Challenges and Solutions in Retrieval-Augmented Generation (RAG)

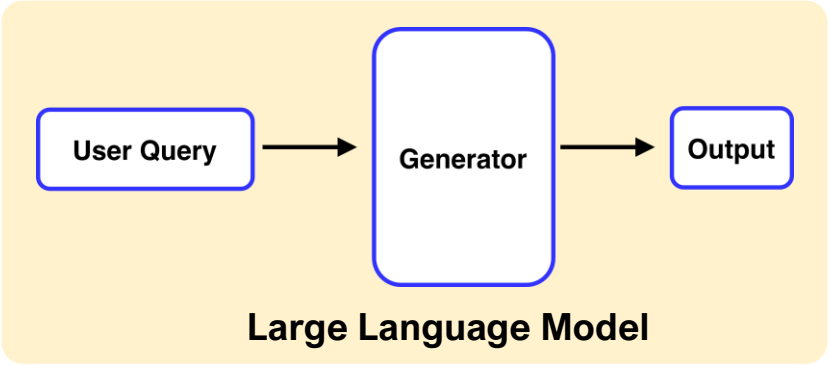
Resource Paper: **Retrieval-Augmented Generation for Large Language Models: A Survey**
Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang

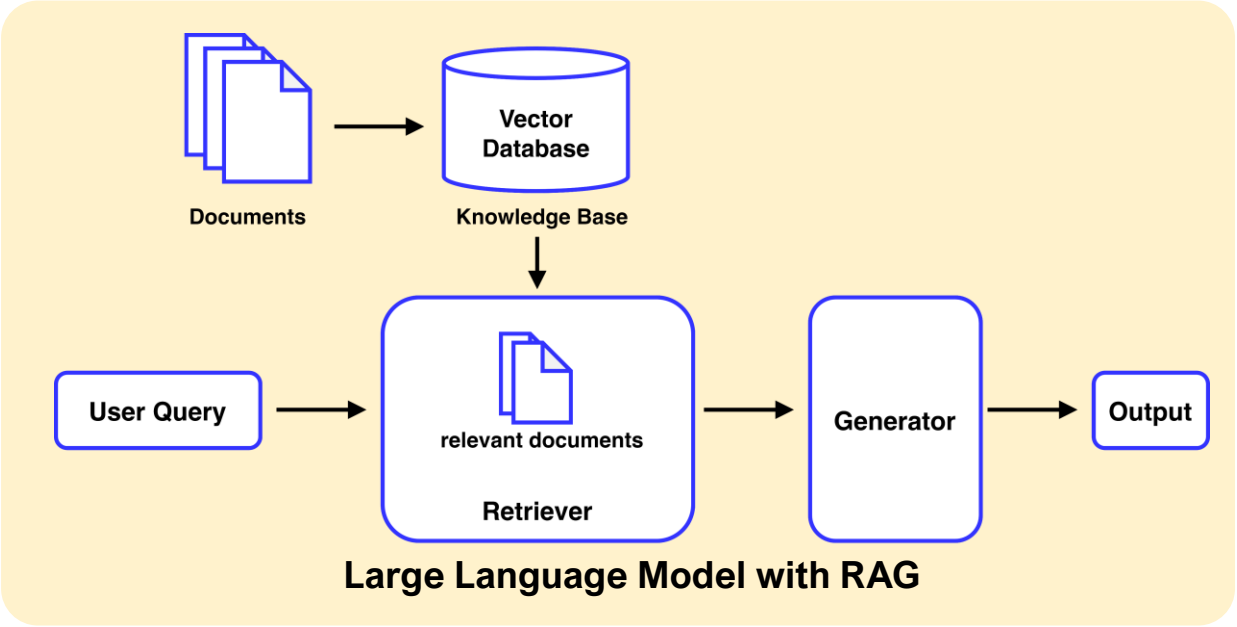
Romen Samuel Rodis Wabina, MSc
PhD(c), Data Science for Healthcare and Clinical Informatics
Department of Clinical Epidemiology and Biostatistics

Overview

- Large Language Models (LLMs) generate hallucinations when handling queries beyond their training data or requiring current information.

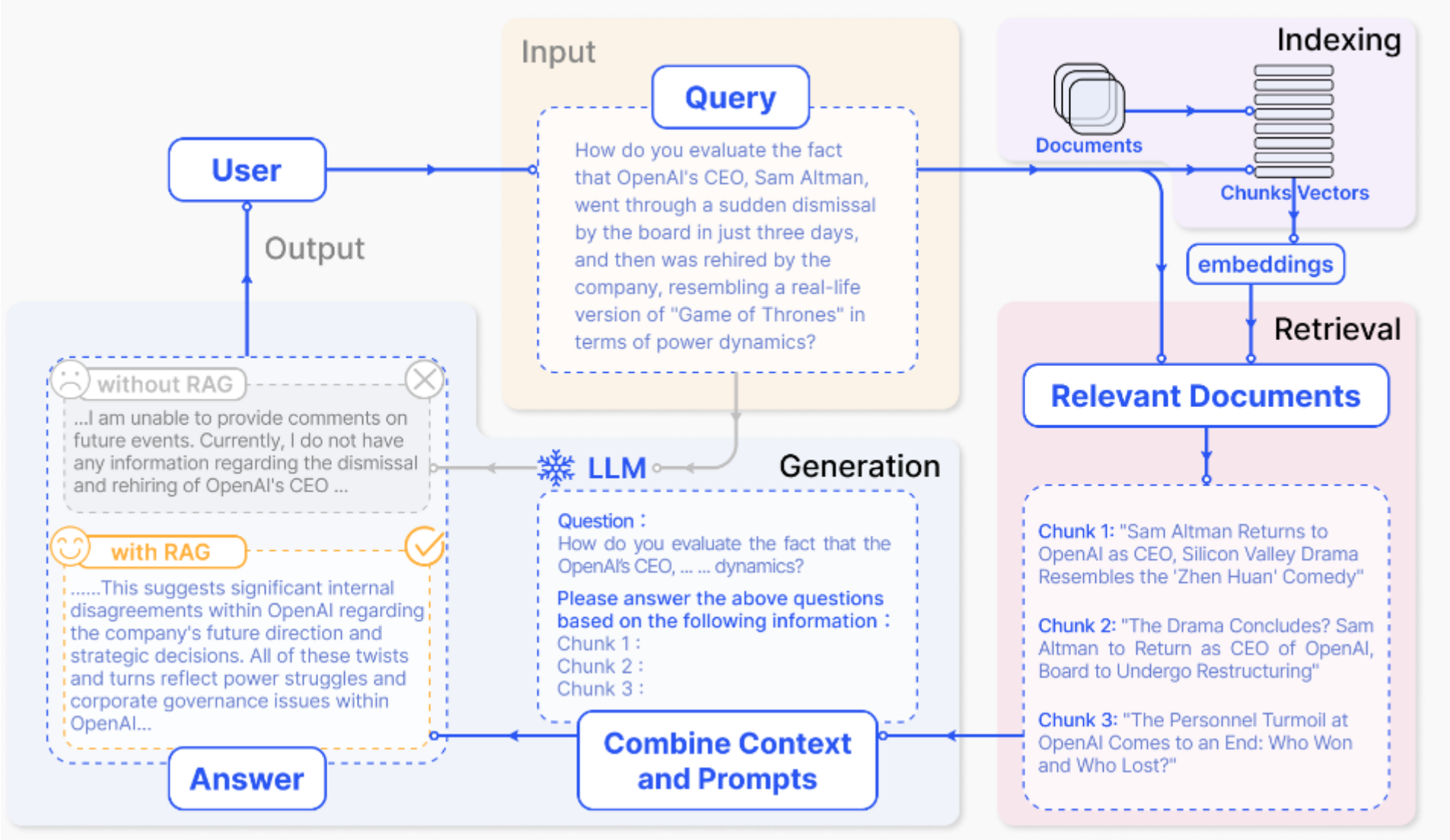
- Retrieval-Augmented Generation (RAG) enhances LLMs by retrieving **document chunks** from external **knowledge base** through semantic similarity.





- As the field of Generative AI continues to evolve, researchers have been exploring various techniques to improve the performance of LLMs in RAG tasks.

