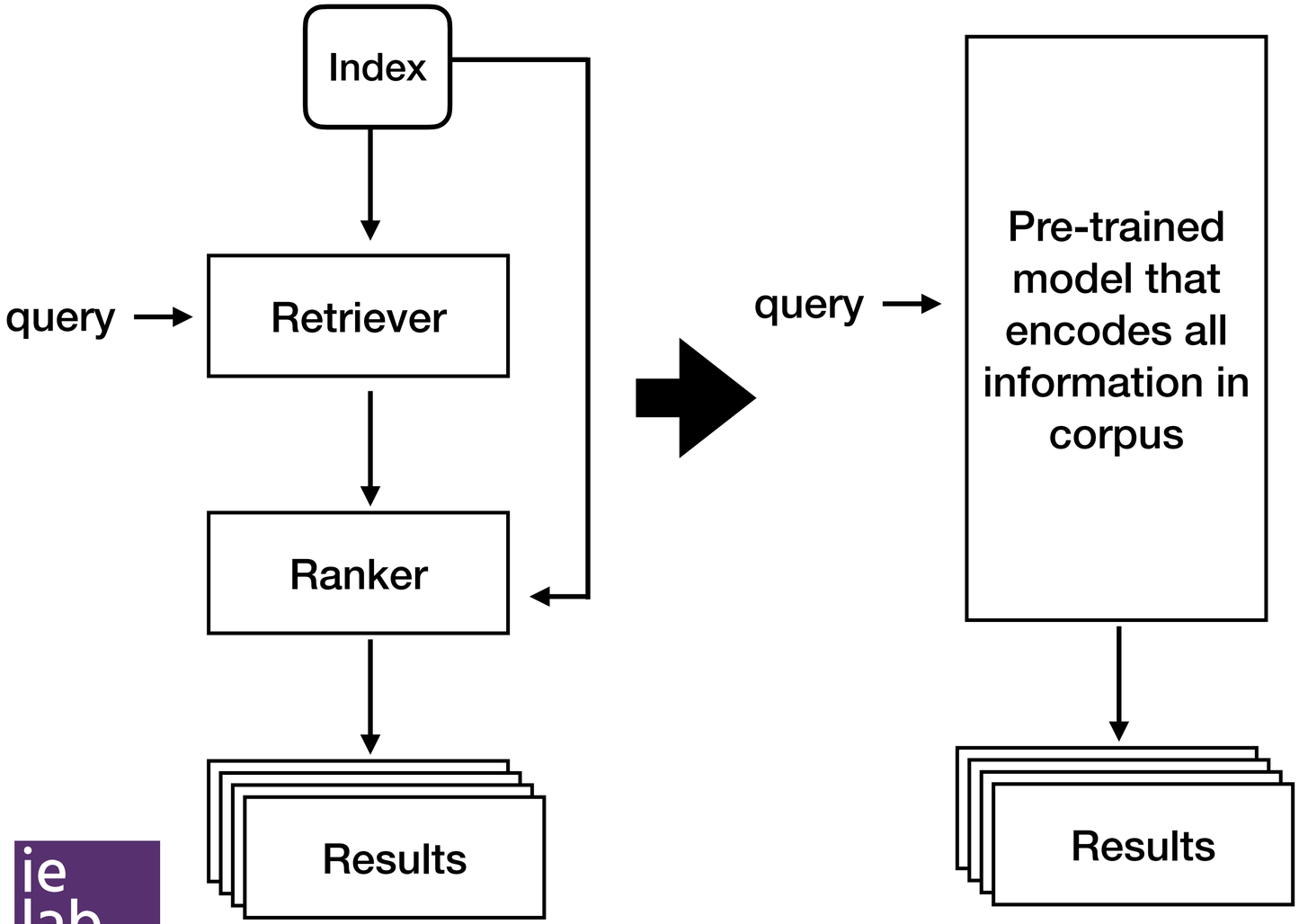
Yi Tay
Google Research
yitay@google.com

Dara Bahri
Google Research
dbahri@google.com

Marc Najork
Google Research
najork@google.com

ACM SIGIR Forum, Volume 55, Issue 1 (2021)

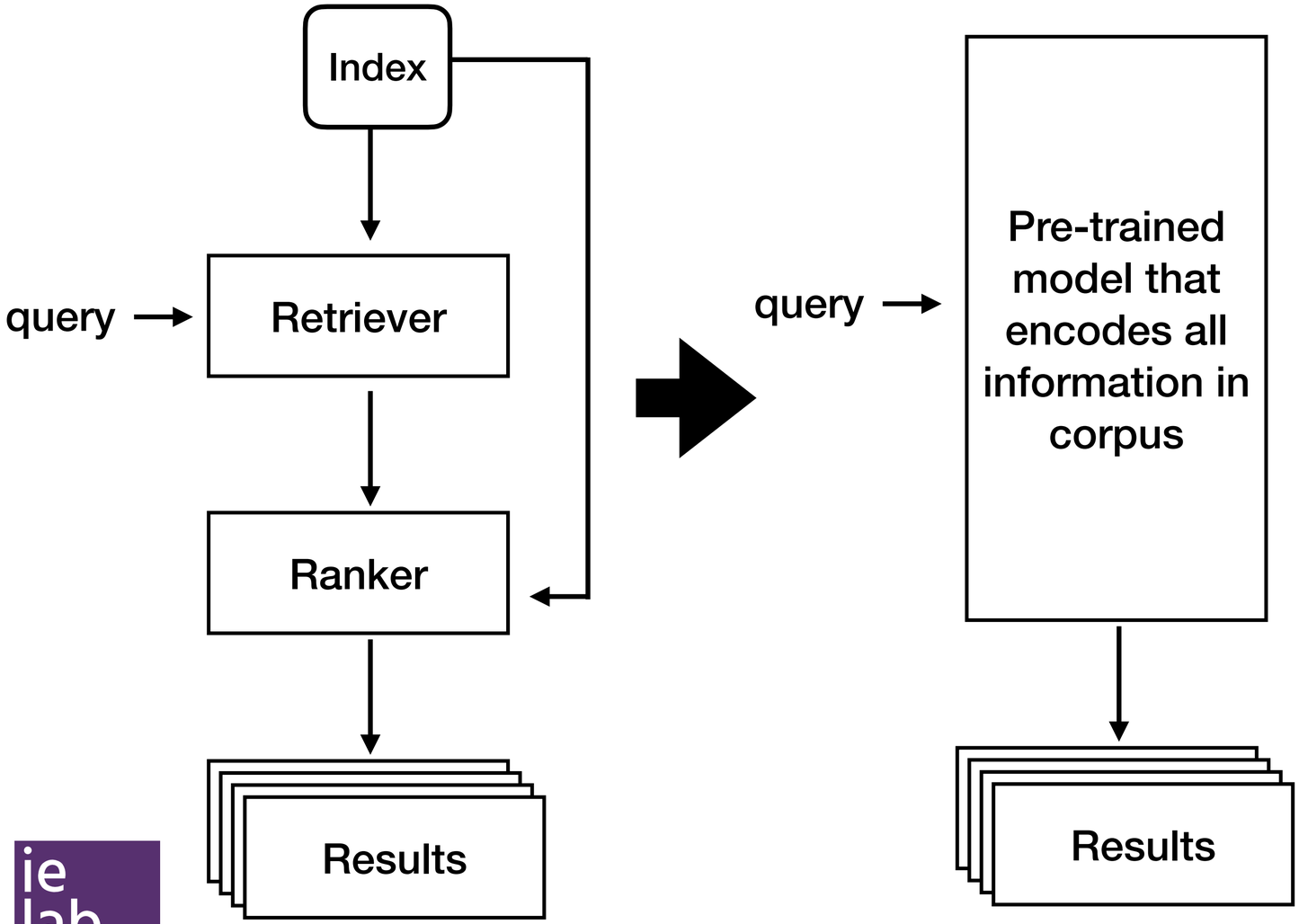
Model-based IR



Key Intuition:

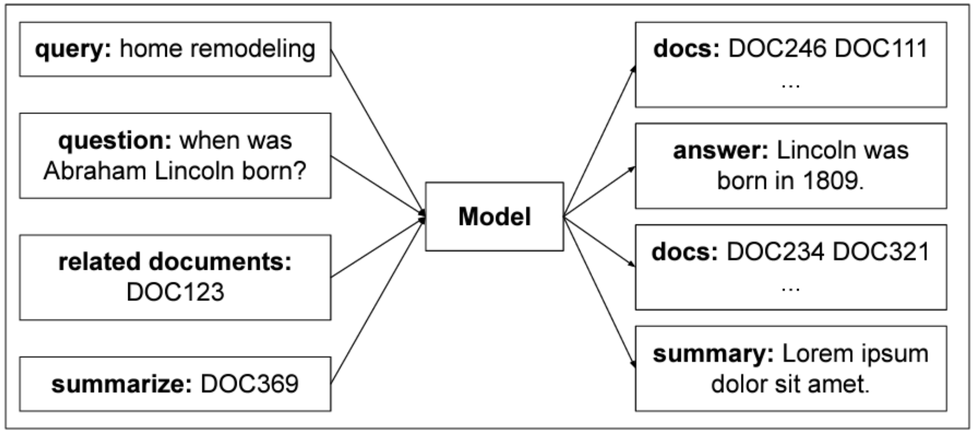
Replace pipelined architecture with a single consolidated model (Unified Retrieve-and-Rank)

Model-based IR



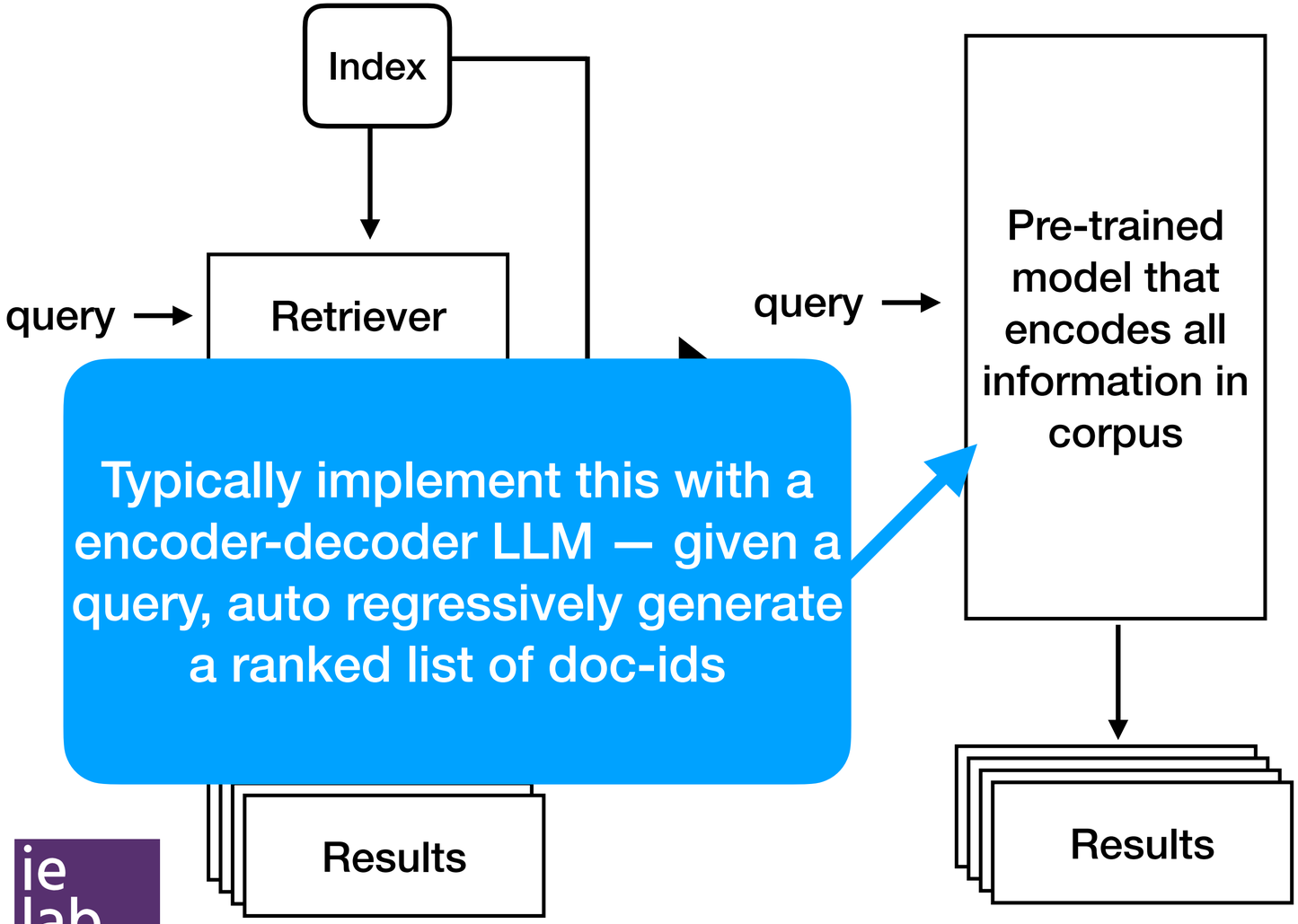
Key Intuition:

Replace pipelined architecture with a single consolidated model (Unified Retrieve-and-Rank)



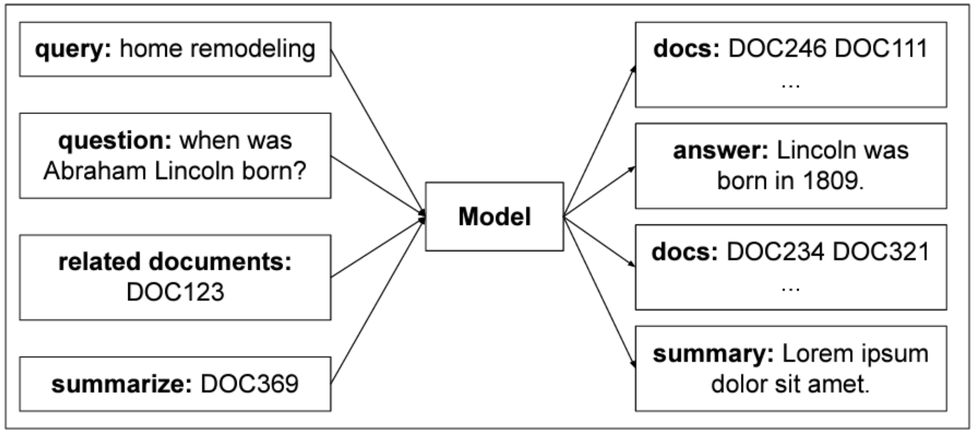
ie lab

Model-based IR



Key Intuition:

Replace pipelined architecture with a single consolidated model (Unified Retrieve-and-Rank)



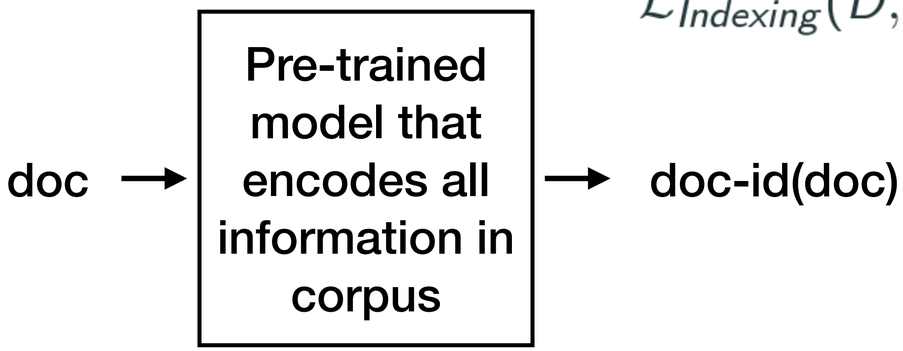
ie lab

“Indexing” in Distributed Search Index (DSI)

the model needs to memorise the content/information of each document

learn to associate the content of each document with its corresponding doc-id

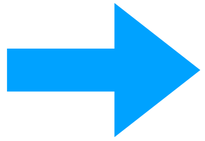
$$\mathcal{L}_{Indexing}(D, I_D; \theta) = -\sum_{d \in D} \log P(id \mid d; \theta),$$



Tay, Y., Tran, V., Dehghani, M., Ni, J., Bahri, D., Mehta, H., Qin, Z., Hui, K., Zhao, Z., Gupta, J., Schuster, T., Cohen, W., Metzler, D.. 2022. Transformer memory as a differentiable search index. NeurIPS 2022

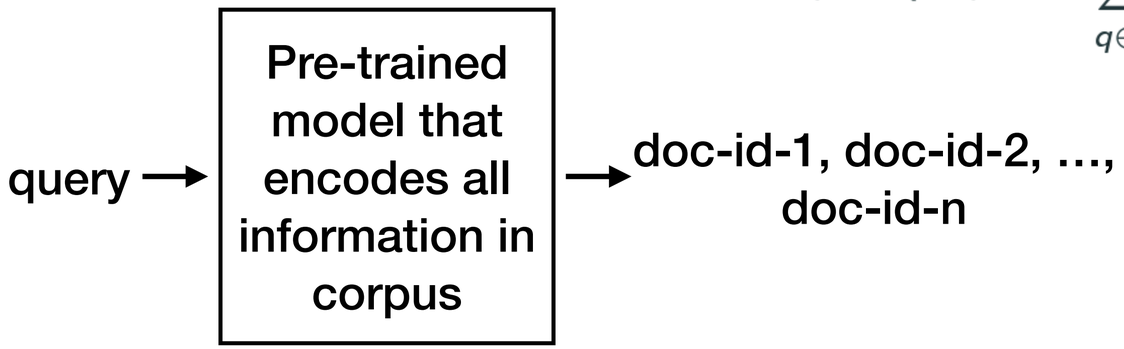
“Retrieval” in Distributed Search Index (DSI)

the model needs to answer a query



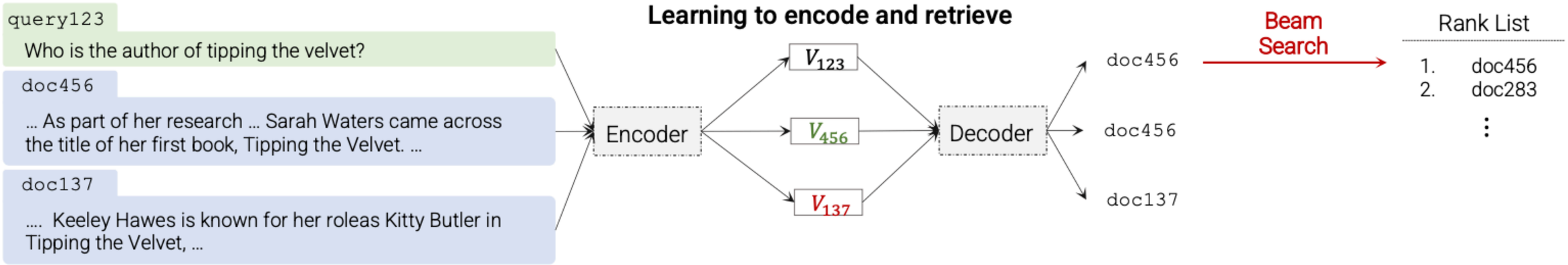
learn to generate a ranked list of candidate docids

$$\mathcal{L}_{Retrieval}(Q, I_Q; \theta) = -\sum_{q \in Q} \sum_{id^q \in I_Q} \log P(id^q \mid q; \theta),$$



Tay, Y., Tran, V., Dehghani, M., Ni, J., Bahri, D., Mehta, H., Qin, Z., Hui, K., Zhao, Z., Gupta, J., Schuster, T., Cohen, W., Metzler, D.. 2022. Transformer memory as a differentiable search index. NeurIPS 2022

Jointly Learn to Index and Retrieve

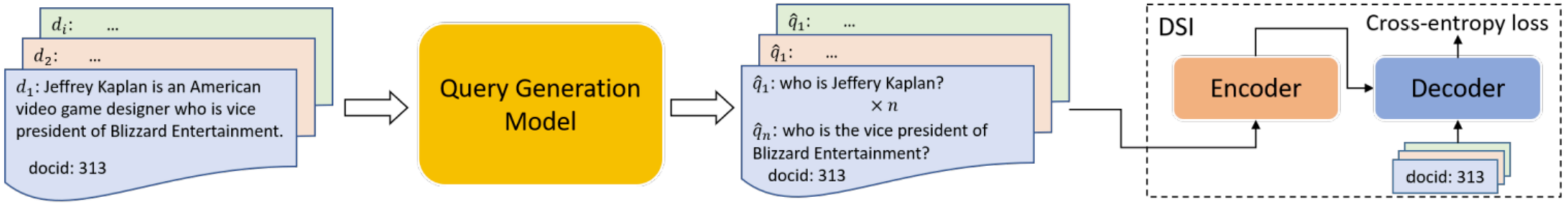


Optimise a single model directly in an end-to-end fashion towards global objective:

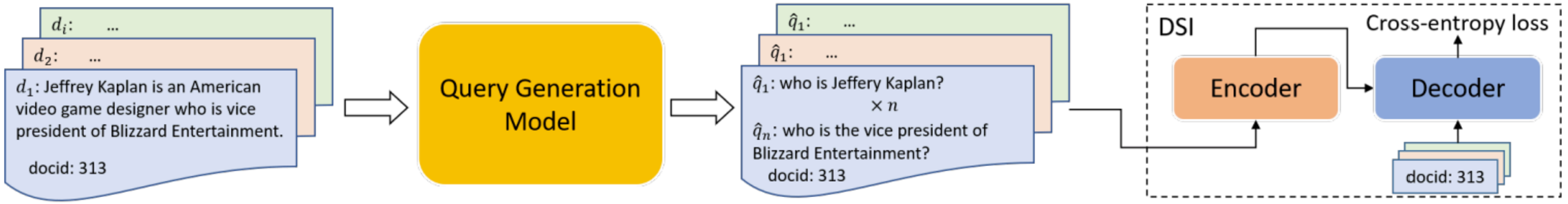
$$\mathcal{L}_{Global}(Q, D, I_D, I_Q; \theta) = \mathcal{L}_{Indexing}(D, I_D; \theta) + \mathcal{L}_{Retrieval}(Q, I_Q; \theta)$$

