

Retrieval-Augmented Generation for Large Language Models: A Survey

Yunfan Gao^a, Yun Xiong^b, Xinyu Gao^b, Kangxiang Jia^b, Jinliu Pan^b, Yuxi Bi^c, Yi Dai^a, Jiawei Sun^a, Meng Wang^c, and Haofen Wang^{a,c}

^aShanghai Research Institute for Intelligent Autonomous Systems, Tongji University

^bShanghai Key Laboratory of Data Science, School of Computer Science, Fudan University

^cCollege of Design and Innovation, Tongji University

24.09.20
발제자: 김태균

Introduction

LLM의 뛰어난 성과에도 hallucination, outdated knowledge, non-transparent, untraceable reasoning process 등의 문제가 있음

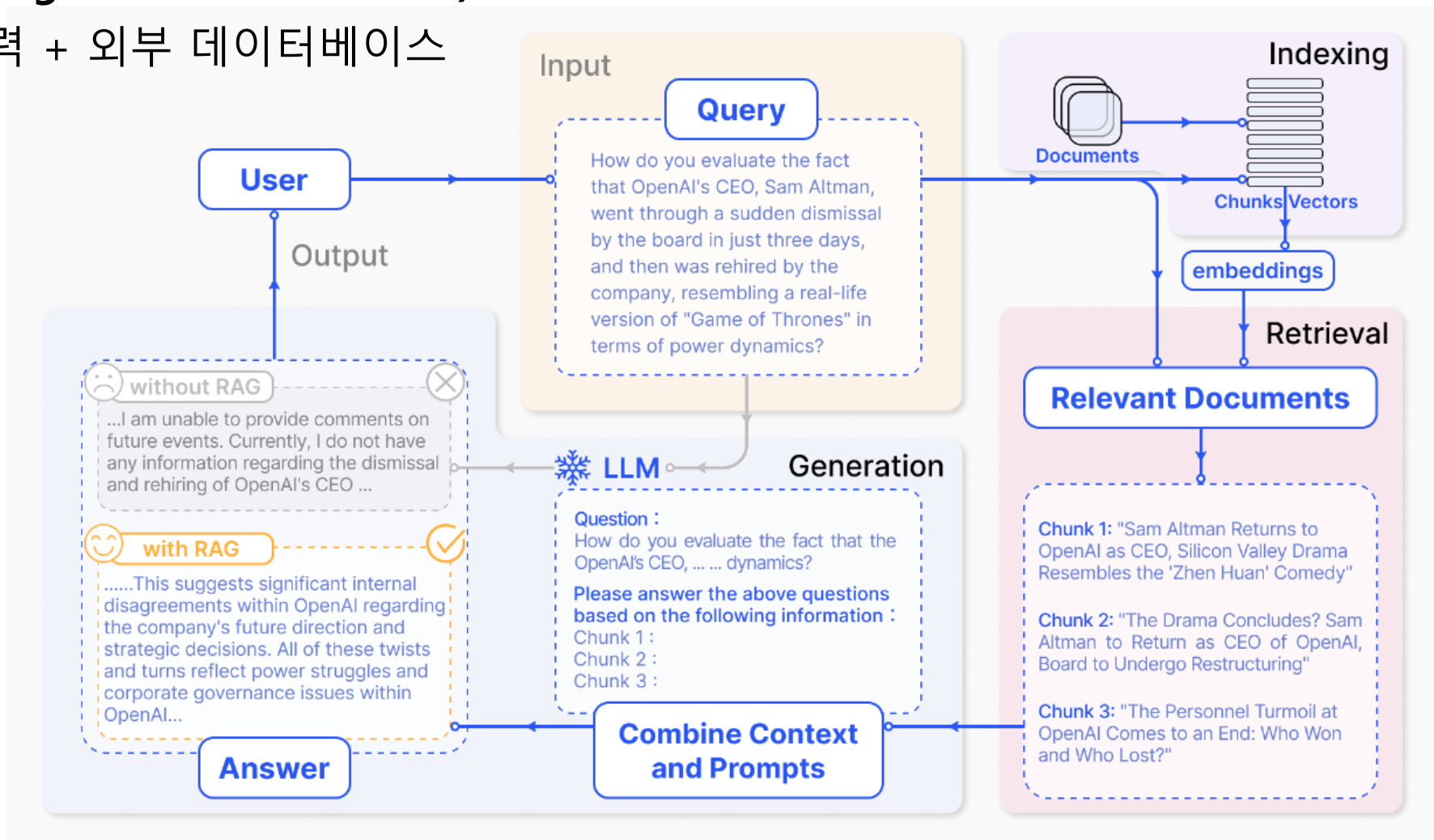
“Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks”(Patrick Lewis 등, 2020)에서 외부 지식을 활용하여 LLM을 보완하는 RAG를 처음 소개

RAG의 개념과 연구 동향

Overview of RAG

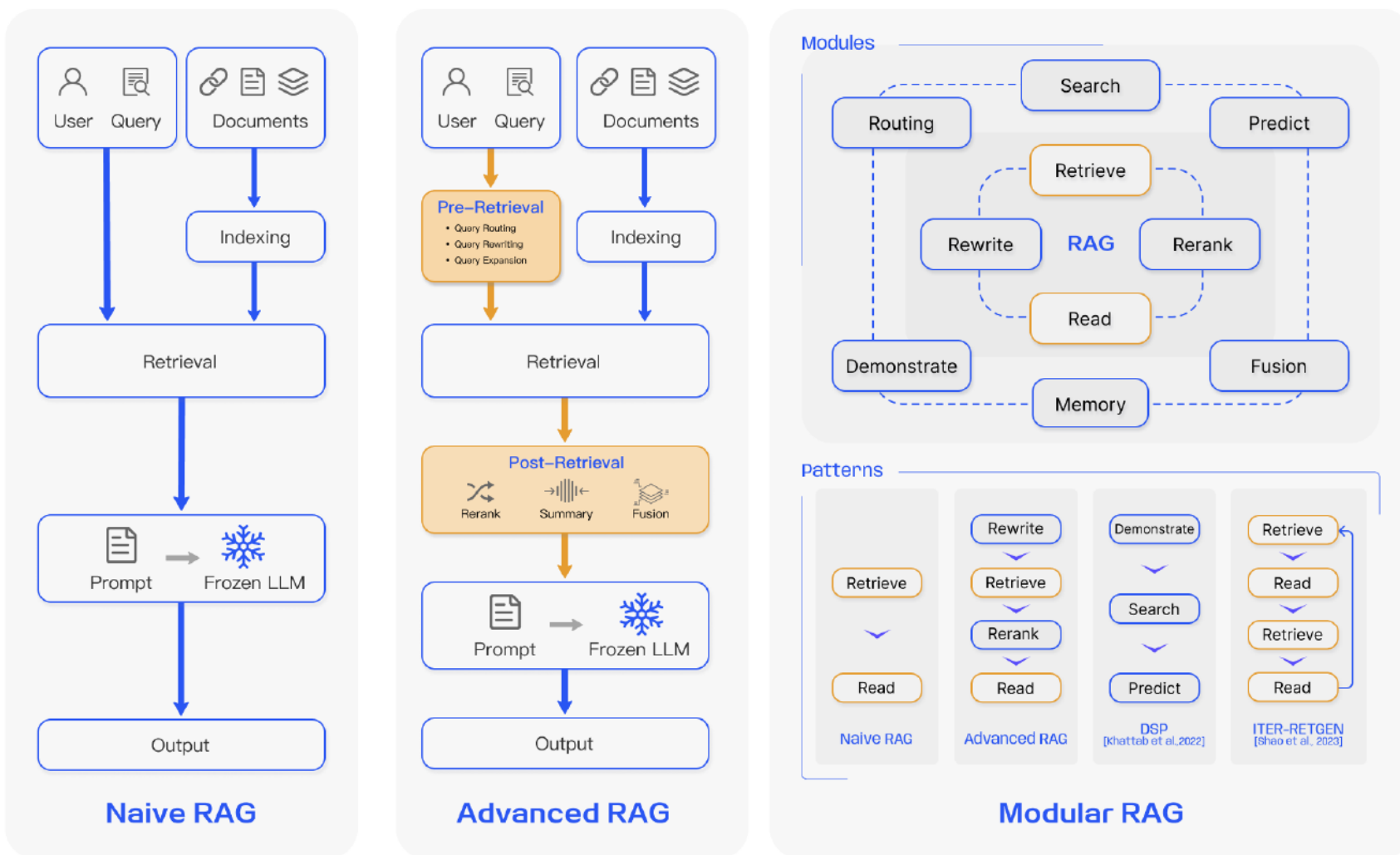
RAG(Retrieval-Augmented Generation)