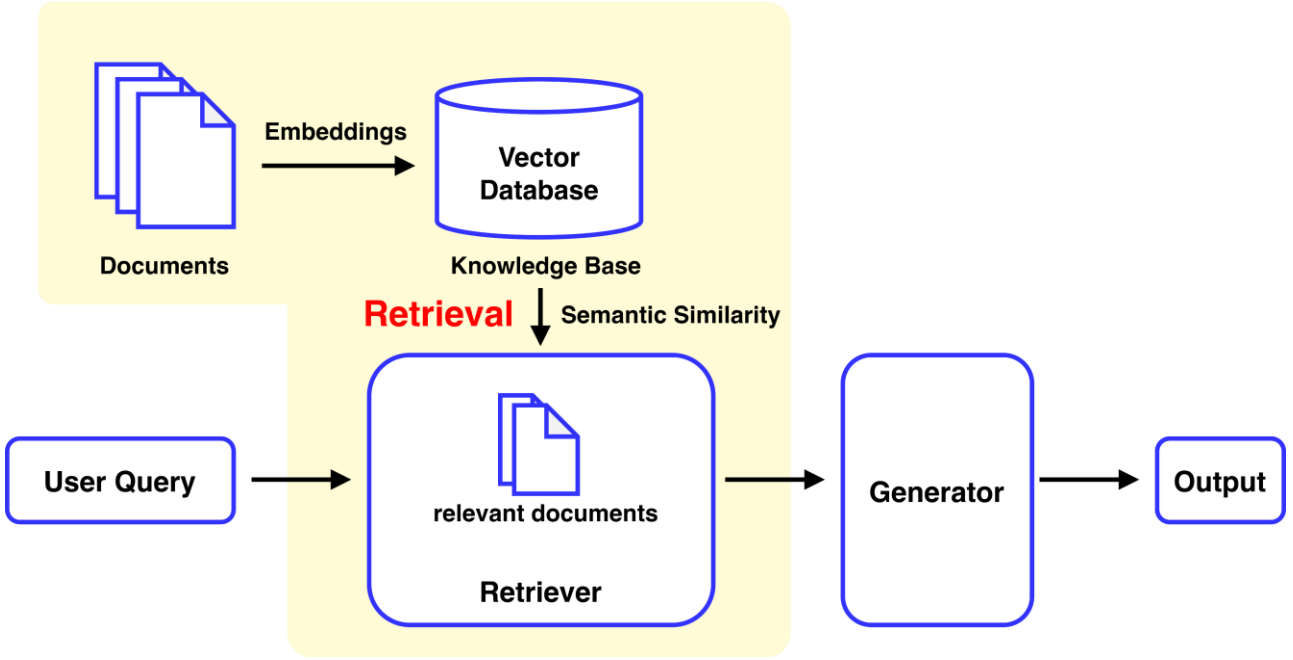
RAG Process



[1] Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., ... & Wang, H. (2023). Retrieval-augmented generation for large language models: A survey. arXiv preprint arXiv:2312.10997.

A. Challenges in RAG

- **Retrieval Challenges**
  - Retrieval phase struggles with **precision** and **recall**, leading to selection of irrelevant chunks of information[1].
  - Inability to retrieve crucial information[1-2]
  - Redundant information
    - Decreased relevance
  - Ambiguous queries
  - Domain-specific jargons

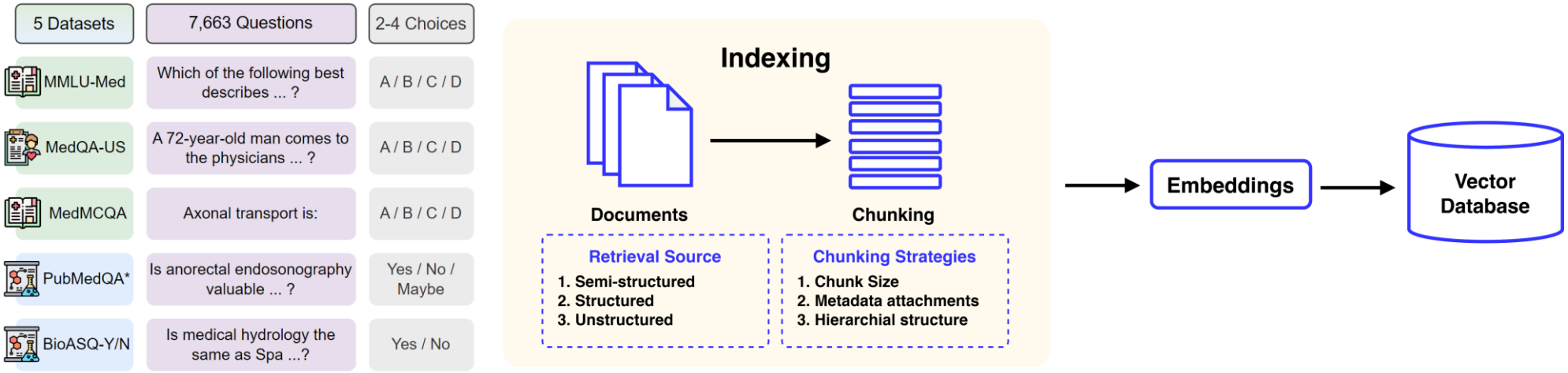


- How to mitigate this problem?
  1. Select appropriate **retrieval sources**[1]
  2. Choose the **right embedding** for RAG[3,5]
  3. Choose the correct **chunk size**[3-4]
  4. Choose the correct **retrieval granularity**[4]
  5. Query **Optimization and Transformation**[1]

[1] Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., ... & Wang, H. (2023). Retrieval-augmented generation for large language models: A survey. arXiv preprint arXiv:2312.10997.
[2] Agrawal, G., Kumarage, T., Alghamdi, Z., & Liu, H. (2024). Mindful-RAG: A Study of Points of Failure in Retrieval Augmented Generation. arXiv preprint arXiv:2407.12216.
[3] Khanna, S., & Subedi, S. (2024). Tabular Embedding Model (TEM): Finetuning Embedding Models For Tabular RAG Applications. arXiv preprint arXiv:2405.01585.
[4] Chen, T., Wang, H., Chen, S., Yu, W., Ma, K., ... & Zhang, H. (2023). Dense x retrieval: What retrieval granularity should we use?. arXiv preprint arXiv:2312.06648.
[5] Balikas, G. (2023, October). Comparative Analysis of Open Source and Commercial Models for Question Answering. *32nd International Conference on Information and Knowledge Management*.

1. Select appropriate retrieval sources

- RAG relies on external knowledge from different document sources to enhance LLMs.



- Use different **domain-specific document sources**:
  - Question-Answering (QA) tasks: Wikipedia, Arxiv, HotpotQA[1,2]
  - Xiong et al. (2024) used six common datasets on medical QA tasks for MIRAGE (medical RAG)[3]
    - PubMedQA, MedCorp, MedQA, MMLU-Med, MedMCQA, BioASQ-Y/N
    - Except PubMedQA, **other datasets are not recommended due to limited volumes of knowledge**.

  - Semi-structured data poses challenges:
    - Text splitting separates tables, leading to information corruption.
    - Incorporating tables with text can complicate embeddings and semantic similarity searches.

[1] Wang, X., Wang, Z., Gao, X., Zhang, F., Wu, Y., Xu, Z., ... & Huang, X. (2024). Searching for Best Practices in Retrieval-Augmented Generation. arXiv:2407.01219.
[2] Xiong, G., Jin, Q., Lu, Z., & Zhang, A. (2024). Benchmarking retrieval-augmented generation for medicine. arXiv preprint arXiv:2402.13178.
[3] Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., ... & Wang, H. (2023). Retrieval-augmented generation for large language models: A survey. arXiv preprint arXiv:2312.10997.

2. How to choose the right embedding model for RAG?

- The retrieval phase misses crucial information from the external knowledge base.