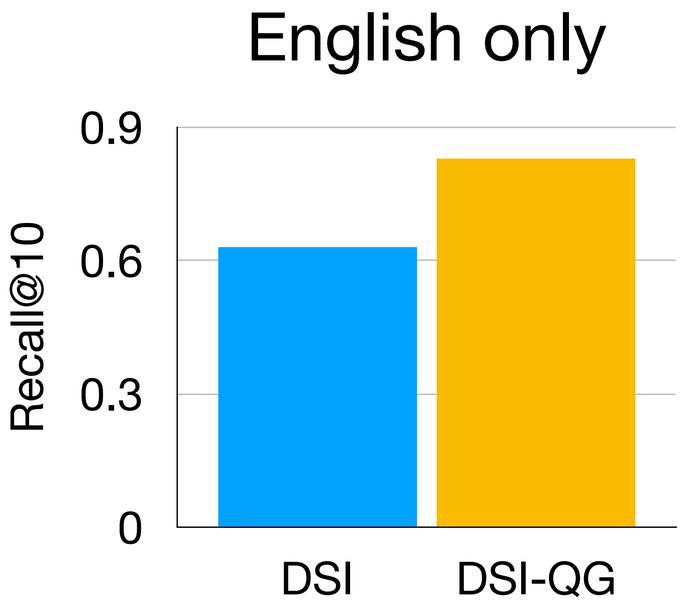
Tay, Y., Tran, V., Dehghani, M., Ni, J., Bahri, D., Mehta, H., Qin, Z., Hui, K., Zhao, Z., Gupta, J., Schuster, T., Cohen, W., Metzler, D.. 2022. Transformer memory as a differentiable search index. NeurIPS 2022

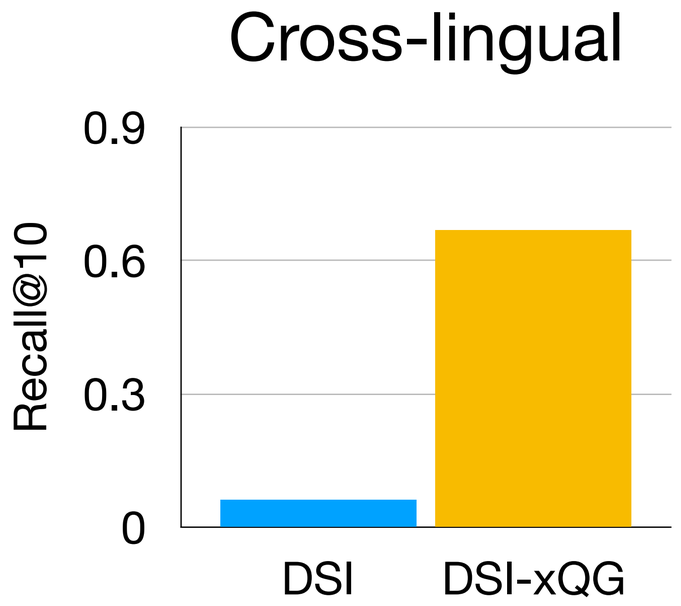
Differentiable Search Index with Query Generation



Differentiable Search Index with Query Generation

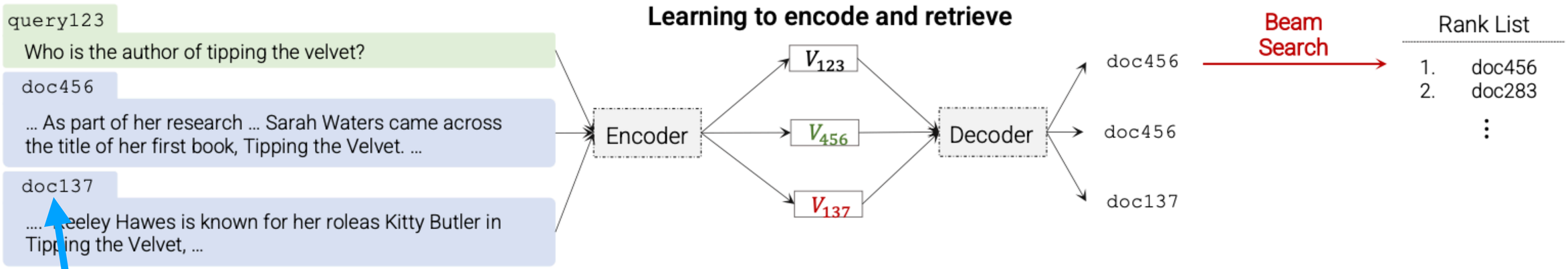






Zhuang, S., Ren, H., Shou, L., Pei, J., Gong, M., Zuccon, G. and Jiang, D., 2022. Bridging the gap between indexing and retrieval for differentiable search index with query generation. *arXiv preprint arXiv:2206.10128*.

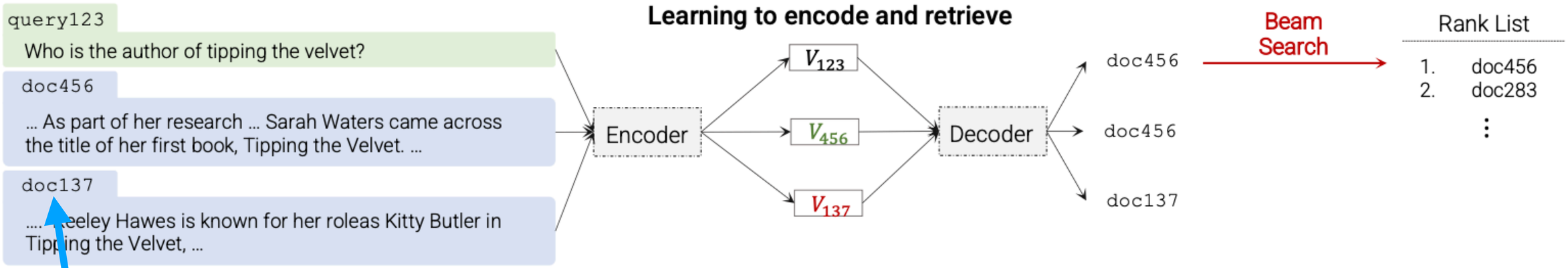
Questions



How should doc-ids look like?