: LLM의 생성 능력 + 외부 데이터베이스



Overview of RAG

Three paradigms of RAG

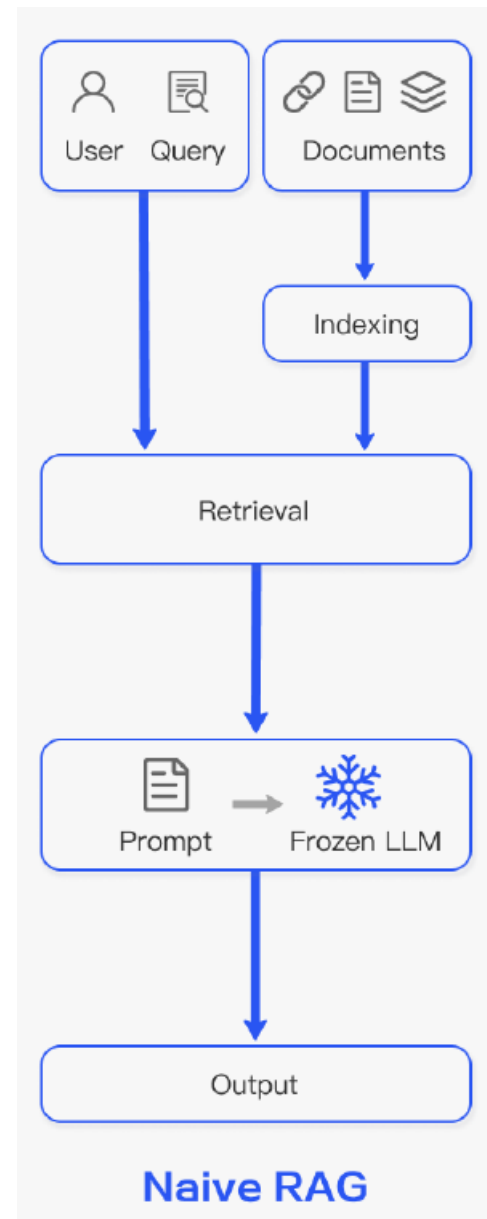


Overview of RAG

Naive RAG

Indexing, Retrieval, Generation의 전통적인 프로세스

1. 다양한 형식의 데이터가 텍스트 청크로 분할되어 벡터 DB에 저장
2. 벡터로 변환한 사용자 쿼리와 유사한 청크를 검색
3. 선택된 청크와 쿼리를 통합하여 LLM을 통해 응답 생성



Overview of RAG

Naive RAG

Indexing, Retrieval, Generation의 전통적인 프로세스

- Retrieval Challenges

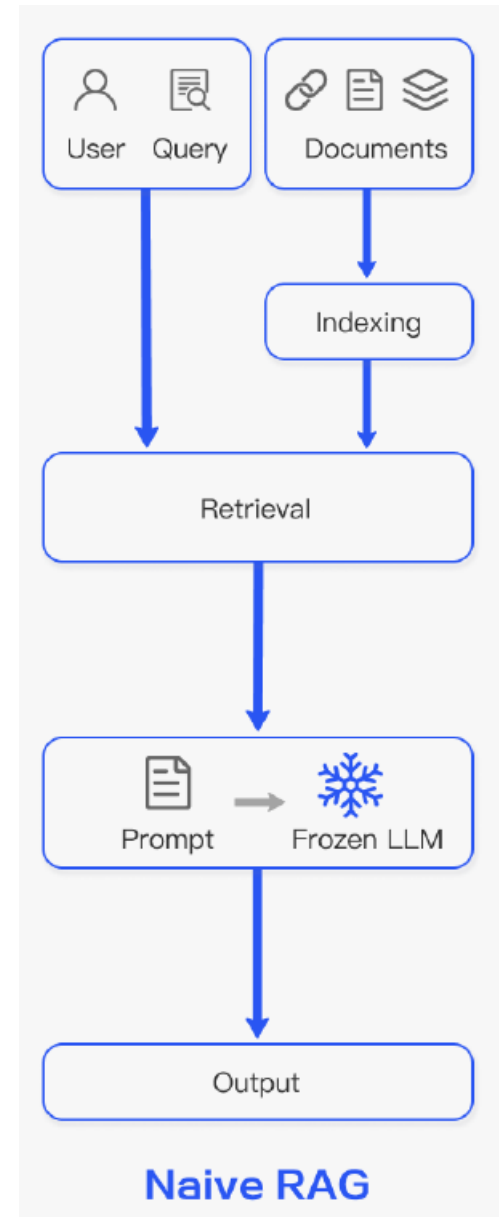
Selection of irrelevant chunks / Missing of crucial information

- Generation Difficulties

Hallucination

- Augmentation Hurdles

Incoherent outputs / Repetitive responses



Overview of RAG

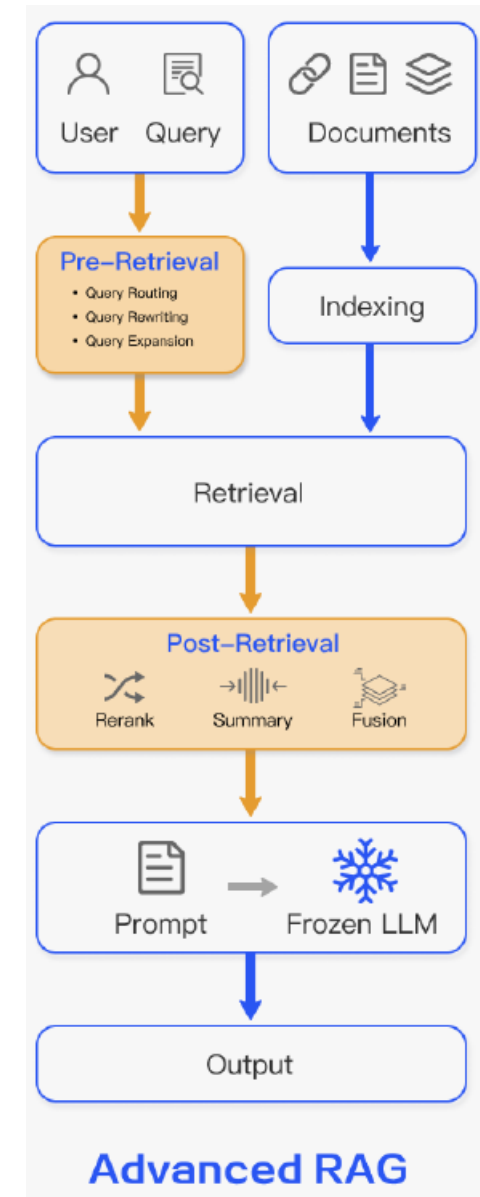
Advanced RAG

Naive RAG + Pre-Retrieval & Post-Retrieval

- Pre-Retrieval
 - Indexing optimization : sliding window, data granularity, adding metadata, ...
 - Query optimization : query rewriting, query transformation, query expansion, ...

