Reasoning-Intensive Information Retrieval

2025 하계세미나

장영준

대 Reasoning 시대

검색에도 Reasoning이 ???

First Last

Backend Developer

WORK EXPERIENCE

Resume Worded, London, United Kingdom
Education technology startup with 50+ employees and $100m+ annual revenue

Backend Developer — 08/2021 – Present

- Developed tools to automate deployment processes using Jenkins Pipelines and Kubernetes, reducing deployment time from 24 hours to 45 minutes.
- Collaborated with a team of 10+ developers to develop an e-commerce platform that could scale up to 1M visitors.
- Designed APIs to support the development of 20+ new web features, course management systems (CMS), and Student Information Systems (SIS).
- Created a microservices-based architecture with Node.js, Kubernetes, Docker, and MySQL for an online education platform that serves 500K daily users.

Polyhire, London, United Kingdom
NYSE-listed recruitment and employer branding company

UI Developer — 10/2019 – 07/2021

- Designed 20+ interactive User Interface (UI) applications using HTML5, CSS3, Angular 4/2, NodeJS, jQuery, and JSON.
- Overhauled the pop-up screens and dropdown menus on 50+ web pages, making the UI 100% functional.
- Developed 10+ web applications using modern JavaScript ES6 features and frameworks.
- Created server-side JavaScript codes to build 120+ web forms and simulate processes for web applications, page navigation, and form validation.

Growthsi, London, United Kingdom & Barcelona, Spain
Digital marketing agency focusing on search engine marketing (SEM)

Web Content Editor — 11/2018 – 09/2019

- Contributed to developing a next-generation website for an international company conducting business in 30+ countries.
- Improved online search-ranking accuracy by optimizing 20+ key content pages for SEO purposes.
- Edited web content pieces supporting Growthsi marketing initiatives by ensuring 97% consistency with the brand's voice and tone.
- Enhanced the company's search rankings through onsite optimization and content development, increasing website traffic by 75%.

PREVIOUS EXPERIENCE

IT Support Engineer, ABC Company, London, UK — 06/2017 – 10/2018
Programmer, XYZ Company, New York, USA — 01/2016 – 05/2017
UI Designer(Internship), ABC, New York, USA — 07/2014 – 12/2015

CONTACT

- Worcester, United Kingdom
- +44 1234567890
- first.last@gmail.com

SKILLS

Hard Skills:
- Debugging
- Server Handling
- Visual Editing
- Data Modeling
- Wireframing
- Troubleshooting

Techniques:
- API Integration
- Technical Analysis
- UI/UX Design

Tools and Software:
- SQL
- Java
- C++
- HTML

Languages:
- English (Native)
- Romanian (Native)
- Spanish (Conversational)

EDUCATION

University of New York
Associate of Science
Computer Science
New York City, New York
10/2011 - 06/2014

OTHER

- Certified Java Programmer (2021)
- Certificate Program In Text And Web Analytics (2019)

Table Of Contents

BRIGHT: A REALISTIC AND CHALLENGING BENCHMARK FOR REASONING-INTENSIVE RETRIEVAL

Hongjin Su[*h] **Howard Yen**[*p] **Mengzhou Xia**[*p] **Weijia Shi**[w] **Niklas Muennighoff**[s]
Han-yu Wang[h] **Haisu Liu**[h] **Quan Shi**[p] **Zachary S. Siegel**[p] **Michael Tang**[p]
Ruoxi Sun[g] **Jinsung Yoon**[g] **Sercan Ö. Arık**[g] **Danqi Chen**[p] **Tao Yu**[h]

[h] The University of Hong Kong [p] Princeton University [s] Stanford University
[w] University of Washington [g] Google Cloud AI Research