Questions

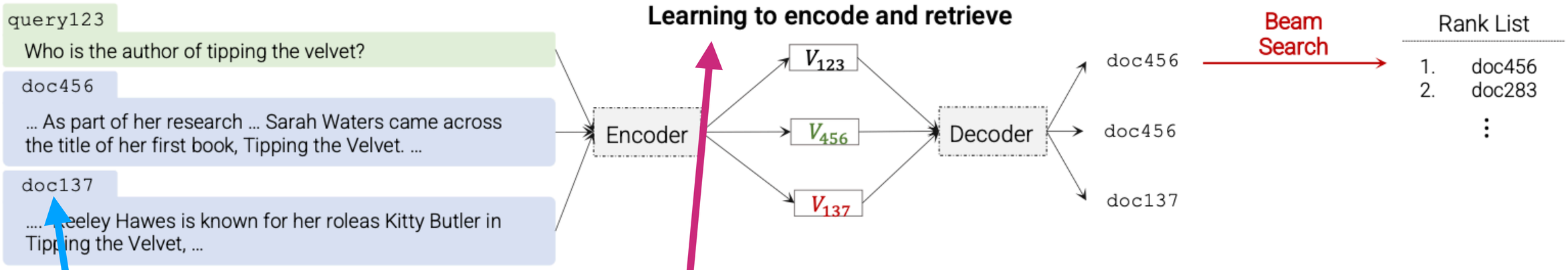
How to handle heterogeneous tasks (data distributions, objective functions)?



How should doc-ids look like?

Questions

How to handle heterogeneous tasks (data distributions, objective functions)?



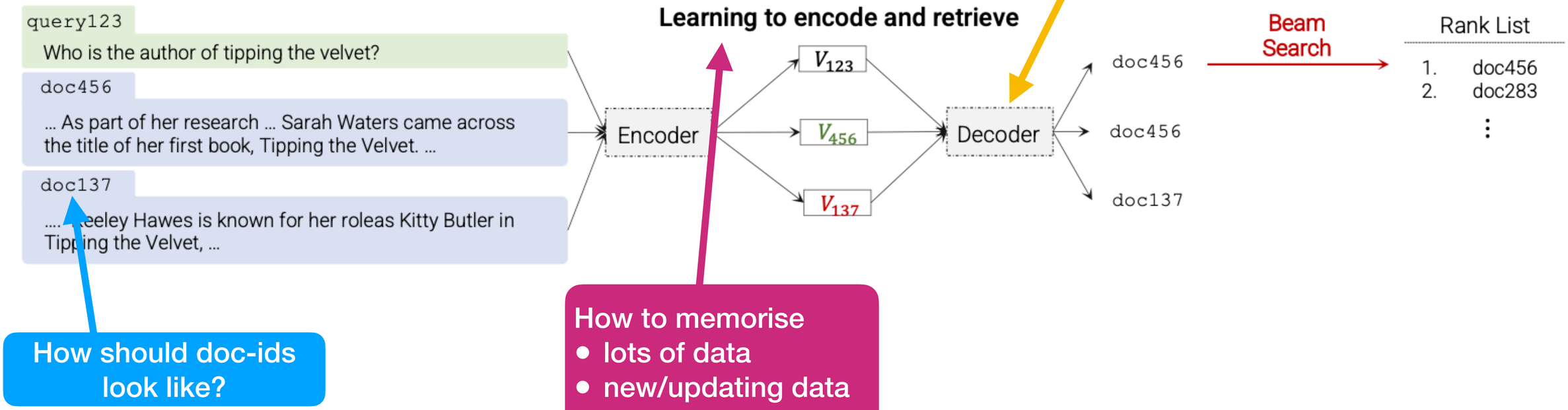
How should doc-ids look like?

How to memorise
- lots of data
- new/updating data
- efficiently

Questions

How to handle heterogeneous tasks (data distributions, objective functions)?

How to generate valid doc-ids?



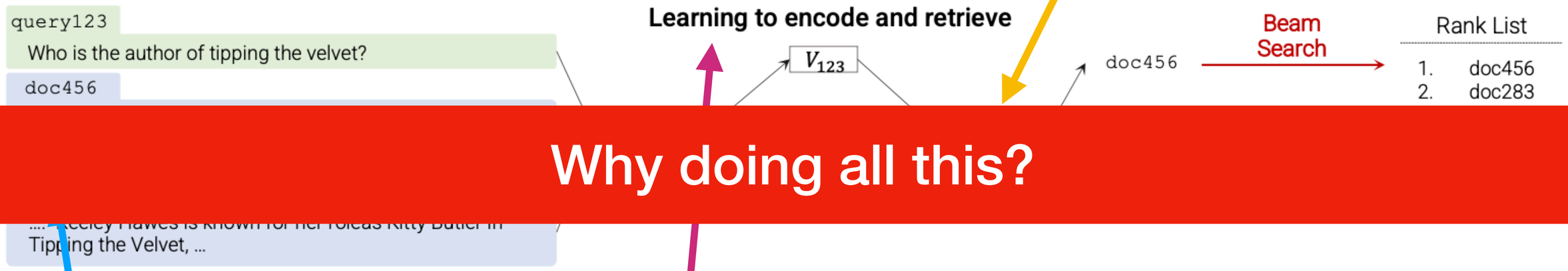
How should doc-ids look like?

How to memorise
- lots of data
- new/updating data
- efficiently

Questions

How to handle heterogeneous tasks (data distributions, objective functions)?

How to generate valid doc-ids?



Why doing all this?

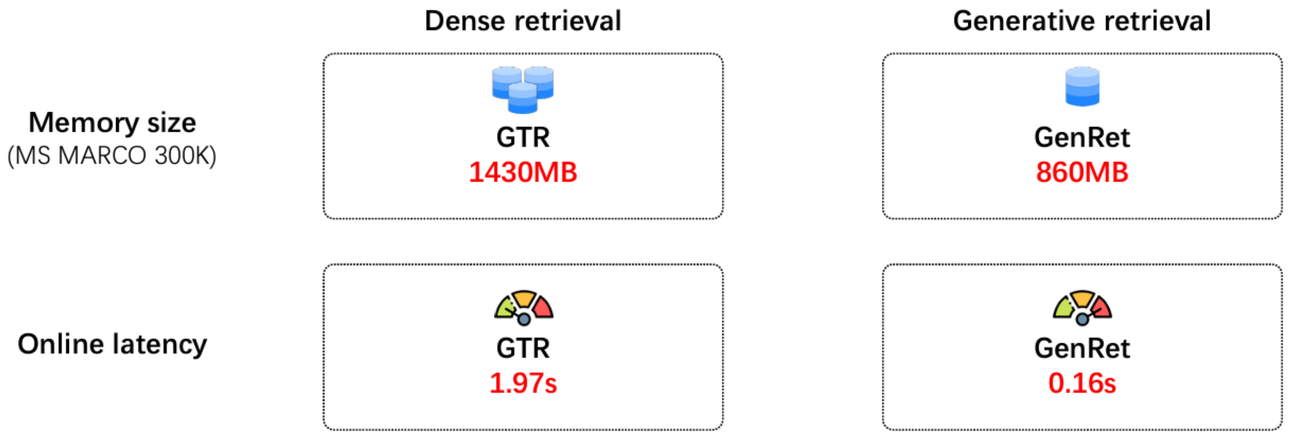
How should doc-ids look like?

How to memorise
- lots of data
- new/updating data
- efficiently

Why doing DSI

1) **Efficiency** at **inference**

- **Memory**:
  - Inverted index: corpus grows -> index grows
  - DSI: corpus grows -> model size stays the same (+ small overhead for doc-id storage)
- **Latency**: light generative process over the vocabulary of identifiers



2) Dramatic **simplification** of whole indexing/retrieval/generation process

- Less engineering/maintenance
- Imaging of instead of just generating doc-ids, you could generate answers, with doc-ids (attribution) ~> RAG

Bonus!

Extreme Model-based IR

Reflections on: “Can Long-Context Language Models Subsume Retrieval, RAG, SQL, and More?”