- Post-Retrieval

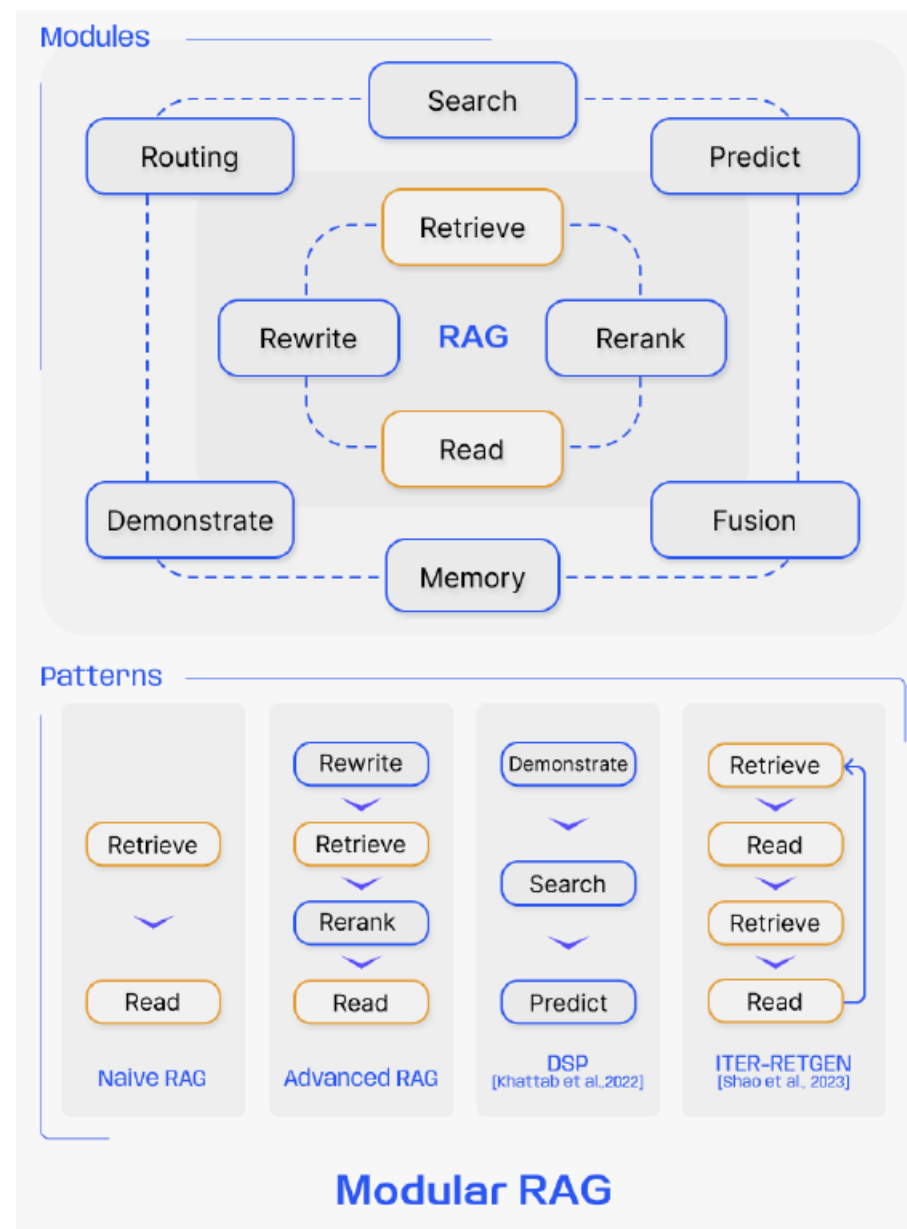
Rerank chunks / Context compressing



Overview of RAG

Modular RAG

모듈을 사용하여 유연성과 다양한 작업으로의 확장성 증가



Overview of RAG

RAG vs Fine-tuning

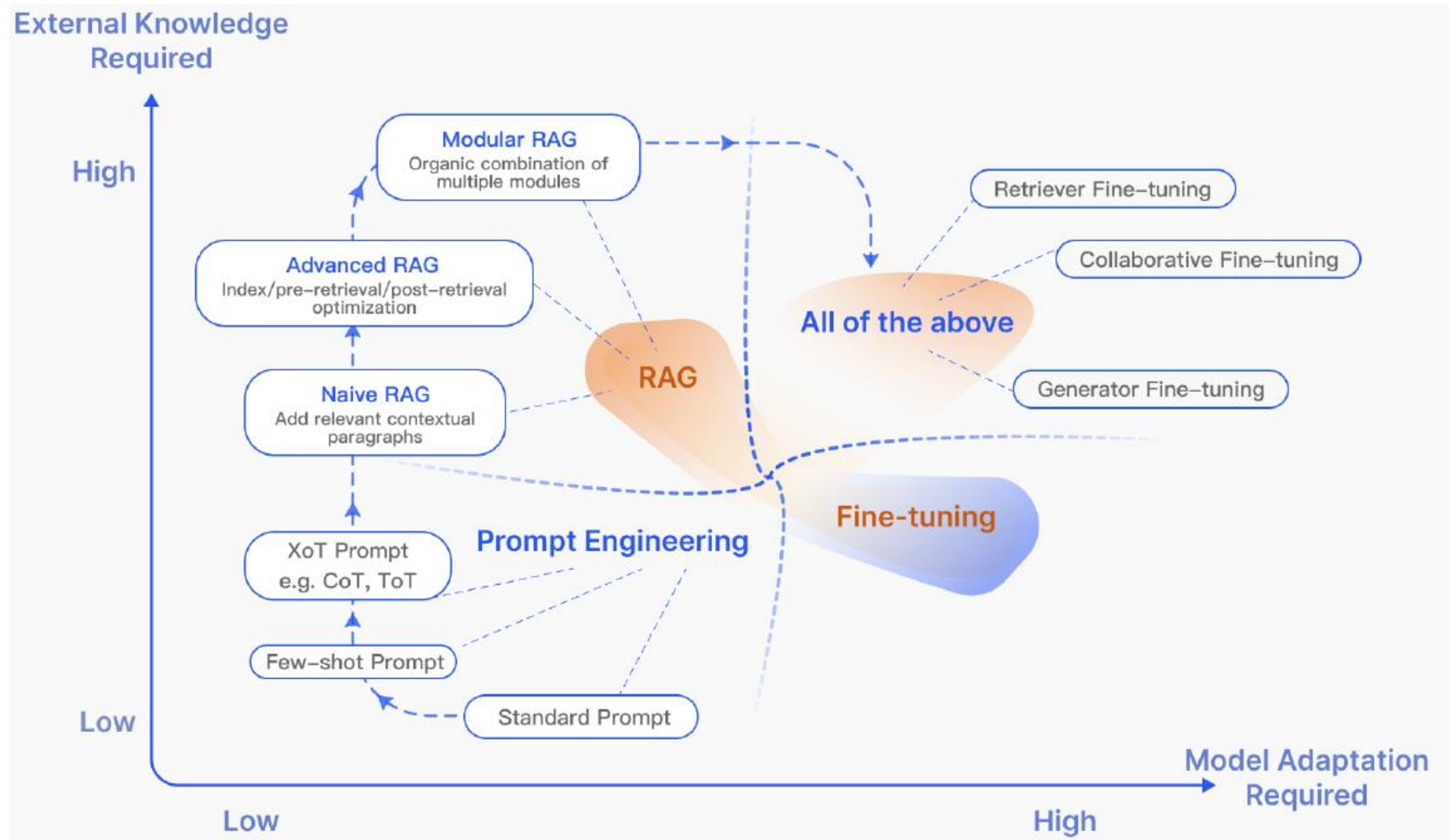
- **RAG**

- 최신 정보 반영 (동적 환경)
- 높은 검색 시간, 윤리적 고려사항

- **Fine-tuning**

- customization
- 업데이트를 하기 위한 훈련 (정적 환경)
- 상당한 컴퓨팅 자원 필요
- 새로운 데이터나 지식 처리 어려움

Overview of RAG



Retrieval

Key issues

- A. Retrieval Source
- B. Indexing Optimization
- C. Query Optimization
- D. Embedding

Retrieval

A. Retrieval Source

- **Data Structure**

- unstructured data(text)
- semi-structured data(PDF),
- structured data(Knowledge Graph)
- LLM-generated

- **Retrieval Granularity**

- text : token, phrase, sentence, proposition, chunk, document
- knowledge graph : entity, triplet, sub-graph

Retrieval

B. Indexing Optimization

- Chunking Strategy : 가장 일반적인 방법으로 문서를 고정된 토큰 수(e.g. 100, 256, 512)로 나눔