- Common embedding models[1-3] :
  - Sparse: TF-IDF, bag-of-words models, BestMatch25
  - Dense: Word2Vec, GloVe, BERT, RoBERTa, GPT3
  - Prominent embedding models from multi-task instruct tuning:
    - OpenAI's GPT-3, GPT-4
    - Angle-Optimized Text Embedding (AnglE), Voyage, BGE

- **Massive Text Embedding Benchmark (MTEB) Leaderboard** on Hugging Face[1]
  - Updated list of open-source text embedding models
    - Accuracy: GPT-3, GPT-4, RoBERTa, or Multilingual E5-Large
    - Speed: MiniLM or DistilBERT

    - Domain-specific models
      - BioBERT, PubMedBERT (PubMed abstracts and PMC articles)
      - ClinicalBERT (MIMIC-III database)

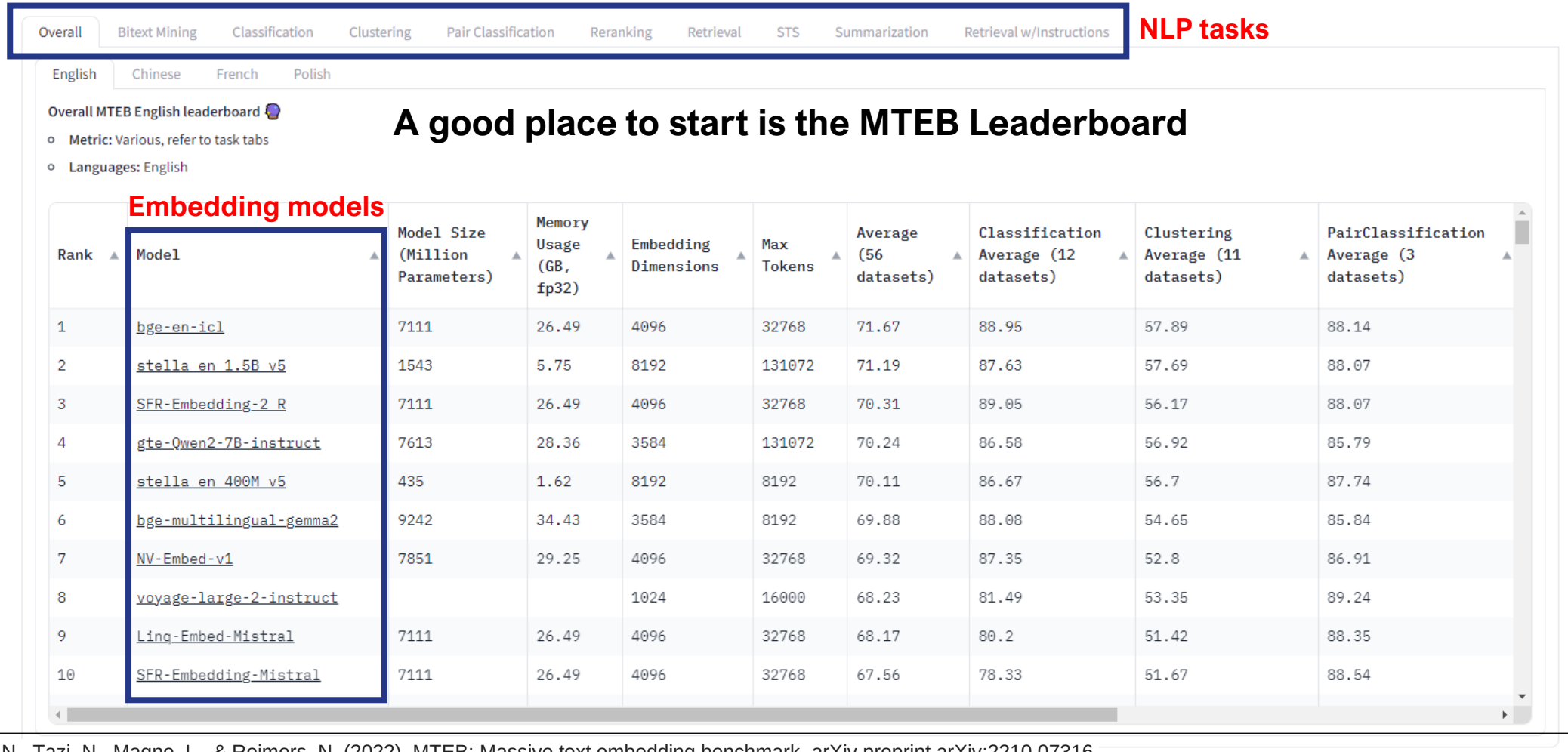
- **Fine-tune** the embedding model

[1] Muennighoff, N., Tazi, N., Magne, L., & Reimers, N. (2022). MTEB: Massive text embedding benchmark. arXiv preprint arXiv:2210.07316.
[2] Rau, D., Wang, S., Déjean, H., & Clinchant, S. (2024). Context Embeddings for Efficient Answer Generation in RAG. arXiv preprint arXiv:2407.09252.
[3] Finardi, P., Avila, L.,... & Caridá, V. (2024). The Chronicles of RAG: The Retriever, the Chunk and the Generator. arXiv preprint arXiv:2401.07883.

2. How to choose the right embedding model for RAG?

Massive Text Embedding Benchmark (MTEB) Leaderboard on Hugging Face

- Updated list of open-source text embedding models



Rank	Model	Model Size (Million Parameters)	Memory Usage (GB, fp32)	Embedding Dimensions	Max Tokens	Average (56 datasets)	Classification Average (12 datasets)	Clustering Average (11 datasets)	PairClassification Average (3 datasets)
1	bge-en-icl	7111	26.49	4096	32768	71.67	88.95	57.89	88.14
2	stella_en_1.5B_v5	1543	5.75	8192	131072	71.19	87.63	57.69	88.07
3	SFR-Embedding-2_R	7111	26.49	4096	32768	70.31	89.05	56.17	88.07
4	gte-Qwen2-7B-instruct	7613	28.36	3584	131072	70.24	86.58	56.92	85.79
5	stella_en_400M_v5	435	1.62	8192	8192	70.11	86.67	56.7	87.74
6	bge-multilingual-gemma2	9242	34.43	3584	8192	69.88	88.08	54.65	85.84
7	NV-Embed-v1	7851	29.25	4096	32768	69.32	87.35	52.8	86.91
8	voyage-large-2-instruct			1024	16000	68.23	81.49	53.35	89.24
9	Linq-Embed-Mistral	7111	26.49	4096	32768	68.17	80.2	51.42	88.35
10	SFR-Embedding-Mistral	7111	26.49	4096	32768	67.56	78.33	51.67	88.54

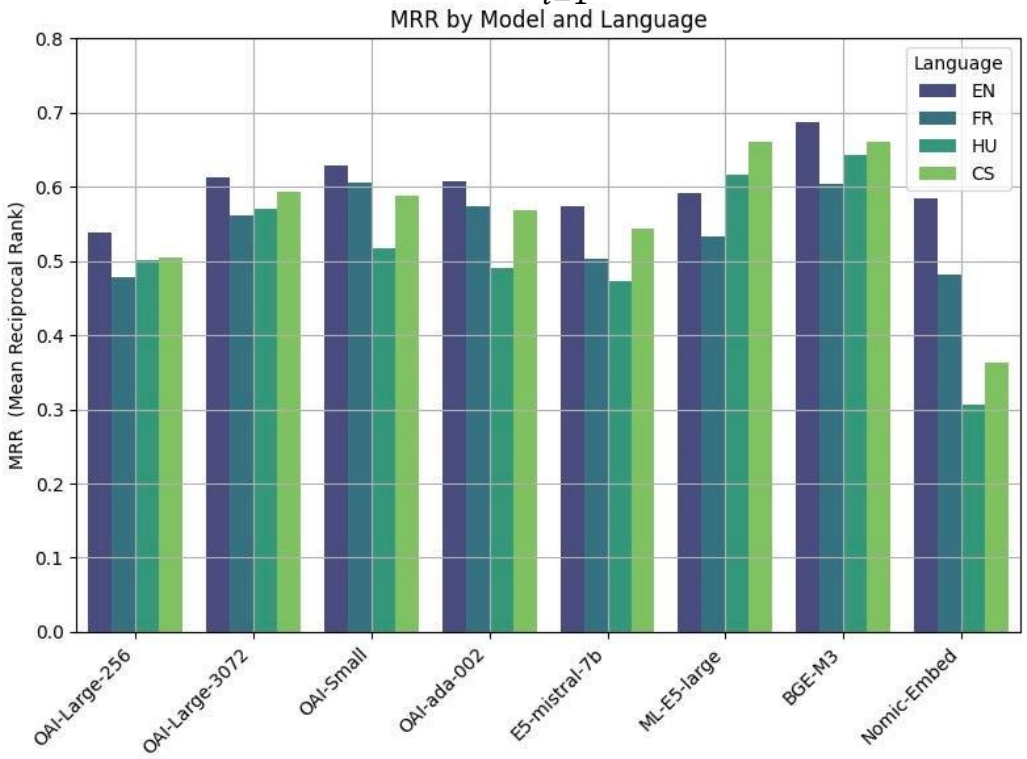
[1] Muennighoff, N., Tazi, N., Magne, L., & Reimers, N. (2022). MTEB: Massive text embedding benchmark. arXiv preprint arXiv:2210.07316.