Can Long-Context Language Models Subsume Retrieval, RAG, SQL, and More?

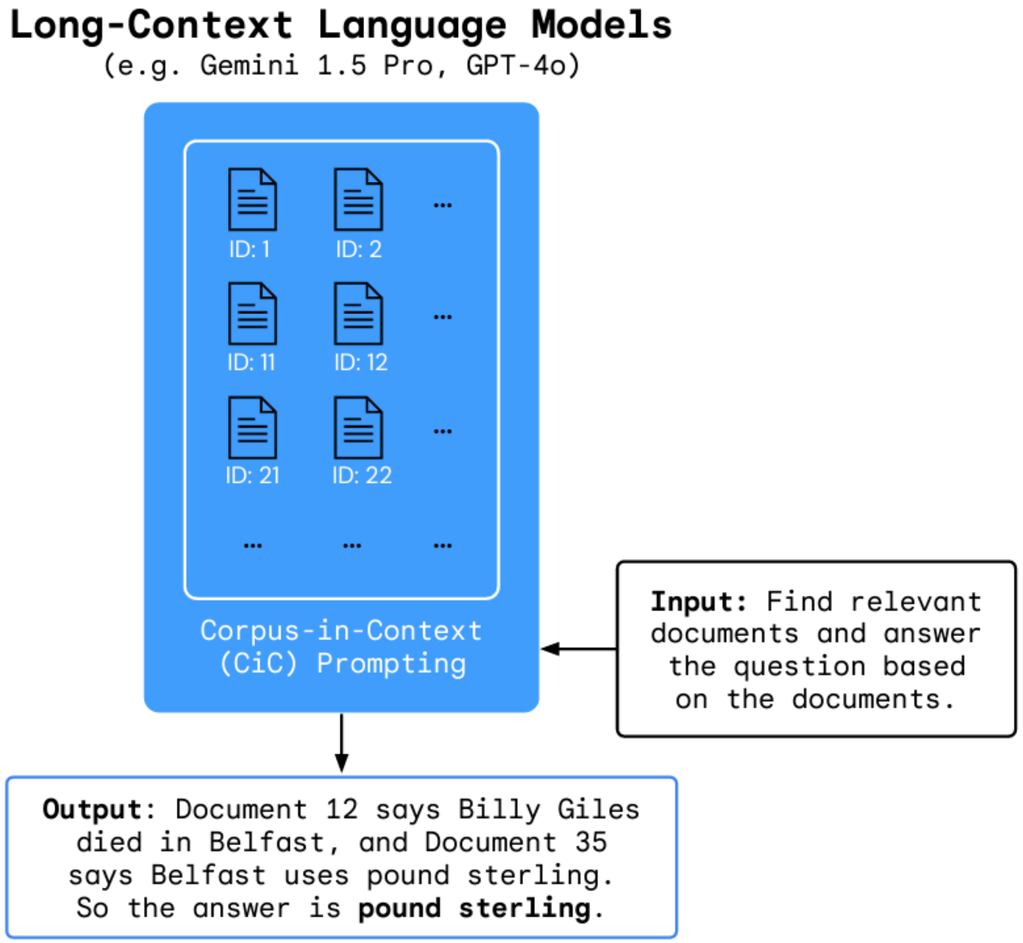
Jinhyuk Lee* **Anthony Chen*** **Zhuyun Dai***
Dheeru Dua **Devendra Singh Sachan** **Michael Boratko** **Yi Luan**
Sébastien M. R. Arnold **Vincent Perot** **Siddharth Dalmia** **Hexiang Hu**
Xudong Lin **Panupong Pasupat** **Aida Amini** **Jeremy R. Cole**
Sebastian Riedel **Iftekhar Naim** **Ming-Wei Chang** **Kelvin Guu**

Google DeepMind

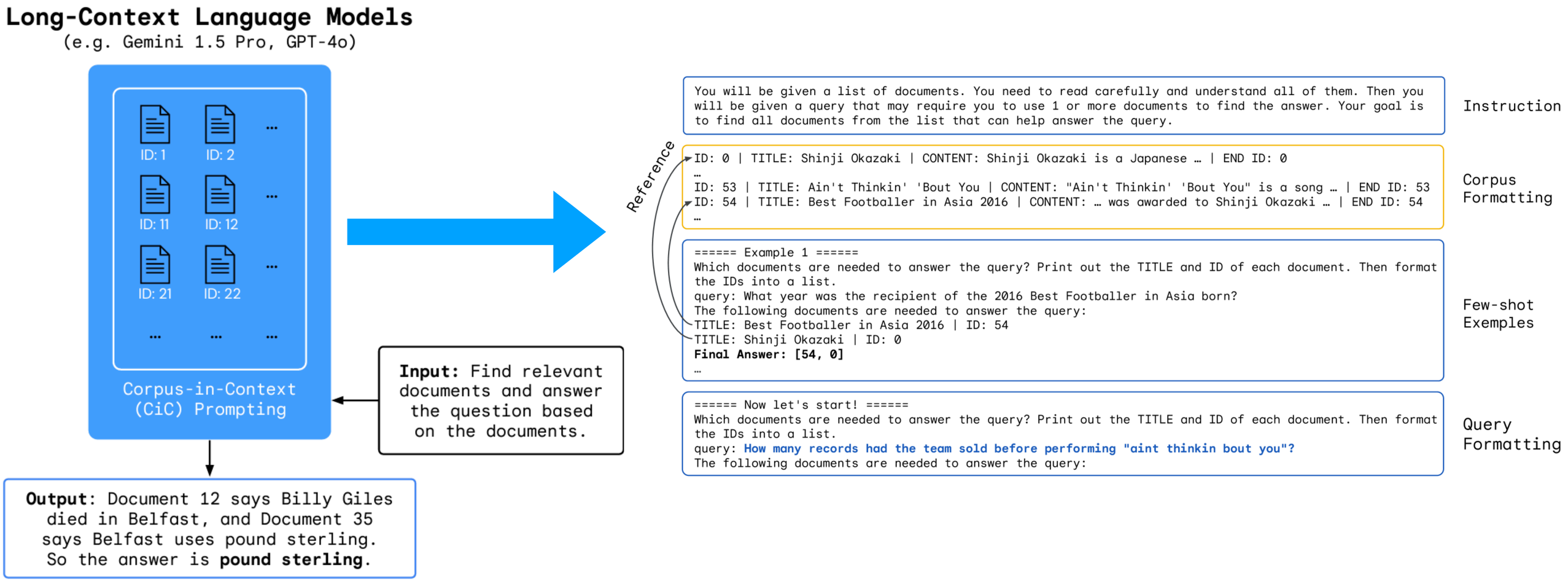
Abstract

Long-context language models (LCLMs) have the potential to revolutionize our approach to tasks traditionally reliant on external tools like retrieval systems or databases. Leveraging LCLMs’ ability to natively ingest and process entire corpora of information offers numerous advantages. It enhances user-friendliness by eliminating the need for specialized knowledge of tools, provides robust end-to-end modeling that minimizes cascading errors in complex pipelines, and allows for the application of sophisticated prompting techniques across the entire system. To assess this paradigm shift, we introduce LOFT, a benchmark of real-world tasks requiring context up to millions of tokens designed to evaluate LCLMs’ performance on in-context retrieval and reasoning. Our findings reveal LCLMs’ surprising ability to rival state-of-the-art retrieval and RAG systems, despite never having been explicitly trained for these tasks. However, LCLMs still face challenges in areas like compositional reasoning that are required in SQL-like tasks. Notably, prompting strategies significantly influence performance, emphasizing the need for continued research as context lengths grow. Overall, LOFT provides a rigorous testing ground for LCLMs, showcasing their potential to supplant existing paradigms and tackle novel tasks as model capabilities scale.[1]