- Metadata Attachments : 청크에 메타데이터(페이지 번호, 파일 이름, 저자 등) 정보를 추가
→ 검색 시 필터링이 가능하고 오래된 정보를 피할 수 있음
- Structural Index : 문서에 계층적 구조를 설정 → 신속한 데이터 검색 및 처리 가능

Retrieval

C. Query Optimization

- **Query Expansion** : 단일 쿼리를 다중 쿼리로 확장

e.g. multi-query, sub-query, chain-of-verification(CoVe)

- **Query Transformation** : 변환된 쿼리를 사용하여 청크를 검색

e.g. query rewrite, prompt engineering

- **Query Routing** : 다양한 쿼리에 따라 적합한 RAG 파이프라인으로 라우팅하여 검색 범위를 좁힘

e.g. metadata router/filter, semantic router

Retrieval

D. Embedding

- **Sparse Retriever** : 텍스트 \rightarrow 고차원 희소 벡터
- **Dense Retriever** : 텍스트 \rightarrow 저차원 밀집 벡터
- **Mix/hybrid Retrieval** : sparse와 dense retriever를 모두 사용
- **Fine-tuning Embedding Model** : 특정 도메인에서 성능 향상과 retriever와 generator의 정렬

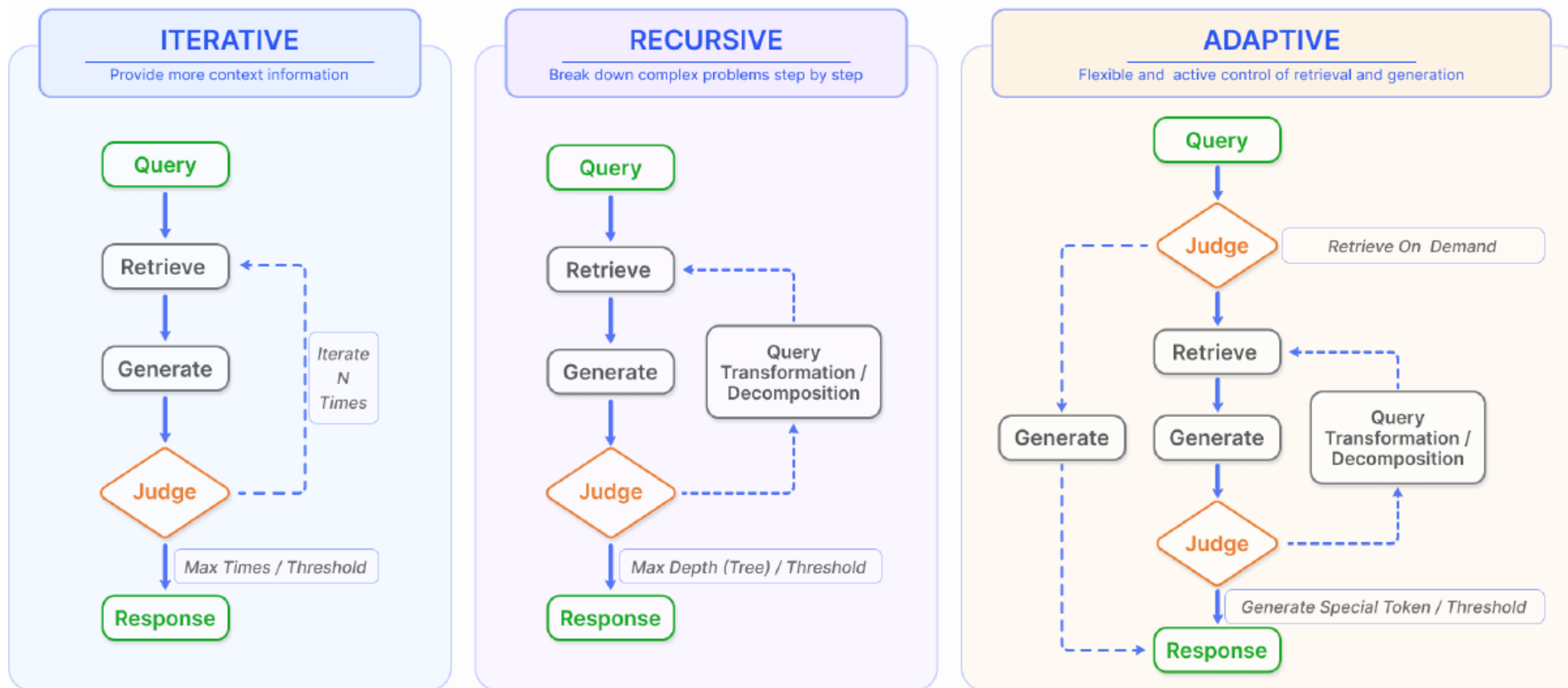
Generation

- **Context Curation**
 - Reranking
 - Context Selection/Compression
- **LLM Fine-tuning**