2. How to choose the right embedding model for RAG?

Comparison of OpenAI vs open-source embedding models

- Mileva et al. (2024) compared OpenAI's embedding models against open-source models[2]

$$\text{MRR} = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{\text{rank}_i}$$



- OpenAI embedding models:
  - OAI-Large-256, OAI-Large-3072, OAI-small, OAI-ada
- Open-source models:
  - Mistral, Multilingual E5-Large, BGE-M3, Nomic

- **Best performances were obtained by open-source models**
  - BGE-M3 emerged as the top performer
  - All models (except Multilingual-E5-Large) performed best on English.

- **Open-source models offer high performance, OpenAI's models prioritizes convenience over privacy**[1-3].

[1] Balikas, G. (2023, October). Comparative Analysis of Open Source and Commercial Embedding Models for Question Answering. In Proceedings of the 32nd ACM International Conference on Information and Knowledge Management (pp. 5232-5233).
[2] Mileva, G. (2023, July 4). OpenAI vs open-source: Multilingual embedding models. Towards Data Science. https://towardsdatascience.com/openai-vs-open-source-multilingual-embedding-models-e5ccb7c90f05
[3] Yao, X. & Liu, J. (2023, October). Towards Robust Token Embeddings for Extractive Question Answering. In International Conference on Web Information Systems Engineering (pp. 82-96). Springer Nature

3. Which chunk size is good?

- Chunking is breaking down large pieces of text into smaller segments.
- Choosing the right chunking strategy for RAG: **chunk size** (vector embedding length)

Wang et al. (2024) evaluated different chunk sizes using GPT3.5 using only on **Lyft (2021)** dataset for QA tasks[1] .

- Larger chunk sizes (e.g., 2048 words) tend to include more irrelevant information, reducing faithfulness.
- Small chunks (e.g., 128 words) may lack sufficient context, slightly impacting faithfulness but maintaining high relevancy.

Chunk Size	lyft_2021	
	Average Faithfulness	Average Relevancy
2048	80.37	91.11
1024	94.26	95.56
512	**97.59**	97.41
256	97.22	**97.78**
128	95.74	97.22

- **Balancing semantic completeness with the context length limits of LLMs is challenging.**
  - Wang et al. (2024) proposed Small2Big[1]
    - Small chunks are used as a retrieved unit, then the following chunks are provided as big context to LLM

[1] Wang, X., Wang, Z., Gao, X., Zhang, F., Wu, Y., Xu, Z., ... & Huang, X. (2024). Searching for Best Practices in Retrieval-Augmented Generation. arXiv:2407.01219.
[2] Xiong, G., Jin, Q., Lu, Z., & Zhang, A. (2024). Benchmarking retrieval-augmented generation for medicine. arXiv preprint arXiv:2402.13178.
[3] Chen, T., Wang, H., Chen, S., Yu, W., Ma, K., ... & Zhang, H. (2023). Dense x retrieval: What retrieval granularity should we use?. arXiv preprint arXiv:2312.06648.

4. What retrieval granularity should we use?

- Chen et al. (2023) proposed that propositions should be used as retrieval granularity[2] .
  - Propositions represent smallest unit of meaningful information

Passage Retrieval	Sentence Retrieval	Proposition Retrieval
Q1: What was the theme of Super Bowl 50?		
Title: Super Bowl X ✗ The overall theme of the Super Bowl entertainment was to celebrate the United States Bicentennial. Each Cowboys and Steelers player wore a special patch with the Bicentennial logo on their jerseys...	Title: Super Bowl X ✗ The overall theme of the Super Bowl entertainment was to celebrate the United States Bicentennial.	Title: Super Bowl XLV ✓ ... As this was the 50th Super Bowl game, the league [Super Bowl 50] emphasized the "golden anniversary" with various gold-themed initiatives during the 2015 season, as well as...

  - Proposition retrieval yields the highest recall, indicating its effectiveness in relevant information retrieval.
    - Models: SimCSE (BERT-base, *Wikipedia*), Contriever (BERT-base, *Wikipedia*)
    - Performance metric: Recall@*K* (no. of relevant docs in top 5/total number of relevant docs)

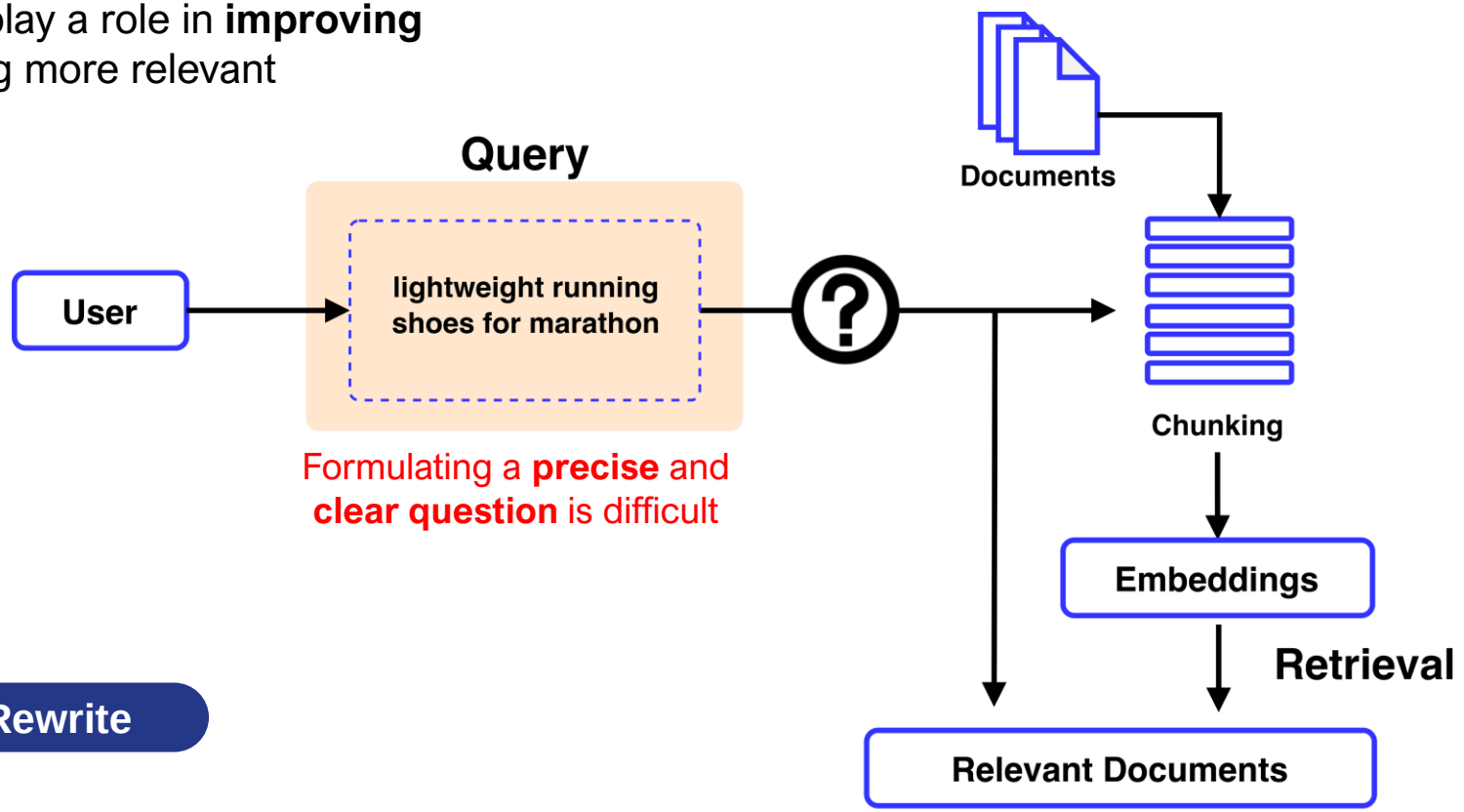
Retriever	Granularity	NQ		TQA		WebQ		SQuAD		EQ		Avg.	
		R@5	R@20	R@5	R@20	R@5	R@20	R@5	R@20	R@5	R@20	R@5	R@20
Unsupervised Dense Retrievers													
SimCSE	Passage	28.8	44.3	44.9	59.4	39.8	56.0	29.5	45.5	28.4	40.3	34.3	49.1
	Sentence	35.5	53.1	50.5	64.3	45.3	64.1	37.1	52.3	36.3	50.1	40.9	56.8
	Proposition	**41.1**	**58.9**	**52.4**	**66.5**	**50.0**	**66.8**	**38.7**	**53.9**	**49.5**	**62.2**	**46.3**	**61.7**
Contriever	Passage	42.5	63.8	58.1	73.7	37.1	60.6	40.8	59.8	36.3	56.3	43.0	62.8
	Sentence	46.4	66.8	60.6	75.7	41.7	63.1	45.1	63.5	42.7	61.3	47.3	66.1
	Proposition	**50.1**	**70.0**	**65.1**	**77.9**	**45.9**	**66.8**	**50.7**	**67.7**	**51.7**	**70.1**	**52.7**	**70.5**