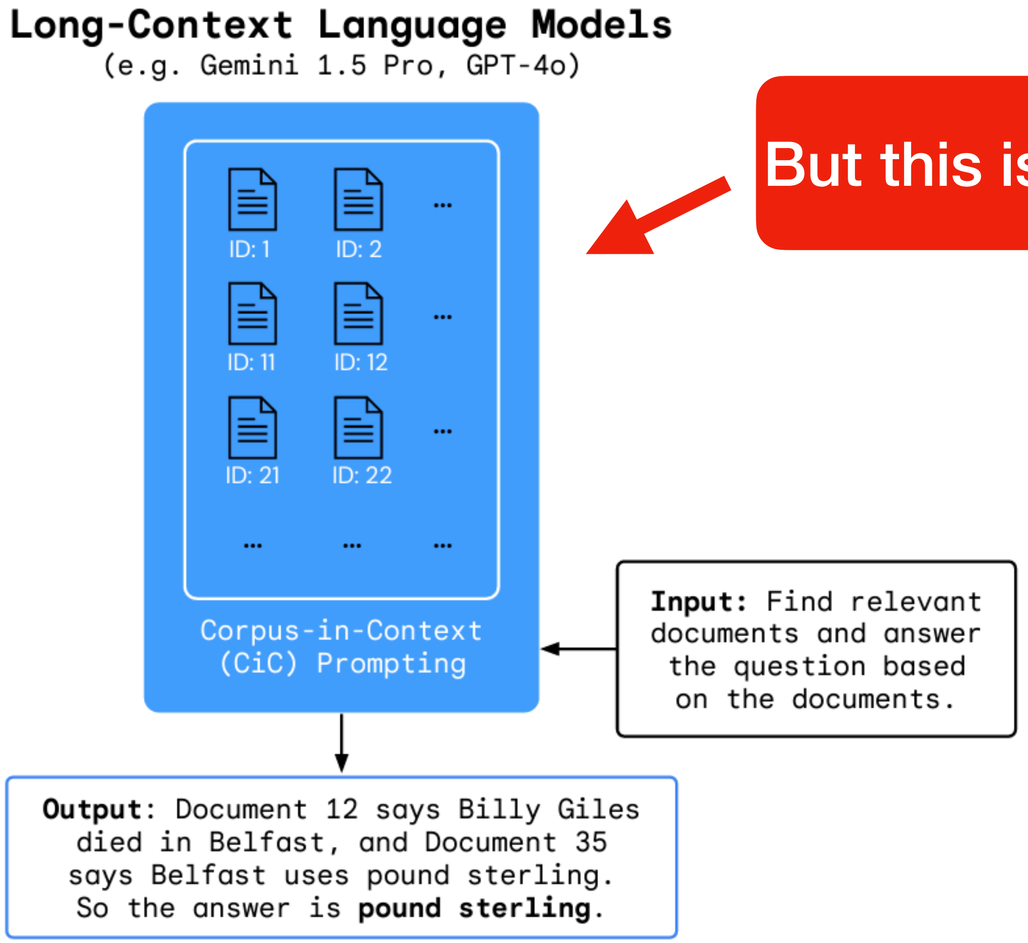
Reflections on: “Can Long-Context Language Models Subsume Retrieval, RAG, SQL, and More?”



Reflections on: “Can Long-Context Language Models Subsume Retrieval, RAG, SQL, and More?”



Reflections on: “Can Long-Context Language Models Subsume Retrieval, RAG, SQL, and More?”



But this is going to be expensive!

Limitations Our experiments were constrained by the computational resources and financial costs associated with utilizing LCLMs. The entire LOFT 128k test sets contain around 35 datasets $\times$ 100 prompts $\times$ 128k tokens $=$ 448M input tokens, which cost $\$1,568$ for Gemini 1.5 Pro, $\$2,240$ for GPT-4o, and $\$6,720$ for Claude 3 Opus at the time of writing. To reduce costs, we also release dev sets, which are 10x smaller and can be evaluated with around $\$200$ using Gemini 1.5 Pro or GPT-4o. We also expect LLM API prices to decrease over time. Another limitation of this work is that we focused on evaluating the quality of LCLMs, and leave efficiency considerations for future work. We could not measure the efficiency improvements from prefix caching [20] due to API constraints at the time of writing. Without caching, the Gemini 1.5 Pro API has a median latency of roughly four seconds for 32k input tokens, twelve seconds for 128k input tokens, and 100 seconds for 1 million input tokens. This speed is likely slower than specialized retrievers or SQL databases; the promising quality results on LOFT encourage further investigation into optimizing LCLMs efficiency. Additionally, our retrieval and RAG tasks was limited to 1 million tokens, which still leaves a large gap from real-world applications that may involve several million or even billions of documents.

Reflections on: “Can Long-Context Language Models Subsume Retrieval, RAG, SQL, and More?”

Long-Context Language Models
(e.g. Gemini 1.5 Pro, GPT-4o)

But this is going to be expensive!

But…

Instruction

```
You will be given a list of documents. You need to read carefully and understand all of them. Then you
will be given a query that may require you to use 1 or more documents to find the answer. Your goal is
to find all documents from the list that can help answer the query.
```

Corpus Formatting

Reference

```
ID: 0 | TITLE: Shinji Okazaki | CONTENT: Shinji Okazaki is a Japanese … | END ID: 0
…
ID: 53 | TITLE: Ain't Thinkin' 'Bout You | CONTENT: "Ain't Thinkin' 'Bout You" is a song … | END ID: 53
ID: 54 | TITLE: Best Footballer in Asia 2016 | CONTENT: … was awarded to Shinji Okazaki … | END ID: 54
…
```

Few-shot Exemples

```
====== Example 1 ======
Which documents are needed to answer the query? Print out the TITLE and ID of each document. Then format
the IDs into a list.
query: What year was the recipient of the 2016 Best Footballer in Asia born?
The following documents are needed to answer the query:
TITLE: Best Footballer in Asia 2016 | ID: 54
TITLE: Shinji Okazaki | ID: 0
Final Answer: [54, 0]
…
```