⇒ LLM의 generation 성능 향상

Augmentation Process

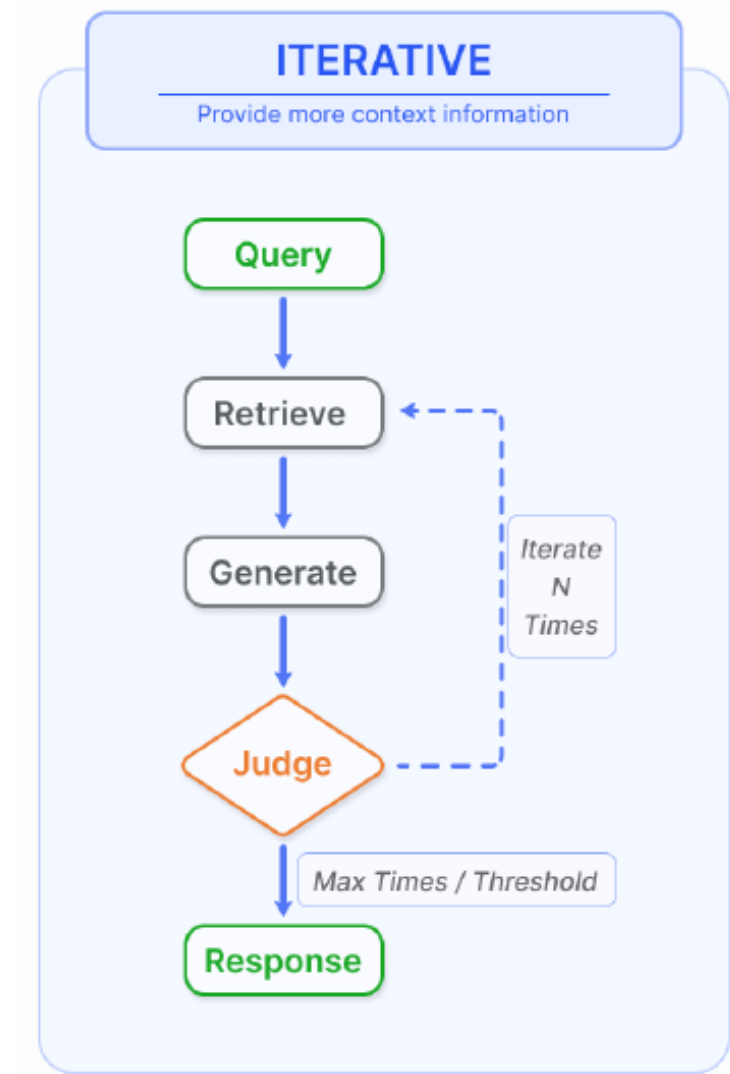
Three types of retrieval augmentation processes