{hjsu,tyu}@cs.hku.hk {hyen,mengzhou,danqic}@cs.princeton.edu

ICLR 2025

1. BRIGHT

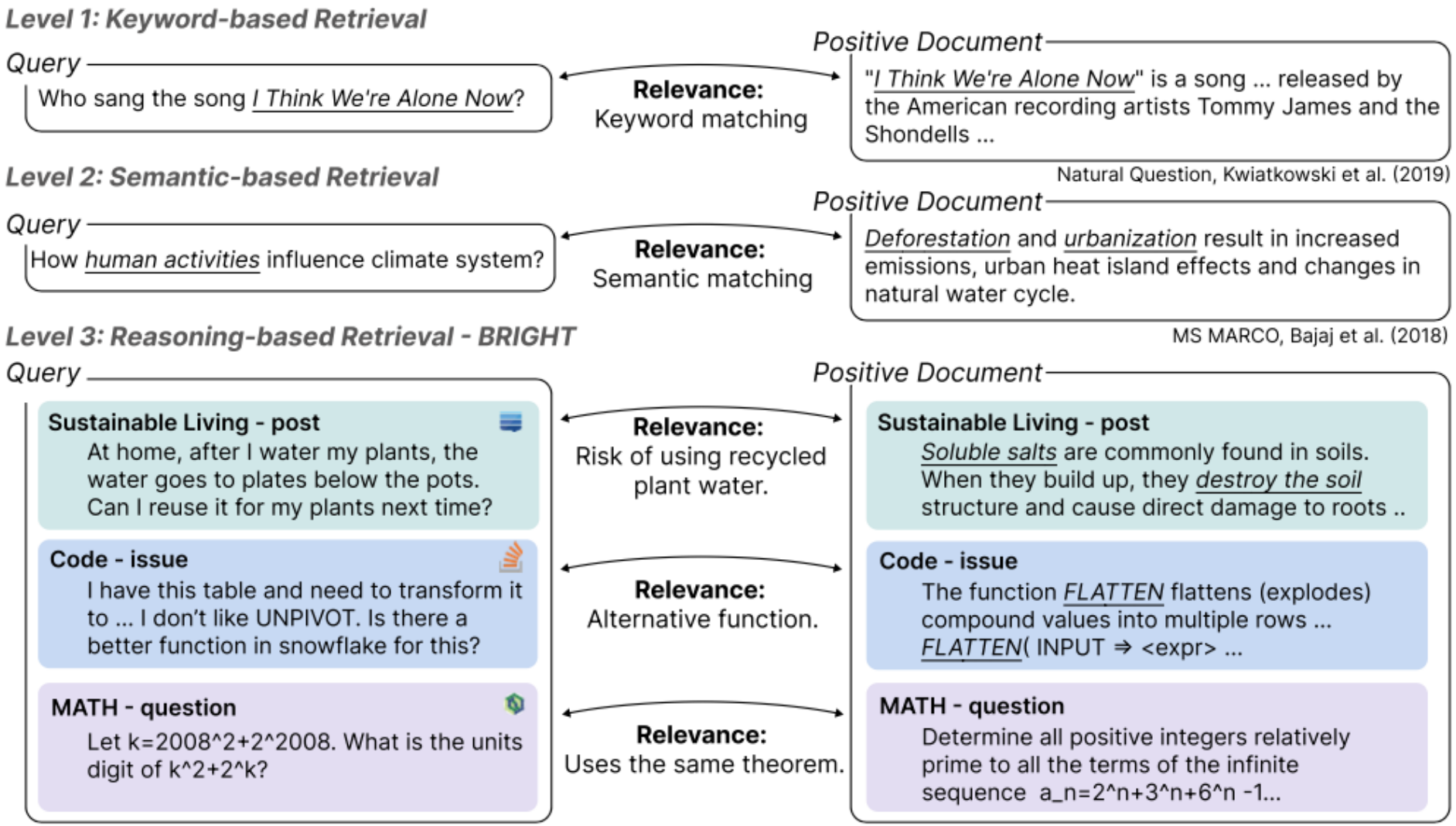
0. Overview

- Problem Statement
  - 기존 IR 벤치마크 (BEIR, MSMARCO, NQ 등)는 주로 <간단한 질의>-<문서> 연결만을 요구
  - 위와 같은 데이터셋은 lexical 혹은 semantic-based 매칭을 통해 답을 찾기 쉬움
  - 그러나 실제 복잡한 검색 (경제학, 수학, 프로그래밍 등)에서는 표면적 텍스트 매칭 뿐만 아니라, 질의를 이해하여 **추가적인 추론**까지 해야 하는 경우가 있음
- Contribution
  - 7개 StackExchange(지식 커뮤니티) 분야 + 5개 수학/코딩 분야 → 총 12개 sub dataset, 1,384 queries
  - 단순한 키워드·의미 매칭 이상의 추론 과정이 없으면 관련 문서를 찾기 어려운 예시들만 선별
  - 평가 결과, 최신 SOTA 검색 모델 (SFR, GritLM, Qwen 등) 조차 nDCG@10에서 낮은 점수(최고 24점대)를 기록

    ➔ 기존 검색 기술로는 해결이 쉽지 않은 난이도의 데이터셋

1. BRIGHT

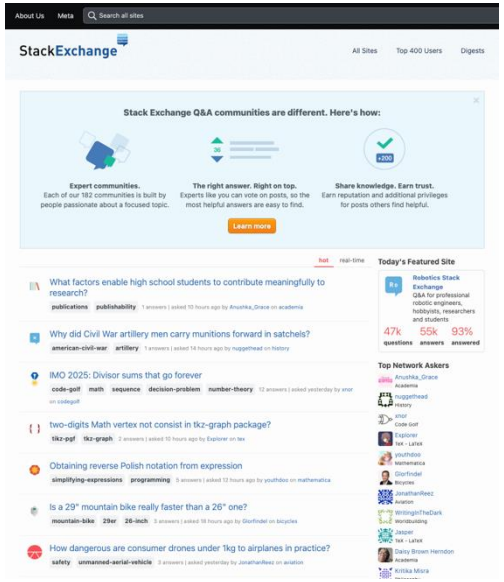
0. 요약


Level 1: Keyword-based Retrieval
Query
Who sang the song I Think We're Alone Now?
Relevance:
Keyword matching
Positive Document
"I Think We're Alone Now" is a song ... released by the American recording artists Tommy James and the Shondells ...
Natural Question, Kwiatkowski et al. (2019)
Level 2: Semantic-based Retrieval
Query
How human activities influence climate system?
Relevance:
Semantic matching
Positive Document
Deforestation and urbanization result in increased emissions, urban heat island effects and changes in natural water cycle.
MS MARCO, Bajaj et al. (2018)
Level 3: Reasoning-based Retrieval - BRIGHT
Query
Sustainable Living - post
At home, after I water my plants, the water goes to plates below the pots. Can I reuse it for my plants next time?
Code - issue
I have this table and need to transform it to ... I don't like UNPIVOT. Is there a better function in snowflake for this?
MATH - question
Let k=2008^2+2^2008. What is the units digit of k^2+2^k?
Relevance:
Risk of using recycled plant water.
Relevance:
Alternative function.
Relevance:
Uses the same theorem.
Positive Document
Sustainable Living - post
Soluble salts are commonly found in soils. When they build up, they destroy the soil structure and cause direct damage to roots ..
Code - issue
The function FLATTEN flattens (explodes) compound values into multiple rows ... FLATTEN(INPUT ⇒ <expr> ...
MATH - question
Determine all positive integers relatively prime to all the terms of the infinite sequence a_n=2^n+3^n+6^n -1...