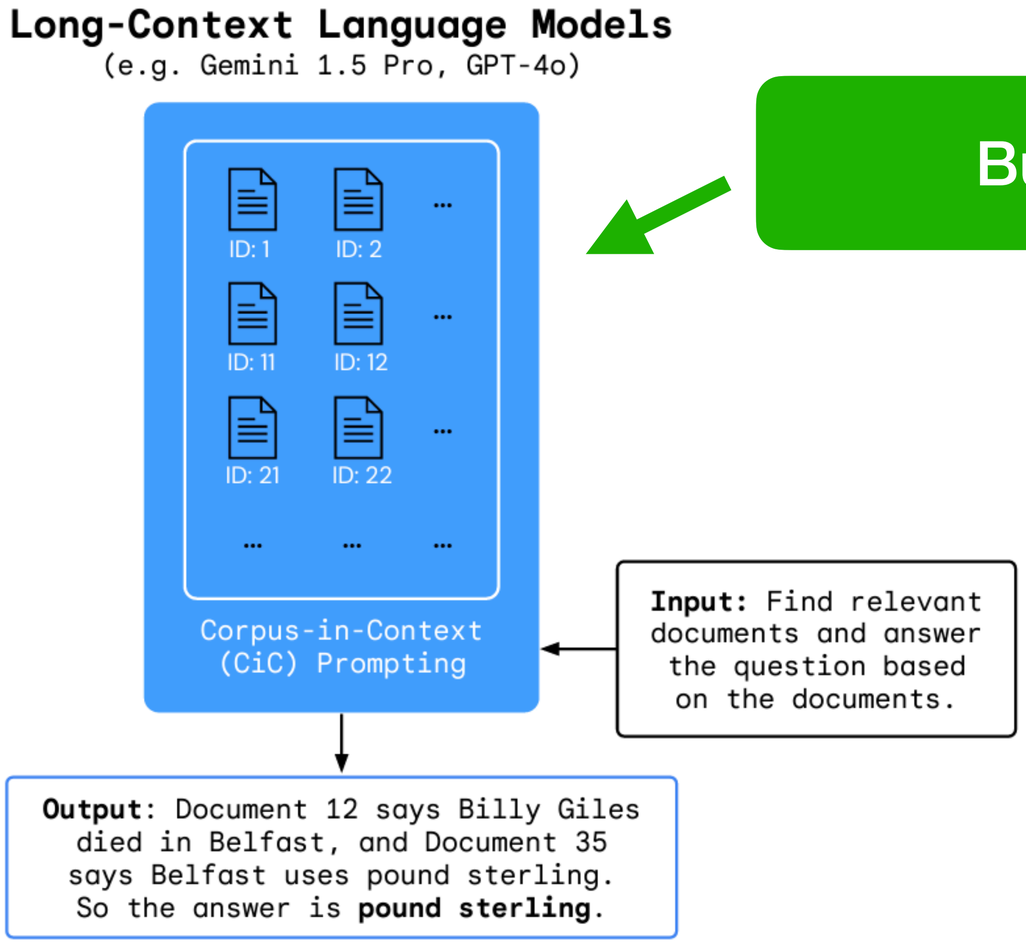
Query Formatting

```
====== Now let's start! ======
Which documents are needed to answer the query? Print out the TITLE and ID of each document. Then format
the IDs into a list.
query: How many records had the team sold before performing "aint thinkin bout you"?
The following documents are needed to answer the query:
```

Limitations Our experiments were constrained by the computational resources and financial costs associated with utilizing LCLMs. The entire LOFT 128k test sets contain around 35 datasets × ens = 448M input tokens, which cost $1,568 for Gemini 1.5 Pro, $2,240 for Claude 3 Opus at the time of writing. To reduce costs, we also release dev ler and can be evaluated with around $200 using Gemini 1.5 Pro or GPT-4o. PI prices to decrease over time. Another limitation of this work is that we he quality of LCLMs, and leave efficiency considerations for future work. he efficiency improvements from prefix caching [20] due to API constraints Without caching, the Gemini 1.5 Pro API has a median latency of roughly put tokens, twelve seconds for 128k input tokens, and 100 seconds for 1 his speed is likely slower than specialized retrievers or SQL databases; the on LOFT encourage further investigation into optimizing LCLMs efficiency. al and RAG tasks was limited to 1 million tokens, which still leaves a large lications that may involve several million or even billions of documents.

Likely to improve further in future

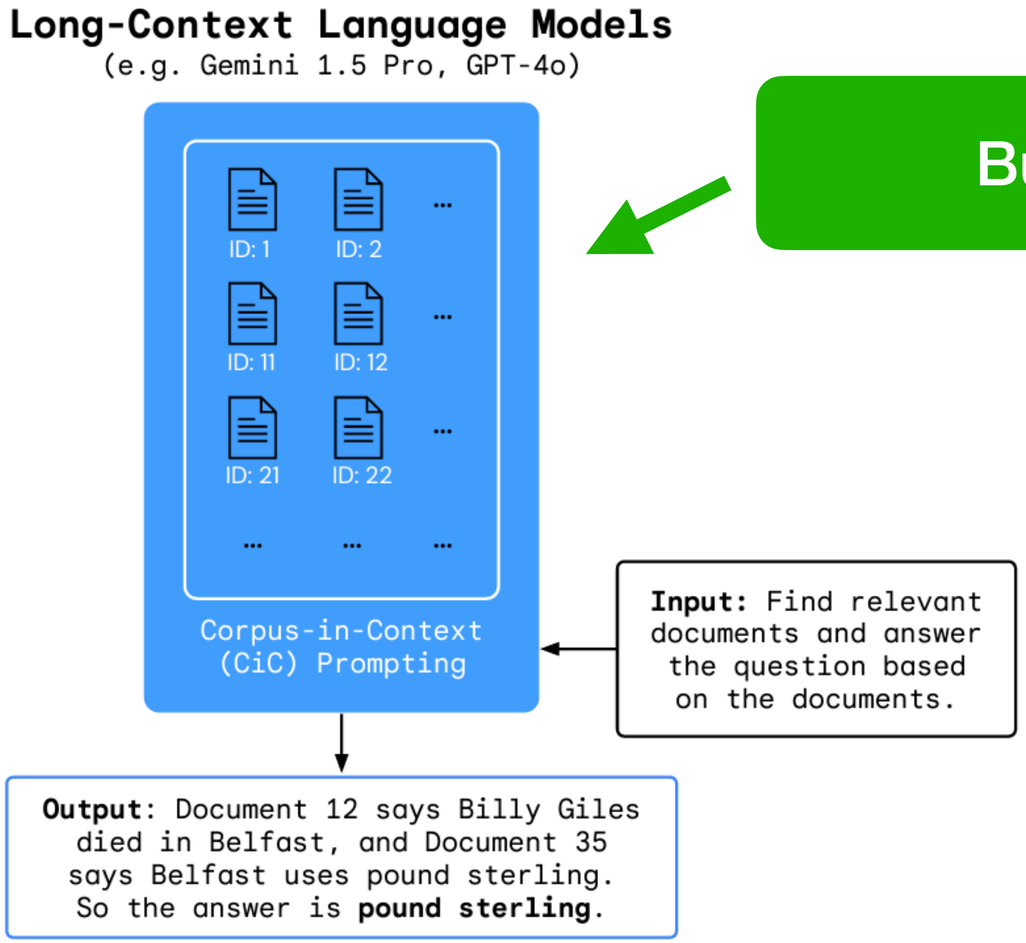
Reflections on: “Can Long-Context Language Models Subsume Retrieval, RAG, SQL, and More?”



But there is more!!

Compared to RAG: No multi-components framework to maintain/train/run

Reflections on: “Can Long-Context Language Models Subsume Retrieval, RAG, SQL, and More?”



“Zero-shot”!

But there is more!!

Compared to RAG: No multi-components framework to maintain/train/run

Compared to DSI: No Training! (“Indexing”)

- Many promising directions for using LLMs to empower search
- Some are radical changes to *“retrieval pipeline status-quo”* and to how we see IR
    - Don’t get tricked by the view that IR is document retrieval! Key to IR is information, not documents
    - i.e. it’s time to move on from Lancaster 1968: “An IR system does not inform (i.e. change the knowledge of) the user on the subject of his inquiry. It merely informs on the existence (or non-existence) and whereabouts of documents relating to his request.'”

No better time to do IR than now!