Augmentation Process

1. Iterative Retrieval

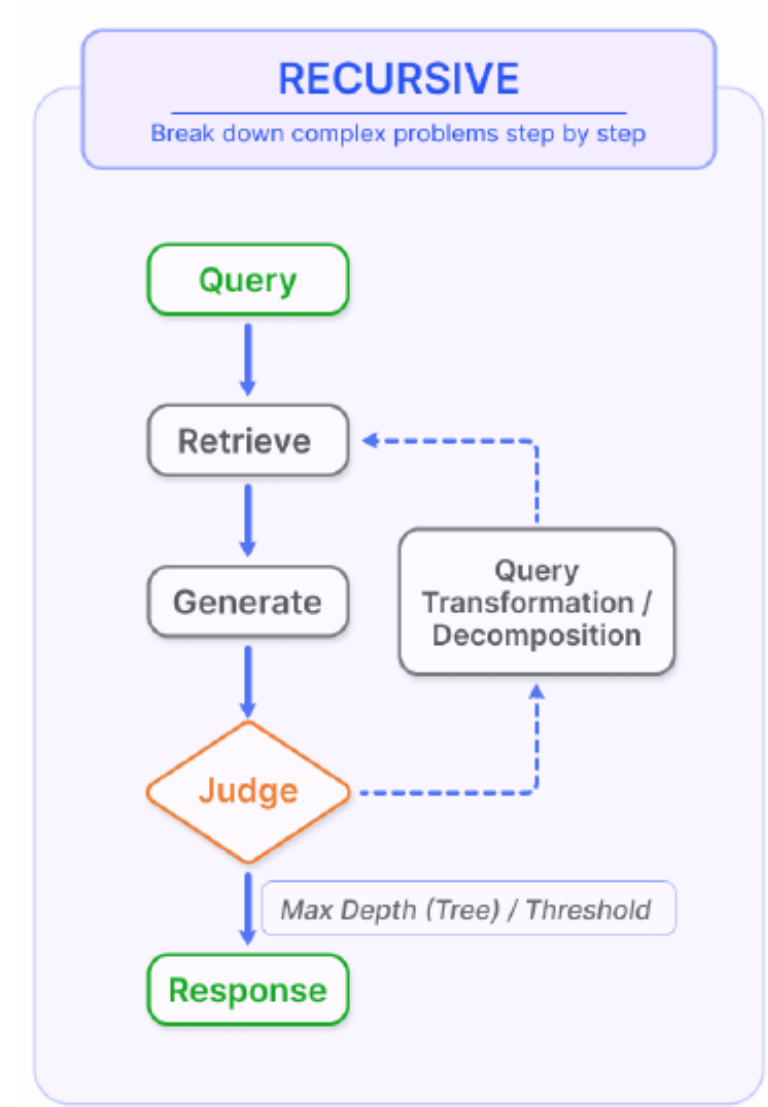
초기 쿼리와 현재까지 생성된 텍스트를 바탕으로 반복적으로 검색하는 방법



Augmentation Process

2. Recursive Retrieval

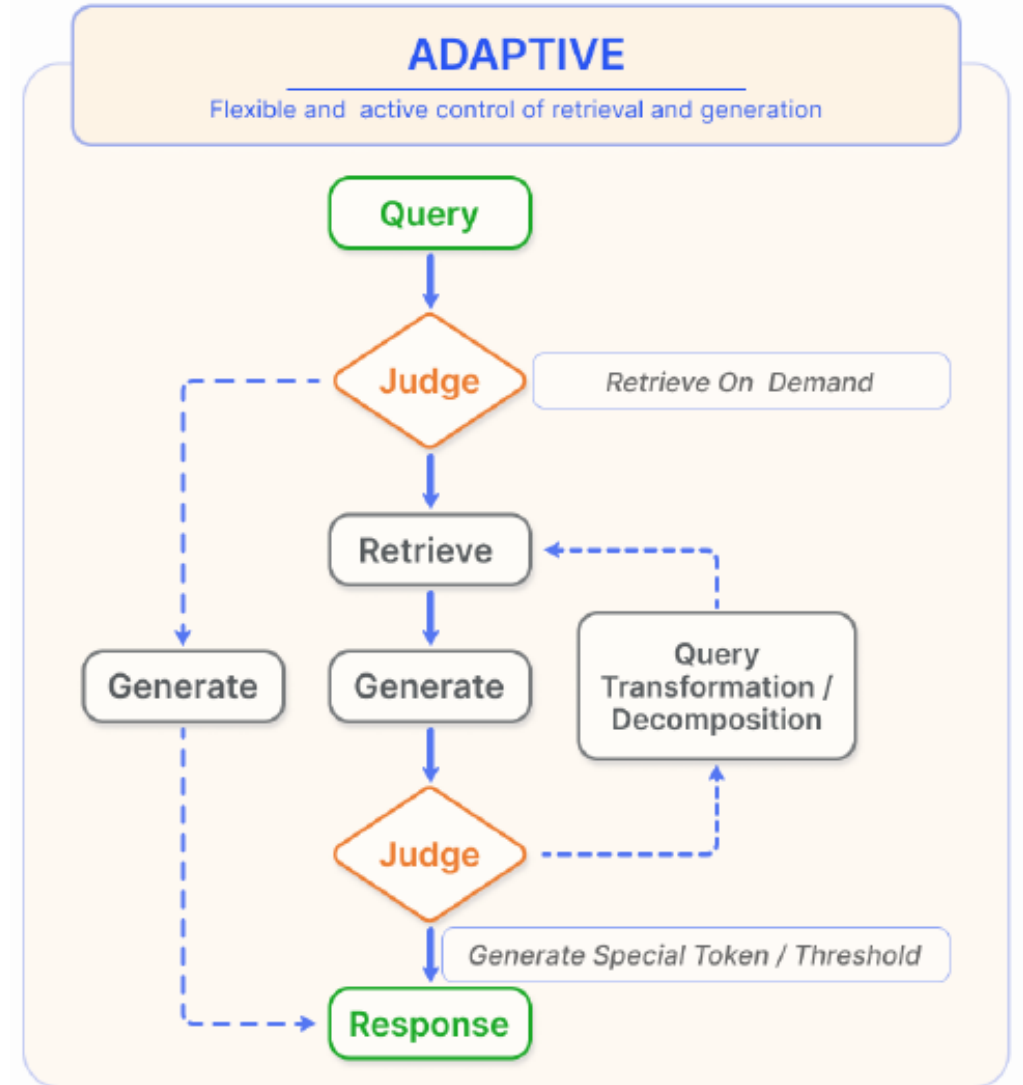
이전 검색을 기반으로 검색 쿼리를 반복적으로 refine하여 검색하는 방법



Augmentation Process

3. Adaptive Retrieval

LLM이 검색 시점과 내용을 능동적으로 결정하여
정보의 효율성과 관련성을 향상시키는 방법



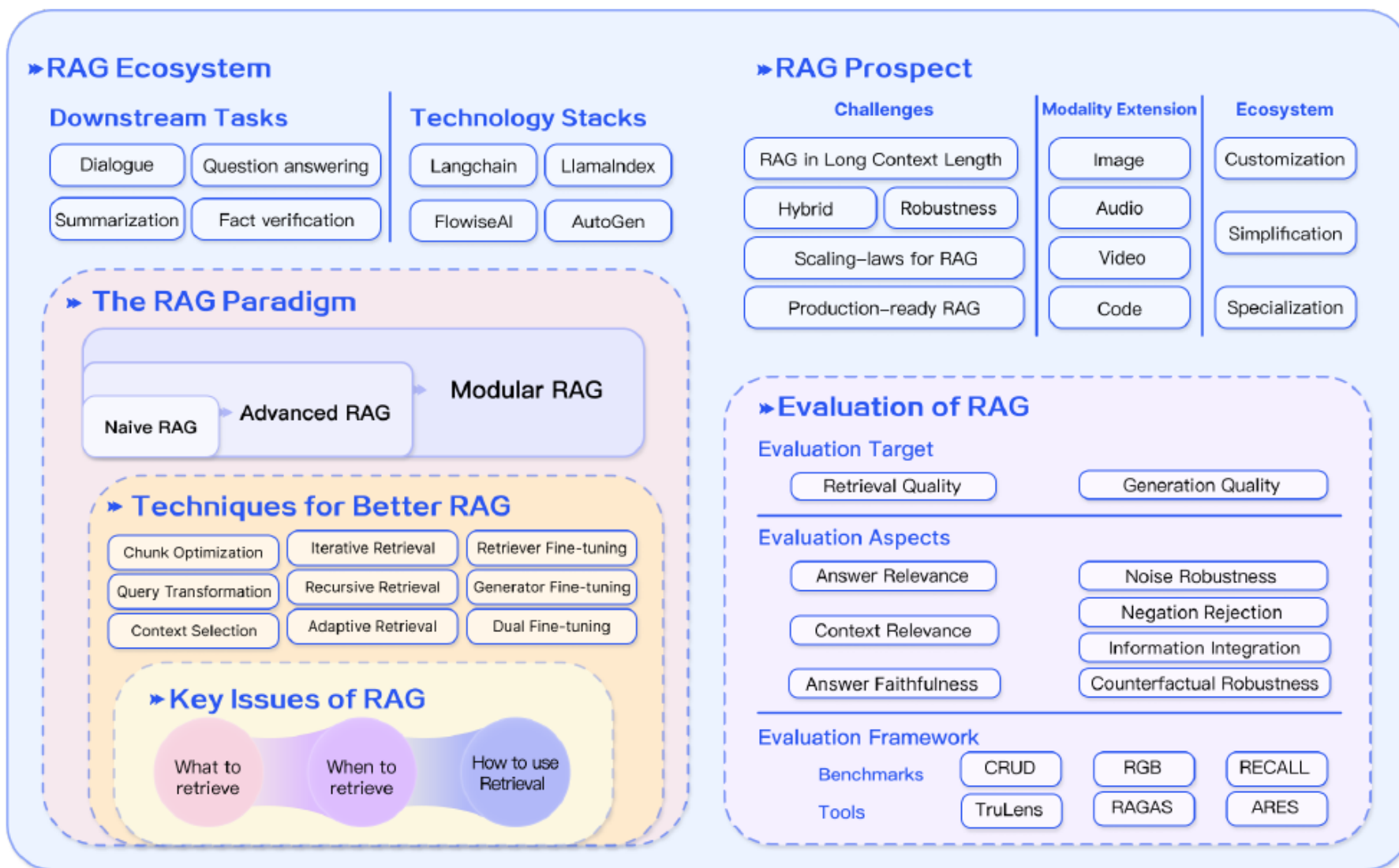
Task and Evaluation

- **Downstream Task** : 질의응답(QA), 정보 추출(IE), 대화 생성, 코드 검색 등
- **Evaluation**
 - Retrieval Quality : context relevance, noise robustness
 - Generation Quality : answer faithfulness, answer relevance, negative rejection, information integration, counterfactual robustness
 - Benchmarks and Tools : RGB, RECALL, CRUD, RAGAS, ARES, TruLens

Future Prospects

- **Long Context** : super-long context에서의 RAG
- **RAG Robustness** : 노이즈나 잘못된 정보에 대한 저항성
- **Hybrid Approaches** : RAG와 fine-tuning의 결합
- **Multi-modal RAG** : 이미지, 영상, 코드 등 다양한 모달 데이터로의 확장

Conclusion



Retrieval

A. Retrieval Source

- **Data Structure**

- unstructured data(text)
- semi-structured data(PDF),
- structured data(Knowledge Graph)
- LLM-generated

- **Retrieval Granularity**

- text : token, phrase, sentence, proposition, chunk, document
- knowledge graph : entity, triplet, sub-graph

Dense X Retrieval: What Retrieval Granularity Should We Use?