1. 데이터셋 상세

Dataset	Total Number			Avg. Length		Source		Examples
	Q	$\mathcal{D}$	$\mathcal{D}^+$	Q	$\mathcal{D}$	Q	$\mathcal{D}$	
StackExchange								
Biology	103	57,359	3.6	115.2	83.6	StackExchange post	Web pages: article, tutorial, news, blog, report ...	Tab. 20
Earth Science	116	121,249	5.3	109.5	132.6			Tab. 21
Economics	103	50,220	8.0	181.5	120.2			Tab. 22
Psychology	101	52,835	7.3	149.6	118.2			Tab. 23
Robotics	101	61,961	5.5	818.9	121.0			Tab. 24
Stack Overflow	117	107,081	7.0	478.3	704.7			Tab. 25
Sustainable Living	108	60,792	5.6	148.5	107.9			Tab. 26
Coding								
LeetCode	142	413,932	1.8	497.5	482.6	Coding question	Coding Q&Sol	Tab. 27
Pony	112	7,894	22.5	102.6	98.3	Coding question	Syntax Doc	Tab. 28
Theorems								
AoPS	111	188,002	4.7	117.1	250.5	Math Olympiad Q	STEM Q&Sol	Tab. 29
TheoremQA-Q	194	188,002	3.2	93.4	250.5	Theorem-based Q	STEM Q&Sol	Tab. 30
TheoremQA-T	76	23,839	2.0	91.7	354.8	Theorem-based Q	Theorems	Tab. 31

1. 데이터셋 상세



1. StackExchange Dataset (7 domains)

   A. High-Quality Document선정

      I. 사이트에서 특정 질의에 대해, 답변이 accept되거나, 5개 이상의 투표를 받고,

      II. 하나 이상의 URL Link를 포함하는 경우

   B. Query(Q), Positive Document(D^+) construction

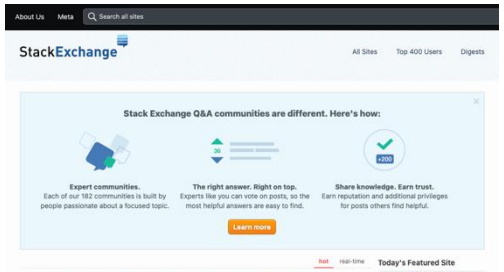
      - 선정한 Document의 title+content를 query로 선정
      - 첨부된 url에 방문하여 Annotator가 직접 보고, 관련된 pargraph를 positive document로 선정
      - Annotator: 1 C.S. student -> 2 PH.D. students -> 2 Expert reviewers

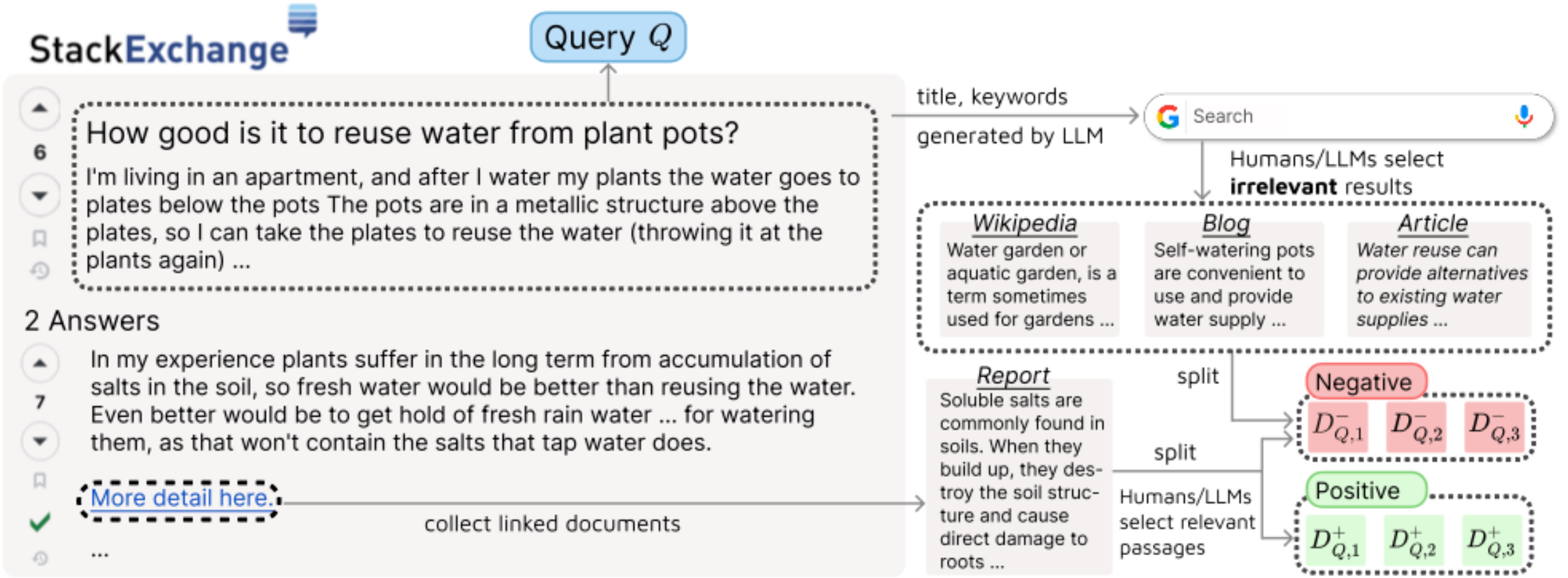
   C. Hard Negative Documents(D^-) construction

      - Document로 부터 LLM을 활용한 keyword 생성 -> 검색 -> Annotator가 hard negative 최대 5개 선정

1. BRIGHT

1. 데이터셋 상세





1. 데이터셋 상세

2. Coding Dataset (2 domains)

- 대상

A. **Pony** 언어 관련 문제 (희귀 언어 Pony의 특정 구문, 문법 함수를 찾아야 하는 질문)

- Pony 언어에 관한 질의와 Pony 표준 문서(함수/구조/클래스/메서드 등)을 가져와서 매핑.
- 질문(“이 언어에서 조건문이나 if를 어떻게 쓰나요?”) ↔ ‘if’를 설명하는 문서가 양성.
- 이렇게 희귀 언어를 사용한 이유는, C/Python/Java와 달리 질문과 정답 문서 간 “표면적 단어”가 거의 안 겹치기 때문
  → 표면적 매칭으로는 찾기 어렵

B. **LeetCode** 문제 (비슷한 알고리즘/자료구조를 쓰는 다른 문제와 해설을 찾기 위함)

- e.g. 트래핑 레인 워터(투 포인터 이용) ↔ 최대 사각형 넓이(투 포인터 이용)
  → 표현 상으로는 달라도, 동일 알고리즘을 필요로 함.

1. BRIGHT

1. 데이터셋 상세

2. Coding Dataset (2 domains) 예시

Query
Given the lengths of a triangle's sides, write a pony program to classify it as equilateral, isosceles or scalene.