[1] Xiong, G., Jin, Q., Lu, Z., & Zhang, A. (2024). Benchmarking retrieval-augmented generation for medicine. arXiv preprint arXiv:2402.13178.
[2] Chen, T., Wang, H., Chen, S., Yu, W., Ma, K., ... & Zhang, H. (2023). Dense x retrieval: What retrieval granularity should we use?. arXiv preprint arXiv:2312.06648.

5. Query Optimization and Transformation

Query **optimization and transformation** play a role in **improving the retrieval effectiveness** and generating more relevant answers in RAG[1-2].

The main goal is to ***address the challenges associated with imprecise, complex, or ambiguous user queries***[2-3]



Methods for query optimization:

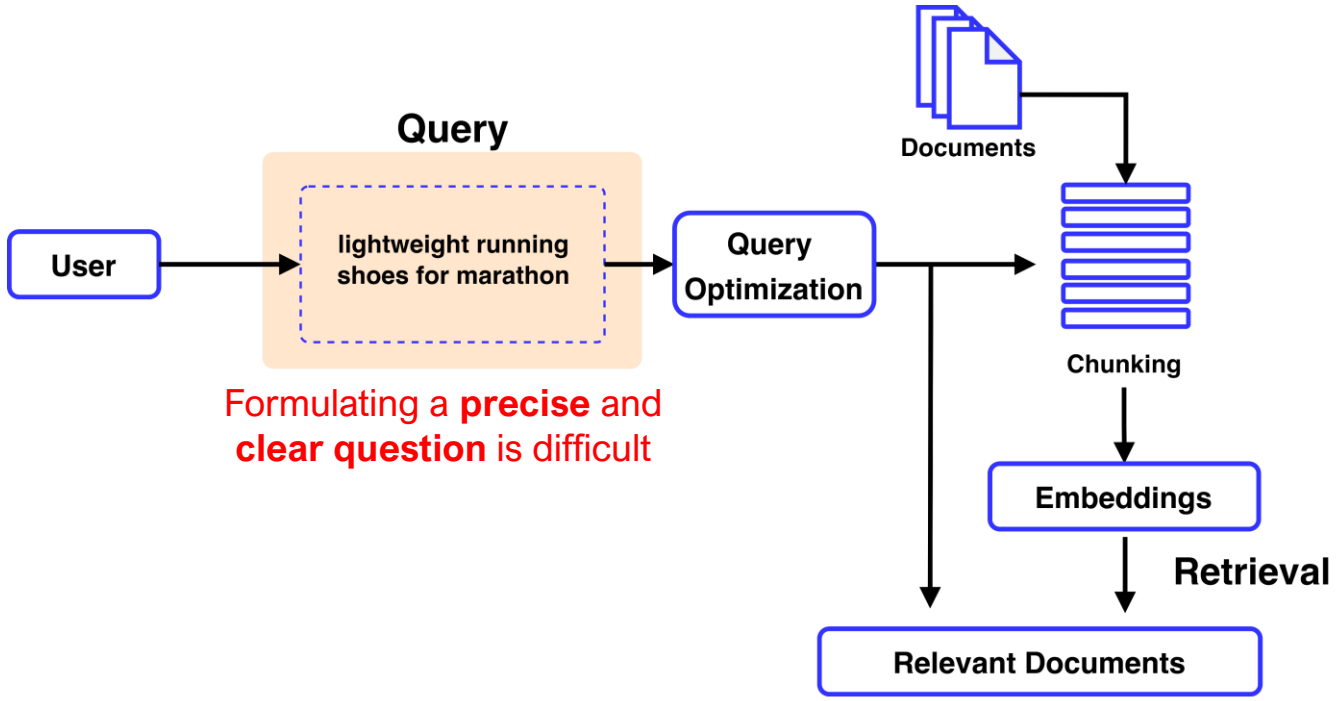
Query Expansion **Query Rewrite**

By expanding queries and transforming them, RAG can better understand user intent and retrieve the most relevant information.

[1] Sawarkar, K. (2024). Blended RAG: Improving RAG (Retriever-Augmented Generation) Accuracy with Semantic Search and Hybrid Query-Based Retrievers. *arXiv preprint arXiv:2404.07220*.
[2] Koo, H., Kim, M., & Hwang, S. J. (2024). Optimizing Query Generation for Enhanced Document Retrieval in RAG. *arXiv preprint arXiv:2407.12325*.
[3] Kulkarni, M., Tangarajan, P., Kim, K., & Trivedi, A. (2024). Reinforcement Learning for Optimizing RAG for Domain Chatbots. *arXiv preprint arXiv:2401.06800*.

5. Query Optimization and Transformation

Query Expansion Expands a single query into multiple queries to enrich the content and provide further context.



Query lightweight running shoes for marathons

Sub-queries: breaks down the original query into sub-questions:

- What are the best lightweight running shoes?
- What running shoes are recommended for marathons?
- What are the features of running shoes for long-distance run?

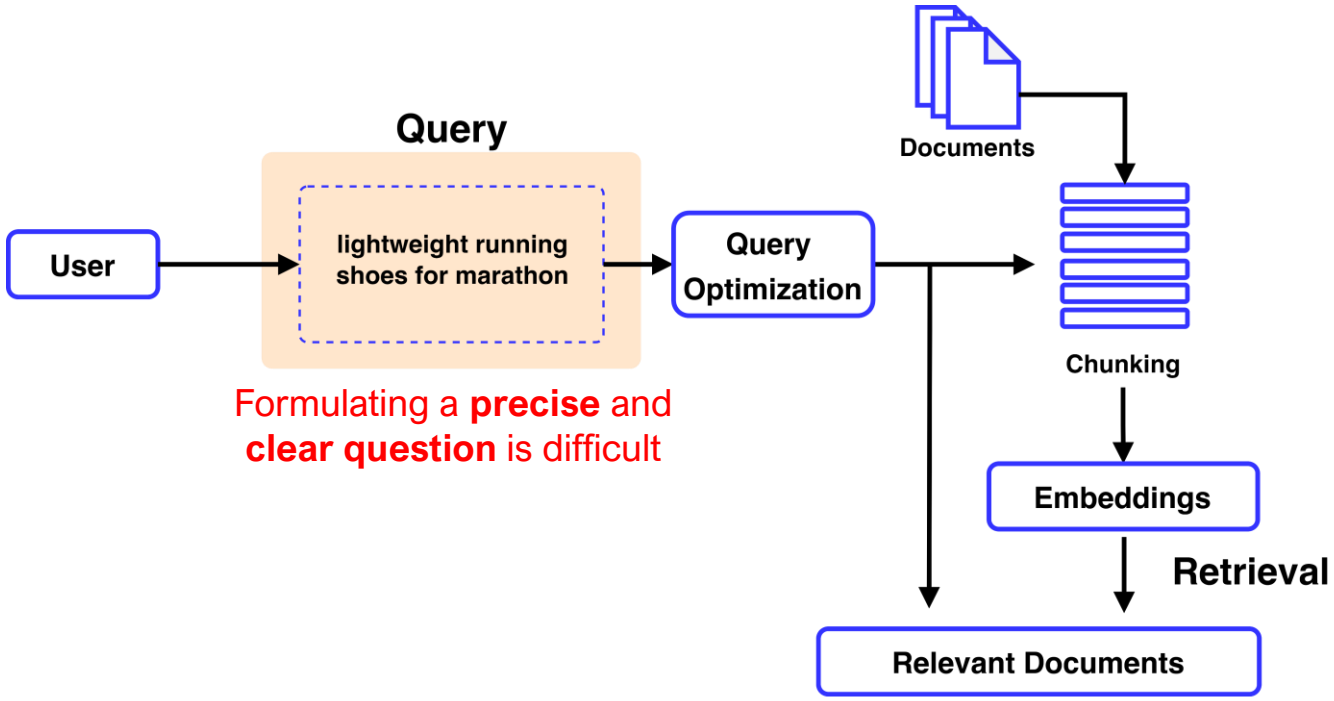
Multi-Query: generates multiple queries to capture different aspects of the original query.