Tong Chen♣* Hongwei Wang◇ Sihao Chen♡ Wenhao Yu◇

Kaixin Ma◇ Xinran Zhao♠ Hongming Zhang◇ Dong Yu◇

♣University of Washington ◇Tencent AI Lab

♡University of Pennsylvania ♠Carnegie Mellon University


Background

일반적으로 dense retrieval을 사용할 때 document, passage, sentence 단위로 검색
검색 단위의 선택이 retrieval과 downstream task의 성능에 영향을 주는 것을 발견

새로운 검색 단위 “**proposition**”을 도입하여
retrieval과 downstream task에서 성능을 분석하고자 함

Proposition

Question: What is the angle of the Tower of Pisa?	
Passage Retrieval	Prior to restoration work performed between 1990 and 2001, the tower leaned at an angle of 5.5 degrees, but the tower now leans at about 3.99 degrees. This means the top of the Leaning Tower of Pisa is displaced horizontally 3.9 meters (12 ft 10 in) from the center.
Sentence Retrieval	Prior to restoration work performed between 1990 and 2001, the tower leaned at an angle of 5.5 degrees, but the tower now leans at about 3.99 degrees.
Proposition Retrieval	The Leaning Tower of Pisa now leans at about 3.99 degrees.

 : 정답에 대응되는 부분



Passage : coarser retrieval unit
종종 쿼리와 관련 없는 세부사항 포함.
e.g. 피사의 사탑의 restoration period



Sentence : finer-grained unit
하나의 문장에 여러 정보가 존재할 수 있으며,
문장 하나만으로는 완전한 의미 전달을 못할 수 있음
e.g. 피사의 사탑을 의미하는 'the tower'를 이해하기
위해 맥락 정보가 필요


Proposition

Question: What is the angle of the Tower of Pisa?	
Passage Retrieval	Prior to restoration work performed between 1990 and 2001, the tower leaned at an angle of 5.5 degrees, but the tower now leans at about 3.99 degrees. This means the top of the Leaning Tower of Pisa is displaced horizontally 3.9 meters (12 ft 10 in) from the center.
Sentence Retrieval	Prior to restoration work performed between 1990 and 2001, the tower leaned at an angle of 5.5 degrees, but the tower now leans at about 3.99 degrees.
Proposition Retrieval	The Leaning Tower of Pisa now leans at about 3.99 degrees.



Proposition

- 텍스트 내의 고유한 의미 단위
- 더 이상 별개의 명제로 분할할 수 없는 최소 단위
- 의미를 해석하는 데 필요한 모든 맥락을 포함

 : 정답에 대응되는 부분

Experimental Settings

FACTOIDWIKI

Propositionizer라는 텍스트 생성 모델을 사용하여 영어 위키피디아 페이지에서 proposition으로 parsing한 데이터셋

Propositionizer

- GPT-4에 proposition의 정의와 1개의 예시를 입력으로 넣어 paragraph-to-propositions pair 생성
- 생성된 pair를 Flan-T5-large 모델에 파인튜닝

Experimental Settings

Open-Domain QA Datasets

: Natural Questions(NQ), TriviaQA(TQA), Web Questions(WebQ), SQuAD, Entity Questions(EQ)

Dense Retrieval Models

: SimCSE, Contriever, DPR, ANCE, TAS-B, GTR

Passage Retrieval Evaluation

쿼리와 해당 passage의 모든 sentence, proposition 간 최대 유사도 점수를 기반으로 k개의 고유한 passage를 반환하여 Recall@k를 평가 지표로 사용

Downstream QA Evaluation

- retrieval model : retrieve-then-read를 사용하여 상위 l개의 단어를 검색 (l=100 or 500)
- reader model : UnifiedQA-v2에 입력하여 생성된 답을 EM으로 평가 (EM@l)

Experiment

A Prior to restoration work performed between 1990 and 2001, the tower leaned at an angle of 5.5 degrees, // but the tower now leans at about 3.99 degrees. // This means the top of the *Leaning Tower of Pisa* is displaced horizontally 3.9 meters (12 ft 10 in) from the center.

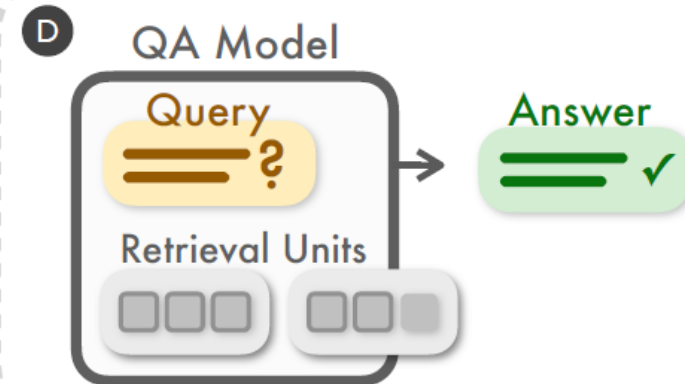
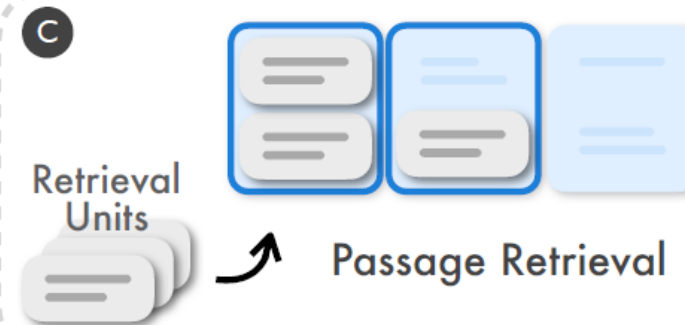
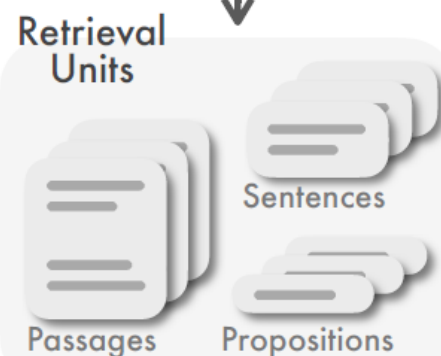
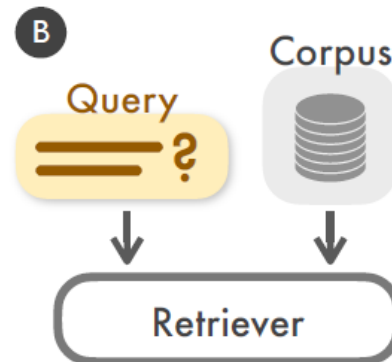
1. Prior to restoration work performed between 1990 and 2001, the *Leaning Tower of Pisa* leaned at an angle of 5.5 degrees.