Example positive document

# Control Structures 	
 	
To do real work in a program you have to be able to make decisions, iterate
through collections of items and perform actions repeatedly. For this, you need
control structures. Pony has control structures that will be familiar to programmers who have
used most languages,
such as 'if', 'while' and 'for', but in Pony, they work slightly differently.
 	
## Conditionals 	
 	
The simplest control structure is the good old 'if'. It allows you to perform some action only
when a condition is true.
In Pony it looks like this:
 	
if condition then
control_body
end
 	
Here is a simple example:
 	
if a > b then
env.out.print("a is bigger")
end
 	
Often the condition may be composed of many sub conditions connected by 'and' and 'or'.
...

Example negative document
Classes Just like other object-oriented languages, Pony has __classes__. A class is declared with the keyword 'class', and it has to have a name that starts with a capital letter, like this: class Wombat Do all types start with a capital letter? Yes! And nothing else starts with a capital letter. So when you see a name in Pony code, you will instantly know whether it's a type or not. ## What goes in a class? A class is composed of: 1. Fields. 2. Constructors. 3. Functions. ...

1. BRIGHT

1. 데이터셋 상세

2. Theorem-based (3 domains)

- 수학/과학 문제 해결에서 핵심이 되는 정리(theorem)나 아이디어를 공유하는 문서를 찾기.
- 수학이나 물리 문제 중, 예컨대 "푸리에 변환"이나 "피죤홀 원리"를 사용해야 하는 문제를 다룬다.

1. TheoremQA-Q

- TheoremQA의 각 질문을 GPT-4로 "표면적인 정리명을 빼고 일상적·실용적 시나리오"로 재작성.
- (ex) "피죤홀 원리를 직접 언급하는 것" → "사탕 바구니가 … 최소 몇 명에게 나눠주면 한 명 이상이 중복?" 같은 식으로 변형.
- 골드 문서: 해당 문제와 똑같은 정리를 쓰는 다른 문제(문제+해설)를 corpus에서 찾아서 양성 처리.

2. AoPS

- AoPS Wiki(수학 올림피아드)에서 "정리/문제 해결 스킬"(ex. 페르마 소정리, Pick's theorem 등)을 모아,
- 이 스킬이 쓰인 여러 문제(각각 문제+솔루션)들 중, 스킬이 같은 문제끼리는 양성으로 간주.

3. TheoremQA-T

- TheoremQA 질문 그대로 두고, ProofWiki(수학 정리 데이터베이스)에서 "문제 해법에 실제 사용된 정리"를 찾아 매핑.
- "정확히 이 정리"가 해법에 필요하면 +, 아니면 -.

1. BRIGHT

1. 데이터셋 상세

2. Theorem-based (3 domains) 예시

Query *(from 2015 AMC 10B Problem 15)*

The town of Hamlet has 3 people for each horse, 4 sheep for each cow, and 3 ducks for each person. Which of the following could not possibly be the total number of people, horses, sheep, cows, and ducks in Hamlet?
(A) 41 **(B)** 47 **(C)** 59 **(D)** 61 **(E)** 66

Chain-of-thought reasoning to find documents

We can use the Chicken McNugget Theorem to solve this problem. We can find other solutions that also apply this theorem.

Example positive document

Find the sum of all positive integers n such that, given an unlimited supply of stamps of denominations $5, n$, and $n+1$ cents, 91 cents is the greatest postage that cannot be formed.
By the Chicken McNugget theorem, the least possible value of n such that 91 cents cannot be formed satisfies $5n-(5+n)=91 \implies n=24$, so n must be at least 24.
For a value of n to work, we must not only be unable to form the value 91, but we must also be able to form the values 92 through 96, as with these five values, we can form any value greater than 96 by using additional 5 cent stamps.
Notice that we must form the value 96 without forming the value 91. If we use any 5 cent stamps when forming 96, we could simply remove one to get 91. This means that we must obtain the value 96 using only stamps of denominations n and $n+1$.
Recalling that $n \geq 24$, we can easily figure out the working $(n, n+1)$ pairs that can used to obtain 96, as we can use at most $\frac{96}{24}=4$ stamps without going over. The potential sets are $(24,25), (31,32), (32,33), (47,48), (48,49), (95,96)$, and $(96,97)$.
The last two obviously do not work, since they are too large to form the values 92 through 94, and by a little testing, only $(24,25)$ and $(47,48)$ can form the necessary values, so $n \in \{24,47\}$. $24+47=\boxed{71}$.

Example negative document

Alice has 24 apples. In how many ways can she share them with Becky and Chris so that each of the three people has at least two apples?
(A) 105 **(B)** 114 **(C)** 190 **(D)** 210 **(E)** 380
Note: This solution uses the non-negative version for stars and bars. A solution using the positive version of stars is similar (first removing an apple from each person instead of 2).
This method uses the counting method of stars and bars (non-negative version). Since each person must have at least 2 apples, we can remove $2*3$ apples from the total that need to be sorted. With the remaining 18 apples, we can use stars and bars to determine the number of possibilities. Assume there are 18 stars in a row, and 2 bars, which will be placed to separate the stars into groups of 3. In total, there are 18 spaces for stars $+2$ spaces for bars, for a total of 20 spaces. We can now do $\binom{20}{2}$. This is because if we choose distinct 2 spots for the bars to be placed, each combo of 3 groups will be different, and all apples will add up to 18. We can also do this because the apples are indistinguishable. $\binom{20}{2}$ is 190, therefore the answer is $\boxed{\textbf{(C)}\ 190}$.