- lightweight running shoes for long-distance running
- running shoes with good cushioning for marathons
- durable running shoes for marathon training

[1] Sawarkar, K. (2024). Blended RAG: Improving RAG (Retriever-Augmented Generation) Accuracy with Semantic Search and Hybrid Query-Based Retrievers. *arXiv preprint arXiv:2404.07220*.
[2] Koo, H., Kim, M., & Hwang, S. J. (2024). Optimizing Query Generation for Enhanced Document Retrieval in RAG. *arXiv preprint arXiv:2407.12325*.
[3] Kulkarni, M., Tangarajan, P., Kim, K., & Trivedi, A. (2024). Reinforcement Learning for Optimizing RAG for Domain Chatbots. *arXiv preprint arXiv:2401.06800*.

5. Query Optimization and Transformation

Query Rewrite The system rewrites the original query to make it more suitable for retrieval



Original Query lightweight running shoes for marathons

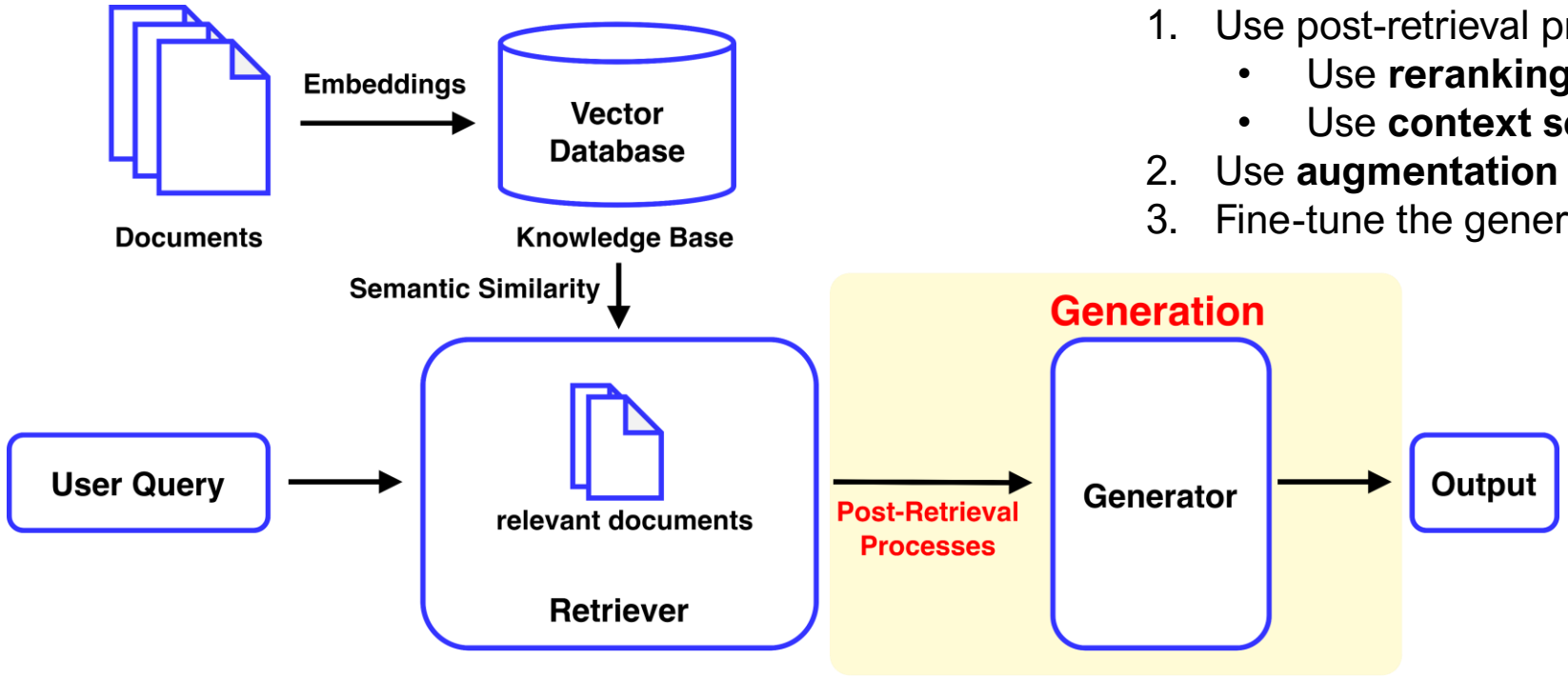
Rewritten Query best running shoes for marathon racing with lightweight design

[1] Sawarkar, K. (2024). Blended RAG: Improving RAG (Retriever-Augmented Generation) Accuracy with Semantic Search and Hybrid Query-Based Retrievers. *arXiv preprint arXiv:2404.07220*.
[2] Koo, H., Kim, M., & Hwang, S. J. (2024). Optimizing Query Generation for Enhanced Document Retrieval in RAG. *arXiv preprint arXiv:2407.12325*.
[3] Kulkarni, M., Tangarajan, P., Kim, K., & Trivedi, A. (2024). Reinforcement Learning for Optimizing RAG for Domain Chatbots. *arXiv preprint arXiv:2401.06800*.

B. Generation Challenges in RAG

- **Generation Challenges**
  - LLMs face **hallucinations** where it produces information not supported by the retrieved content
  - May encounter **redundancy** when similar information is retrieved from multiple sources.
  - **Lost-in-the-middle** problem

- How to mitigate this problem?
  1. Use post-retrieval processes
     - Use **reranking methods**
     - Use **context selection/compression**
  2. Use **augmentation**
  3. Fine-tune the generator