2. The *Leaning Tower of Pisa* now leans at about 3.99 degrees.

3. The top of the *Leaning Tower of Pisa* is displaced horizontally 3.9 meters (12 ft 10 in) from the center.



Proposition-izer



Results: Passage Retrieval

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
Supervised Dense Retrievers													
DPR	Passage	66.0	78.0	71.6	80.2	62.9	74.9	38.3	53.9	47.5	60.4	57.3	69.5
	Sentence	66.0	78.0	71.8	80.5	64.1	74.4	40.3	55.9	53.7	66.0	59.2	71.0
	Proposition	65.4	77.7	70.7	79.6	62.8	75.1	41.4	57.2	59.4	71.3	59.9	72.2
ANCE	Passage	70.7	81.4	73.9	81.4	65.7	77.2	43.3	58.6	57.0	69.1	62.1	73.5
	Sentence	70.3	81.6	73.9	81.5	65.2	77.4	45.8	60.7	61.4	72.8	63.3	74.8
	Proposition	69.9	81.1	72.8	80.6	65.1	77.1	46.2	61.9	66.7	76.6	64.1	75.5
TAS-B	Passage	64.2	77.9	70.4	79.3	65.1	77.0	54.3	69.2	72.2	81.3	65.2	76.9
	Sentence	64.0	78.4	71.4	80.2	63.9	76.7	58.9	72.3	72.7	82.0	66.2	77.9
	Proposition	63.8	78.6	71.4	80.0	63.8	76.8	59.8	73.4	75.1	83.3	66.8	78.4
GTR	Passage	66.3	78.4	70.1	79.4	63.3	76.5	54.4	68.1	71.7	80.5	65.2	76.6
	Sentence	66.4	79.4	71.6	80.9	62.2	76.8	60.9	73.4	72.5	81.3	66.7	78.4
	Proposition	66.5	79.6	72.2	80.9	63.2	77.4	63.3	75.0	74.9	83.0	68.0	79.2

Results: Open-Domain QA

Retriever	Granularity	NQ		TQA		WebQ		SQuAD		EQ		Avg.	
		EM		EM		EM		EM		EM		EM	
		@ 100	@ 500	@ 100	@ 500	@ 100	@ 500	@ 100	@ 500	@ 100	@ 500	@ 100	@ 500
Unsupervised Dense Retrievers													
SimCSE	Passage	8.1	16.3	22.6	33.7	7.7	14.9	9.8	17.8	10.9	17.5	11.8	20.0
	Sentence	10.1	18.0	27.2	37.2	9.6	15.6	17.3	24.8	13.0	19.8	15.4	23.1
	Proposition	12.7	20.2	28.4	37.7	11.2	17.2	18.0	25.1	18.3	25.0	17.7	25.0
Contriever	Passage	11.1	22.4	25.7	41.4	6.8	14.9	15.6	27.7	10.9	21.5	14.0	25.6
	Sentence	13.8	23.9	30.5	44.2	9.1	17.2	22.6	32.8	12.2	22.2	17.6	28.1
	Proposition	16.5	26.1	37.7	48.7	13.3	19.9	25.6	34.4	16.1	27.3	21.8	31.3
Supervised Dense Retrievers													
DPR	Passage	<u>24.8</u>	<u>36.1</u>	<u>40.3</u>	<u>51.0</u>	<u>14.0</u>	<u>22.2</u>	<u>12.4</u>	<u>21.7</u>	18.6	25.9	22.0	31.4
	Sentence	<u>27.6</u>	<u>35.9</u>	<u>44.6</u>	<u>52.8</u>	<u>16.3</u>	<u>23.7</u>	<u>18.6</u>	<u>26.1</u>	21.8	28.2	25.8	33.3
	Proposition	<u>28.3</u>	<u>34.3</u>	<u>45.7</u>	<u>51.9</u>	<u>19.0</u>	<u>23.8</u>	<u>19.8</u>	<u>26.3</u>	26.3	31.9	27.8	33.6
ANCE	Passage	<u>27.1</u>	<u>38.3</u>	<u>43.1</u>	<u>53.1</u>	<u>15.2</u>	<u>23.0</u>	<u>15.3</u>	<u>26.0</u>	23.4	31.1	24.8	34.3
	Sentence	<u>30.1</u>	<u>37.3</u>	<u>47.0</u>	<u>54.7</u>	<u>16.6</u>	<u>23.8</u>	<u>22.9</u>	<u>30.5</u>	25.9	32.0	28.5	35.7
	Proposition	<u>29.8</u>	<u>37.0</u>	<u>47.4</u>	<u>53.5</u>	<u>19.3</u>	<u>24.1</u>	<u>22.9</u>	<u>30.1</u>	29.1	33.7	29.7	35.7
TAS-B	Passage	21.1	33.9	39.3	50.5	13.1	20.7	23.9	34.6	30.9	37.3	25.7	35.4
	Sentence	24.6	33.9	43.6	52.3	14.4	21.4	33.8	40.5	31.4	36.1	29.6	36.8
	Proposition	26.6	34.0	44.9	51.8	18.1	23.7	34.2	38.9	34.2	37.8	31.6	37.2
GTR	Passage	<u>23.4</u>	<u>34.5</u>	38.7	49.3	13.1	20.1	23.9	33.8	31.3	36.7	26.1	34.9
	Sentence	<u>26.8</u>	<u>35.1</u>	43.9	52.2	15.9	21.6	35.6	41.3	31.3	35.1	30.7	37.1
	Proposition	<u>29.5</u>	<u>34.4</u>	45.9	52.6	18.7	23.8	37.0	40.4	34.1	37.1	33.0	37.7

Error Case Study

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...
Q2: The atomic number of indium which belongs to 5th period is?		
Title: Period 5 element ✗ The periodic table is laid out in rows to illustrate recurring (periodic) trends in the chemical behaviour of the elements as their atomic number increases: ...	Title: Period 5 element ✓ Indium is a chemical element with the symbol In and <u>atomic number 49</u> .	Title: Period 5 element ✓ Indium is a chemical element with the symbol In and [Indium has a] <u>atomic number 49</u> . This rare, very soft, malleable ...
Q3: What is the function of the pericardial sac?		
Title: Pericardium ✓ The pericardium, also called pericardial sac ... It separates the heart from interference of other structures, protects <u>it against infection and blunt trauma, and lubricates the heart's movements</u> .	Title: Pericardium ✗ The pericardium, also called pericardial sac, is a double-walled sac containing the heart and the roots of the great vessels.	Title: Cardiac muscle ✓ On the outer aspect of the myocardium is the <u>epicardium which forms part of the pericardial sac that surrounds, protects, and lubricates the heart</u> .
Q4: What is the main character's name in layer cake?		
Title: Layer Cake (film) ✓ ... The film's plot revolves around a London-based criminal, played by Daniel Craig, ... Craig's character is unnamed in the film and is listed in the credits as " <u>XXXX</u> ".	Title: Angelic Layer ✗ The primary protagonist is Misaki Suzuhara.	Title: Plot twist ✗ Sometimes the audience may discover that the true identity of a character is , in fact, unknown [in Layer Cake] , as in Layer Cake or the eponymous assassins in V for Vendetta and The Day of the Jackal.

Conclusion

dense retrieval와 open-domain QA task에서 'proposition' 검색 단위가 다른 검색 단위 'passage', 'sentence'에서의 성능을 능가

소감

특정 주제에 대한 전반적인 지식을 얻고 관련 연구에 대해 처음 입문할 때
survey 논문 읽기

어떤 문서를 retrieve 하는지도 중요하지만 어떻게 retrieve 하는지도 중요하다

Open Questions

- Error case와 같이 쿼리에 대응되는 정보가 multi-hop으로 분리되어 있는 경우 올바른 답변을 얻기 위해 어떻게 retrieve해야 할까?
- sparse retriever도 proposition 단위로 검색할 때 성능이 좋을까?

QnA