1. BRIGHT

2. Experiments & Results

- Baseline
  - Sparse: BM25
  - Dense (small): Sentence-BERT(109M), BGE(335M), Instructor-Large(335M)
  - Dense (large): Instructor-XL(1.5B), E5(7B), SFR(7B), GritLM(7B), gte-Qwen1.5(7.7B)
  - Proprietary API: Cohere, OpenAI, Voyage, Google
- Metric
  - nDCG@10, Precision@10, Recall@10

1. BRIGHT

2. Experiments & Results

	StackExchange							Coding		Theorem-based			Avg.
	Bio.	Earth.	Econ.	Psy.	Rob.	Stack.	Sus.	Leet.	Pony	AoPS	TheoQ.	TheoT.	
Sparse model													
BM25	18.9	27.2	14.9	12.5	13.6	18.4	15.0	24.4	7.9	6.2	10.4	4.9	14.5
Open-sourced models (<1B)													
BGE	11.7	24.6	16.6	17.5	11.7	10.8	13.3	26.7	5.7	6.0	13.0	6.9	13.7
Inst-L	15.2	21.2	14.7	22.3	11.4	13.3	13.5	19.5	1.3	8.1	20.9	9.1	14.2
SBERT	15.1	20.4	16.6	22.7	8.2	11.0	15.3	26.4	7.0	5.3	20.0	10.8	14.9
Open-sourced models (>1B)													
E5	18.6	26.0	15.5	15.8	16.3	11.2	18.1	28.7	4.9	7.1	26.1	26.8	17.9
SFR	19.1	26.7	17.8	19.0	16.3	14.4	19.2	27.4	2.0	7.4	24.3	26.0	18.3
Inst-XL	21.6	34.3	**22.4**	27.4	**18.2**	21.2	19.1	27.5	5.0	8.5	15.6	5.9	18.9
GritLM	24.8	32.3	18.9	19.8	17.1	13.6	17.8	29.9	**22.0**	8.8	25.2	21.2	21.0
Qwen	**30.6**	**36.4**	17.8	24.6	13.2	**22.2**	14.8	25.5	9.9	**14.4**	**27.8**	**32.9**	**22.5**
Proprietary models													
Cohere	18.7	28.4	20.4	21.6	16.3	18.3	17.6	26.8	1.9	6.3	15.7	7.2	16.6
OpenAI	23.3	26.7	19.5	27.6	12.8	14.3	**20.5**	23.6	2.4	8.5	23.5	11.7	17.9
Voyage	23.1	25.4	19.9	24.9	10.8	16.8	15.4	**30.6**	1.5	7.5	27.4	11.6	17.9
Google	22.7	34.8	19.6	**27.8**	15.7	20.1	17.1	29.6	3.6	9.3	23.8	15.9	20.0

1. BRIGHT

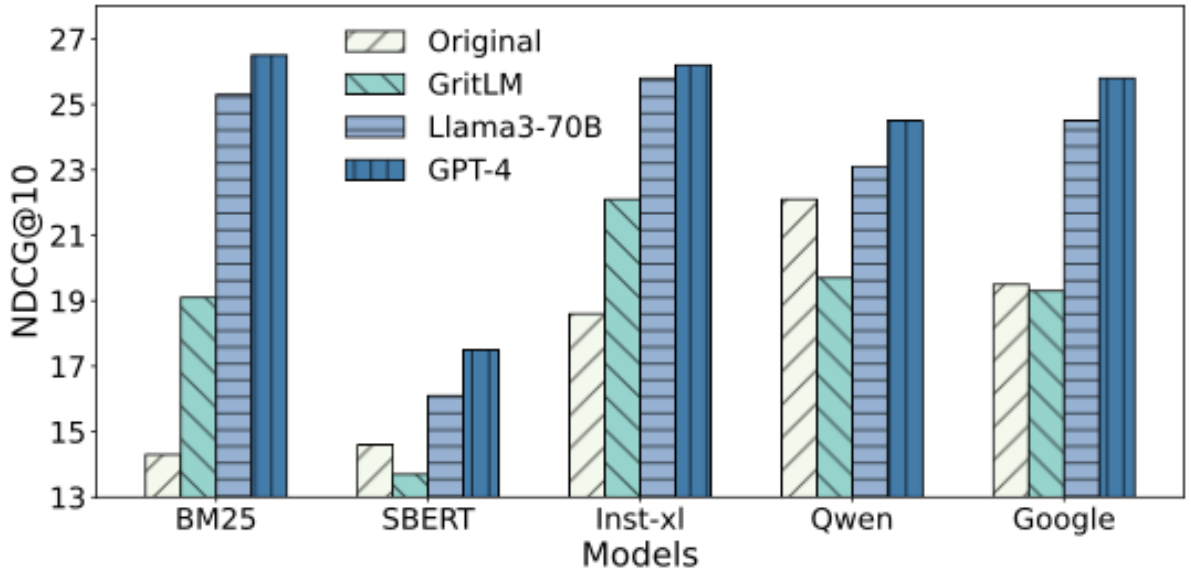
2. Experiments & Results

- LLM-generated reasoning path이 검색 성능을 향상시킨다?
- GPT-4, GritLM, Llama-3-70B-Instruct로 query가 주어진 경우, reasoning path를 생성하고, 그 path를 query로 사용하여 검색

  "(1) Identify the essential problem in the post.

  (2) Think step by step to reason about what should be included in the relevant documents.

  (3) Draft an answer."



1. BRIGHT

2. Experiments & Results

- (당연하게도) 검색 시 query rewriting ➔ 검색 perf. 증가 ➔ RAG 성능 증가

Table 4: **Question-answering results with different retrievers.** We use Claude-3.5-sonnet as the generation model and evaluate the answers with Claude-3.5-sonnet. We find that stronger retrieval typically results in better QA results, indicating the helpfulness of the annotated documents for addressing the posts in StackExchange.