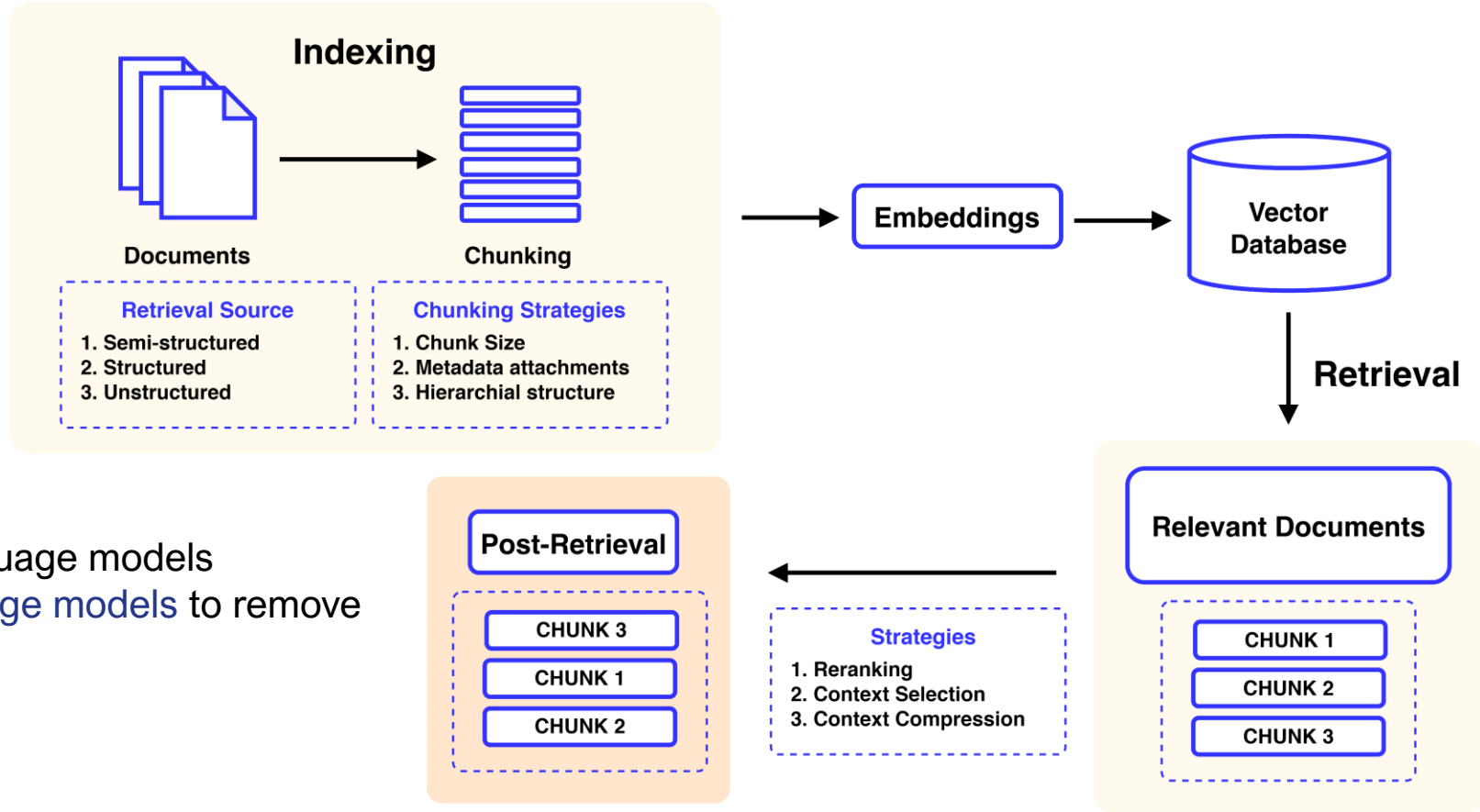
[1] Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., ... & Wang, H. (2023). Retrieval-augmented generation for large language models: A survey. arXiv preprint arXiv:2312.10997.
[2] Agrawal, G., Kumarage, T., Alghamdi, Z., & Liu, H. (2024). Mindful-RAG: A Study of Points of Failure in Retrieval Augmented Generation. arXiv preprint arXiv:2407.12216.
[3] Khanna, S., & Subedi, S. (2024). Tabular Embedding Model (TEM): Finetuning Embedding Models For Tabular RAG Applications. arXiv preprint arXiv:2405.01585.
[4] Chen, T., Wang, H., Chen, S., Yu, W., Ma, K., ... & Zhang, H. (2023). Dense x retrieval: What retrieval granularity should we use?. arXiv preprint arXiv:2312.06648.
[5] Balikas, G. (2023, October). Comparative Analysis of Open Source and Commercial Models for Question Answering. *32nd International Conference on Information and Knowledge Management*.

1. Use post-retrieval processes

Post-retrieval strategies to improve generation process:

- **Use reranking methods**
  - Rule-based: Diversity, Relevance
  - Mean Retrieval Rank
  - Cohere Rerank, Score Fusion
  - Cross-Encoder

- **Context Selection or Compression**
  - Cosine Similarity
  - Abstractive Summarization
  - Token Assessment using small language models
    - LLMLingua utilize small language models to remove unimportant tokens
    - SLMs: GPT-2, LLaMA-7B



[1] Xiong, G., Jin, Q., Lu, Z., & Zhang, A. (2024). Benchmarking retrieval-augmented generation for medicine. arXiv preprint arXiv:2402.13178.

[1] Chen, T., Wang, H., Chen, S., Yu, W., Ma, K., ... & Zhang, H. (2023). Dense x retrieval: What retrieval granularity should we use?. *arXiv preprint arXiv:2312.06648*.

1. Use post-retrieval processes

- Traditional RAG uses bi-encoder models for reranking.
  - Separate encoding, loss of context

- Reranking method: **Cross-Encoder** model
- Cross-encoder: pair of inputs (e.g., query + input)
  - Bi-encoder processes each input separately
- PLM encoder*: BERT-based models
- Output: **Relevance score**

- Disadvantages (cross-encoder):
  - Cross-encoder can return redundant passages[1]
  - Computationally-expensive than bi-encoders[2]

- How to mitigate redundancy?
  - Ensemble approach: multiple reranking strategies and combine them[2-3]

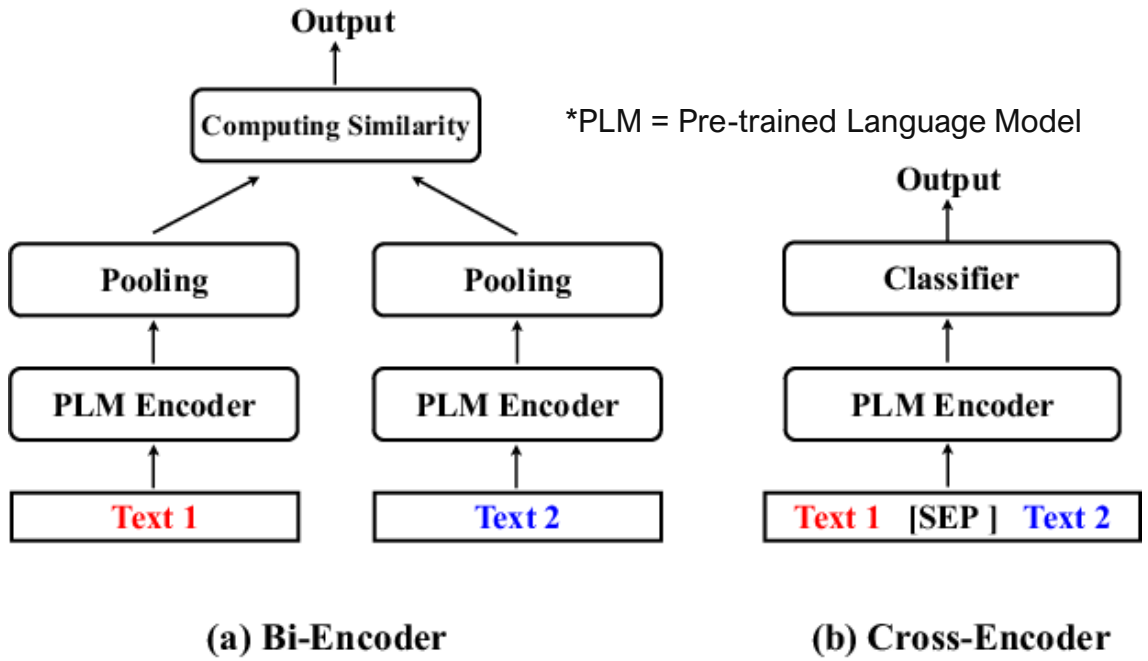
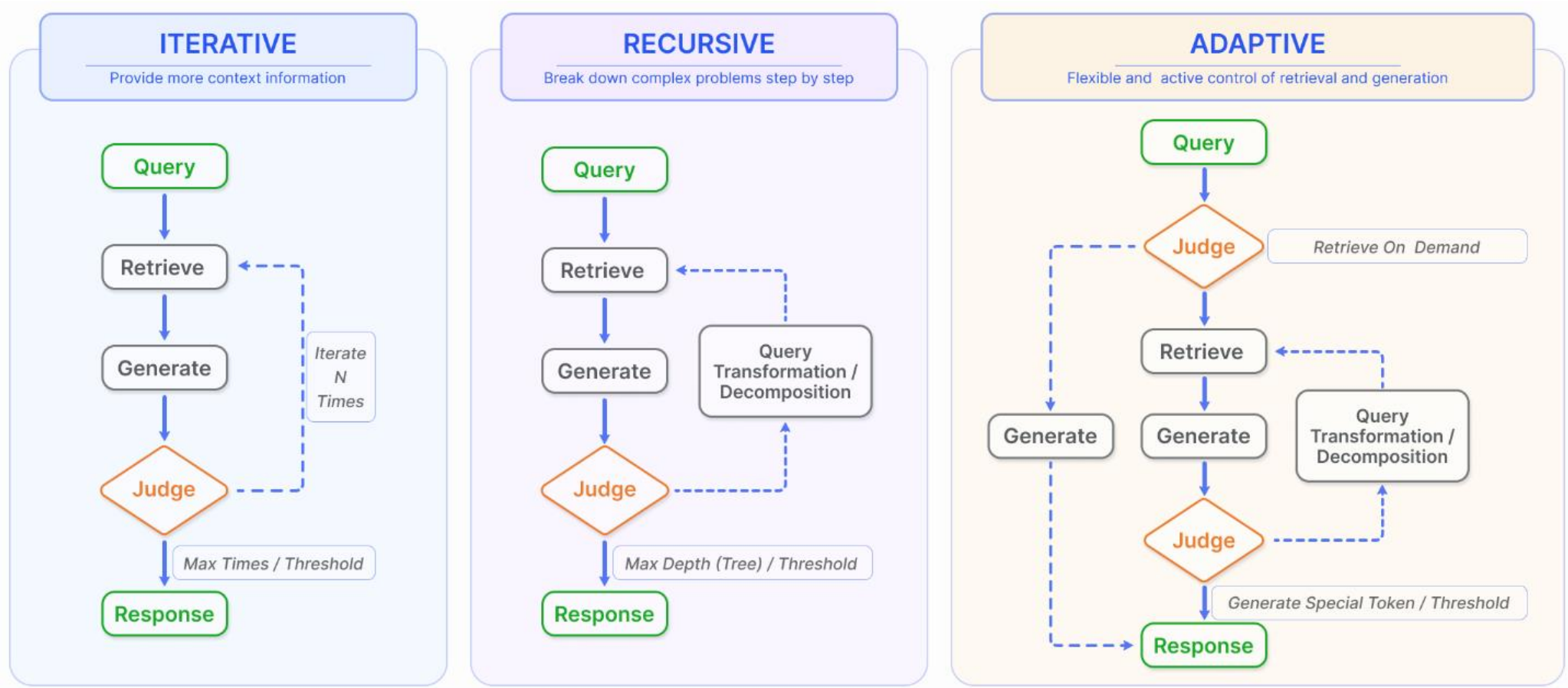