Retriever	Bio.	Earth.	Econ.	Psy.	Rob.	Stack.	Sus.	Average
None	79.4	82.3	75.6	74.5	76.7	81.8	73.5	77.7
BM25	78.2	82.6	76.3	78.2	76.3	83.0	73.6	78.3
SBERT	79.6	82.5	75.8	80.6	77.0	83.4	**74.1**	79.0
Qwen	**80.2**	**83.5**	**77.0**	**81.1**	**77.2**	**85.8**	72.6	79.6
Oracle	*82.4*	*84.5*	*78.3*	*82.4*	*78.5*	*87.9*	*78.6*	*81.8*

3. Analysis

3-1. RERANKING WITH LLMS ENHANCES RETRIEVAL PERFORMANCE

Retriever	Reranker	k	nDCG@10
BM25	None	-	14.3
	MiniLM	10	13.1
	MiniLM	100	8.3
	Gemini	10	15.7
	GPT-4	10	17.4
	GPT-4	100	17.0
Google	None	-	19.5
	MiniLM	10	16.0
	MiniLM	100	9.4
	Gemini	10	20.1
	GPT-4	10	21.5
	GPT-4	100	22.6

1. BRIGHT

3. Analysis

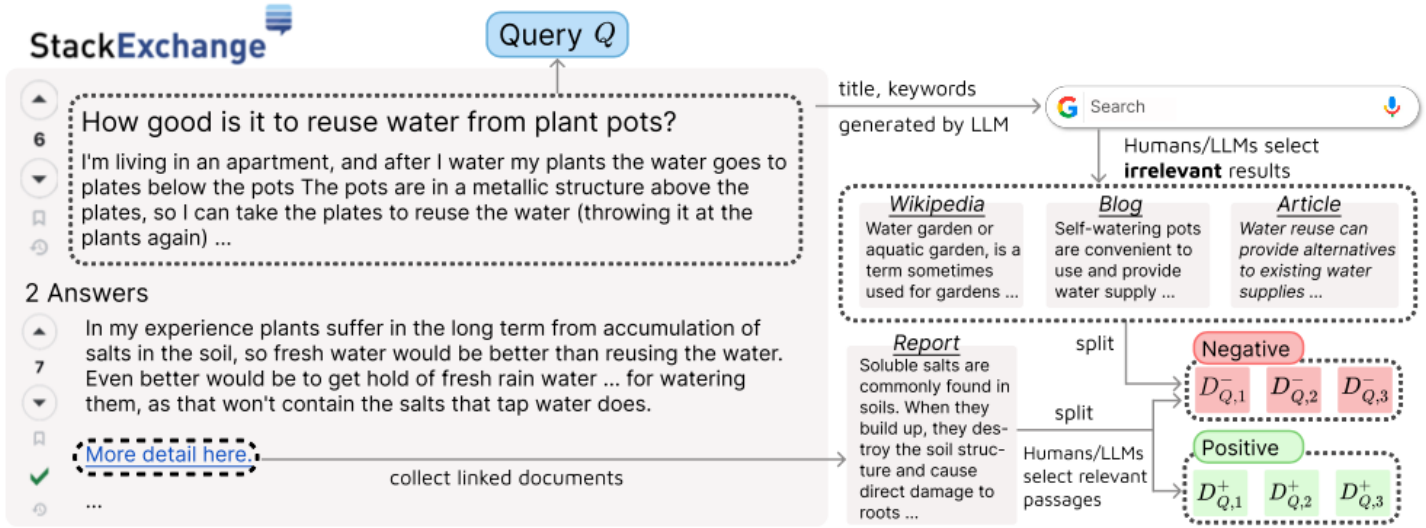
3-2. ROBUSTNESS AGAINST DATA LEAKAGE FROM PRETRAINING

- 그간 검색 모델들은 data leakage 문제를 의심 받아왔음

  (실제 mteb의 경우, llm 기반의 임베딩 모델이 LLM pretrain 시 해당 데이터를 본 경우가 있을 수도 있음)

- 따라서, 이런 것들을 테스트해보고자 GritLM 모델 학습 레시피에 맞추어, StackExchange 등에서

  1) positive, negative document로 LM CPT, 2)**Query-Answer** pair를 수집해서 Contrastive train



3. Analysis

3-2. ROBUSTNESS AGAINST DATA LEAKAGE FROM PRETRAINING

Epoch	Bio.	Earth.	Econ.	Psy.	Rob.	Stack.	Sus.	Avg.
0 (GritLM)	25.0	32.8	19.0	19.9	17.3	11.6	18.0	20.5
1	22.2	25.4	17.6	28.1	11.1	9.8	19.6	19.1
2	18.7	23.8	13.5	19.3	10.7	10.2	16.5	16.1
3	20.9	23.6	16.9	25.2	11.1	8.5	16.6	17.5
4	24.3	28.0	18.3	26.9	13.4	13.3	20.0	20.6
5	23.1	28.5	18.4	26.1	14.6	11.7	21.6	20.6
6	19.9	26.4	16.0	27.9	9.6	9.3	19.3	18.3
7	24.3	25.4	16.5	28.1	11.0	9.8	17.0	18.9
8	21.6	28.7	19.2	28.7	11.1	11.8	22.4	20.5
9	21.3	29.0	20.0	28.7	11.4	14.3	22.0	21.0
10	21.1	25.5	18.8	30.7	12.7	12.1	21.9	20.4

1. BRIGHT

3. Analysis

Dataset	Total Number			Avg. Length		Source		Examples
	Q	$\mathcal{D}$	$\mathcal{D}^+$	Q	$\mathcal{D}$	Q	$\mathcal{D}$	
StackExchange								
Biology	103	57,359	3.6	115.2	83.6	StackExchange post	Web pages: article, tutorial, news, blog, report ...	Tab. 20
Earth Science	116	121,249	5.3	109.5	132.6			Tab. 21
Economics	103	50,220	8.0	181.5	120.2			Tab. 22
Psychology	101	52,835	7.3	149.6	118.2			Tab. 23
Robotics	101	61,961	5.5	818.9	121.0			Tab. 24
Stack Overflow	117	107,081	7.0	478.3	704.7			Tab. 25
Sustainable Living	108	60,792	5.6	148.5	107.9			Tab. 26
Coding								
LeetCode	142	413,932	1.8	497.5	482.6	Coding question	Coding Q&Sol	Tab. 27
Pony	112	7,894	22.5	102.6	98.3	Coding question	Syntax Doc	Tab. 28
Theorems								
AoPS	111	188,002	4.7	117.1	250.5	Math Olympiad Q	STEM Q&Sol	Tab. 29
TheoremQA-Q	194	188,002	3.2	93.4	250.5	Theorem-based Q	STEM Q&Sol	Tab. 30
TheoremQA-T	76	23,839	2.0	91.7	354.8	Theorem-based Q	Theorems	Tab. 31

1. BRIGHT

3. Analysis

3-3. LONG-CONTEXT RETRIEVAL WITH A REDUCED SEARCH SPACE IS CHALLENGING

- 실제 Application에서는 long document retrieval이 굉장히 중요. 이러한 능력을 평가하기 위해 StackExchange 데이터셋을 long-context retrieval setting으로 변환 (기존에 Annoator가 passage 단위를 positive document로 채택했던 것 대신 원본 long document 자체를 positive으로 채택)
- 이렇게 길게 바꾸니, 전체 pool은 적어지기 때문에 (few hundreds) Recall@1을 평가지표로 사용

Table 6: **Long-context retrieval performance where retrievers retrieve from unsplit web pages.** The results are reported as the average recall@1 score of StackExchange and Pony datasets. More detailed numbers can be found in Table 39.

BM25	BGE	Inst-L	SBERT	E5	SFR	Inst-XL	GritLM	Qwen	Cohere	OpenAI	Voyage	Google
11.4	14.8	18.2	17.4	25.5	26.0	17.8	26.0	27.8	18.4	21.9	24.6	22.4

4. Conclusion

- **'검색에도 reasoning이 필요하다. 그리고 현대 검색 모델들은 해당 태스크를 잘 수행하지 못한다 !'** 를 알려주는 벤치마크
- 다양한 실험과 평가, 여러 분석들을 제시하며 BRIGHT가 얼마나 challenging한 태스크인지 입증
- 벤치마크 페이퍼의 좋은 예시지만, 몇몇 실험들은 왜 해본건지 이해가 안되기도 함

ReasonIR:

Training Retrievers for Reasoning Tasks

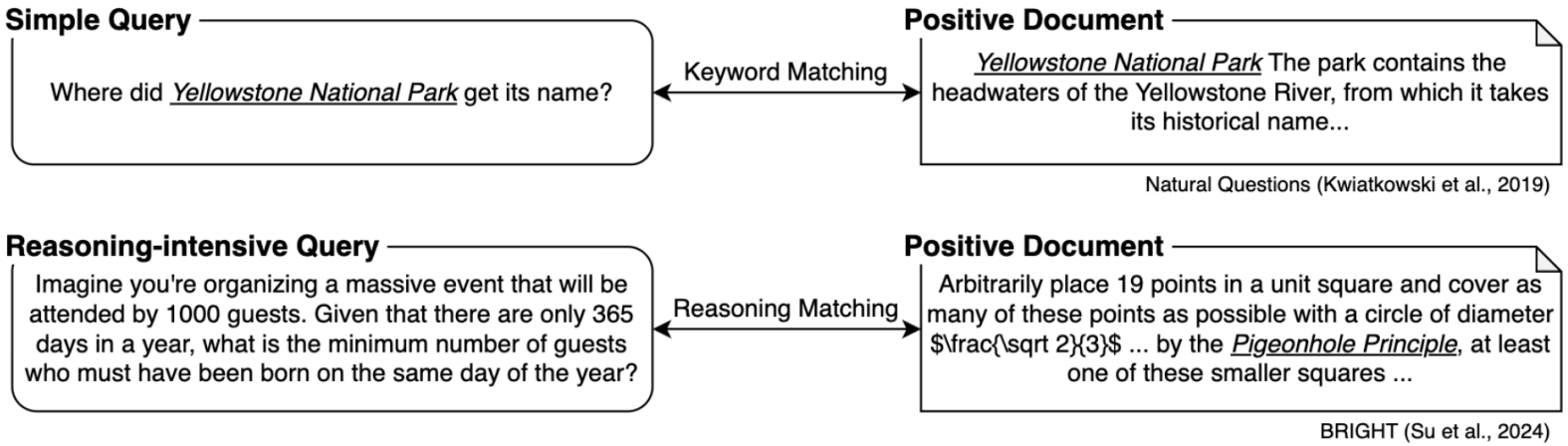
ArXiv (Meta)

2. ReasonIR

0. Overview

- Problem Statement

1. 현대 검색 모델들이 Reasoning-Intensive IR에 약한 이유는, **그러한 데이터로 학습된 적이 없기** 때문이다.
   - 기존 데이터 - 단순 팩트를 물어보는 질의 / BRIGHT – 엄청난 추론력을 요하는 질의
   - 질의의 평균 길이에서도 차이가 크다 (21 tokens vs. 194 tokens)