Figure adapted from: FBC: Fusing Bi-Encoder and Cross-Encoder for Long-Form Text Matching[4]

[1] Pickett, M., Hartman, J., Bhowmick, A. K., Alam, R. U., & Vempaty, A. (2024). Better RAG using Relevant Information Gain. arXiv preprint arXiv:2407.12101.
[2] Sun, X., Yu, L., Wang, Y., Bi, K., & Guo, J. (2023). Ensemble Ranking Model with Multiple Pretraining Strategies for Web Search. arXiv preprint arXiv:2302.09340.
[3] Borges, L & Callan, J. (2021, March). Assessing the Benefits of Model Ensembles in Neural Re-ranking for Passage Retrieval. In European Conference on Information Retrieval (pp. 225-232). Cham: Springer
[4] Liao, J., Jia, M., Duan, J., & Wang, J. (2023). FBC: Fusing Bi-Encoder and Cross-Encoder for Long-Form Text Matching. In ECAI (pp. 1473-1480).

2. Use augmentation

- A single retrieval may not suffice to acquire adequate context information.
  - RAGs can use **augmentation** to integrate context from retrieved passages with the current generation task



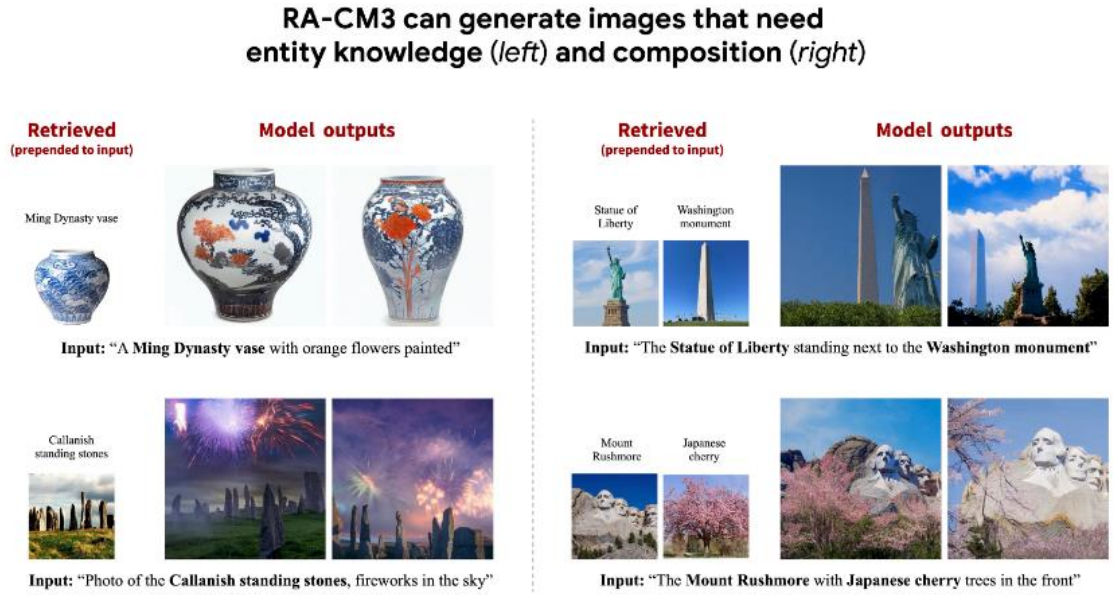
[1] Pickett, M., Hartman, J., Bhowmick, A. K., Alam, R. U., & Vempaty, A. (2024). Better RAG using Relevant Information Gain. *arXiv preprint arXiv:2407.12101.*
[2] Sun, X., Yu, L., Wang, Y., Bi, K., & Guo, J. (2023). Ensemble Ranking Model with Multiple Pretraining Strategies for Web Search. *arXiv preprint arXiv:2302.09340*.

C. Future Works in RAG

1. **Multimodal RAG**
   - Several research has created vision-language RAGs that can retrieve and generate text and images
     - RA-CM3 and BLIP-2 stand as a pioneering multimodal for vision-language RAG models
     - These models can be used for visual QA, image captioning, and multimodal retrieval.

   - RAG systems for audio and video data

2. **Production-ready RAG**
   - LangChain and LLamaIndex



[1] Pickett, M., Hartman, J., Bhowmick, A. K., Alam, R. U., & Vempaty, A. (2024). Better RAG using Relevant Information Gain. *arXiv preprint arXiv:2407.12101.*
[2] Sun, X., Yu, L., Wang, Y., Bi, K., & Guo, J. (2023). Ensemble Ranking Model with Multiple Pretraining Strategies for Web Search. *arXiv preprint arXiv:2302.09340*.
[3] Borges, L & Callan, J. (2021, March). Assessing the Benefits of Model Ensembles in Neural Re-ranking for Passage Retrieval. In *European Conference on Information Retrieval* (pp. 225-232). Cham: Springer International Publishing.
[4] Liao, J., Jia, M., Duan, J., & Wang, J. (2023). FBC: Fusing Bi-Encoder and Cross-Encoder for Long-Form Text Matching. In *ECAI* (pp. 1473-1480).

D. Conclusion

- The combination of LLMs and RAG has emerged as a **powerful approach for producing informed and contextual responses.**
- Despite the progress in RAG, there are research opportunities to improve its robustness and its ability to handle extended contexts.
- Further research with this technique will be crucial in pushing the boundaries of LLM-based RAG and unlocking the full potential of Generative AI.

Questions?