2. Query rewriting을 통해 query를 길게, information-rich하게 만들어고 학습하면 성능이 향상된다

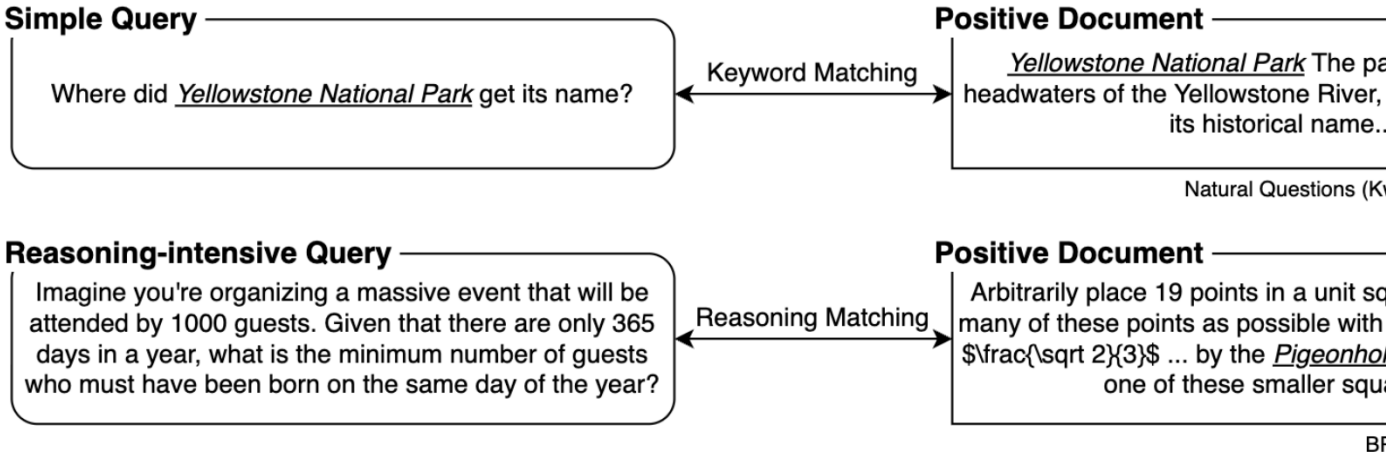
2. ReasonIR

0. Overview

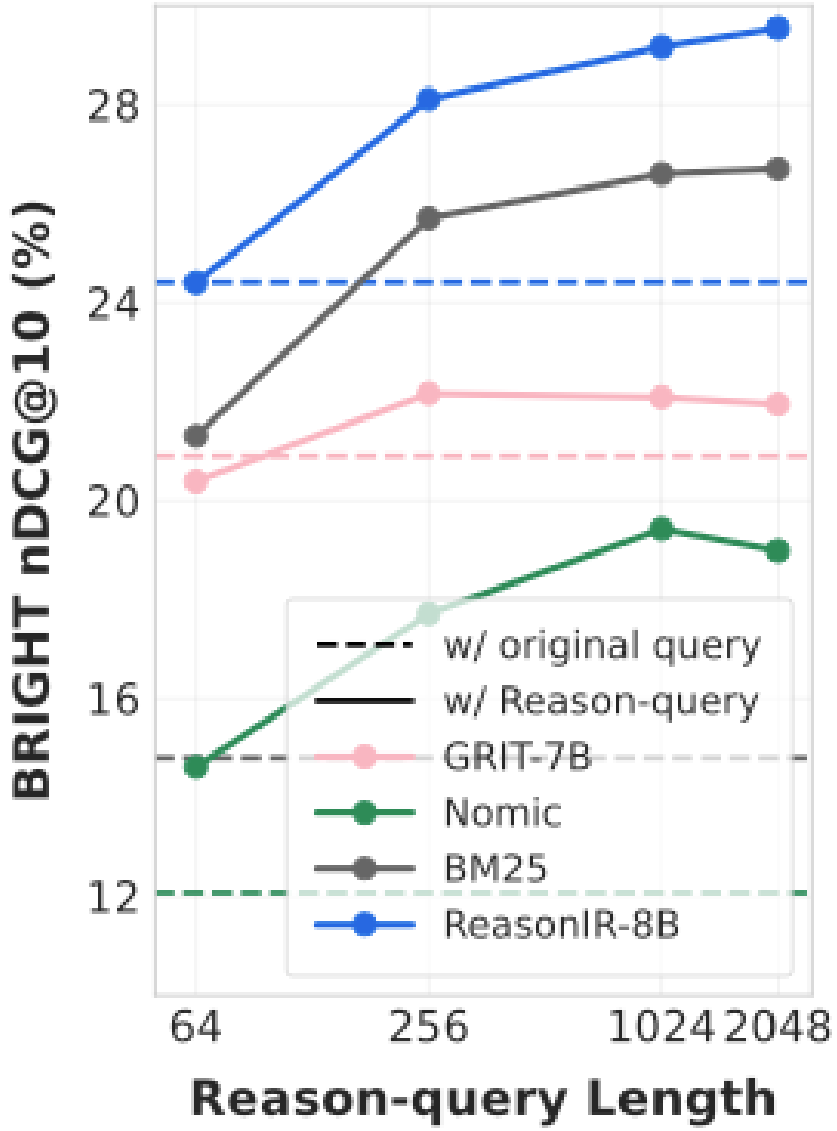
- Problem Statement

1. 현대 검색 모델들이 Reasoning-Intensive IR에 약한 이유는, 그러한 더

   - 기존 데이터 - 단순 팩트를 물어보는 질의 / BRIGHT – 엄청난 추론력을 요하
   - 질의의 평균 길이에서도 차이가 크다 (21 tokens vs. 194 tokens)



2. Query rewriting을 통해 query를 길게, information-rich하게 만들어



2. ReasonIR

0. Overview

- Contribution

    1. Reasoning-Intensive IR을 학습시키기 위해 REASONIR-SYNTHESIZER (합성 데이터 파이프라인) 만듦
    2. 합성데이터 + 공개데이터로 Llama3.1-8B SFT ➔ REASONIR-8B 모델 개발

2. ReasonIR

1. Preliminaries

기존 임베딩 모델 개발 방식

$$\mathcal{L} = -\log \frac{\exp(s(q, d^+)/\tau)}{\sum_{i=1}^{N} \exp(s(q, d^i)/\tau)},$$

Batch Size

Query	Positive	Hard Negative
고양이의 습성	고양이는 독립적인 행동을 보이며, 사냥 본능이 강하게 남아 있습니다. 특히 밤에 활동이 활발해지는 야행성 습성이 있습니다.	개와 고양이는 둘 다 반려동물로 사랑받고 있으며, 각기 다른 특성을 가지고 있습니다. 개는 주로 사람에게 친근하게 다가가며, 사회적인 성격을 가집니다.
생명 보험의 필요성	생명 보험은 갑작스러운 사고나 질병으로 인한 경제적 부담을 덜어줍니다. 특히 가족을 부양하는 사람들에게 금융적인 안전망을 제공합니다.	보험은 다양한 종류가 있으며, 자동차 보험, 건강 보험 등 각 분야에 특화된 상품이 많습니다. 서로 다른 위험 요소를 커버합니다.
커피의 건강 효과에 대하여	커피는 항산화 물질을 포함하고 있어 신진대사를 촉진하고, 에너지를 증가시킵니다. 적당량의 커피 섭취는 집중력을 높이는 데 도움이 됩니다.	커피와 차는 모두 전 세계적으로 사랑받는 음료로, 카페인 함유량에서 차이를 보입니다. 차는 주로 테아닌이라는 성분을 통해 안정 효과를 주기도 합니다.

2. ReasonIR

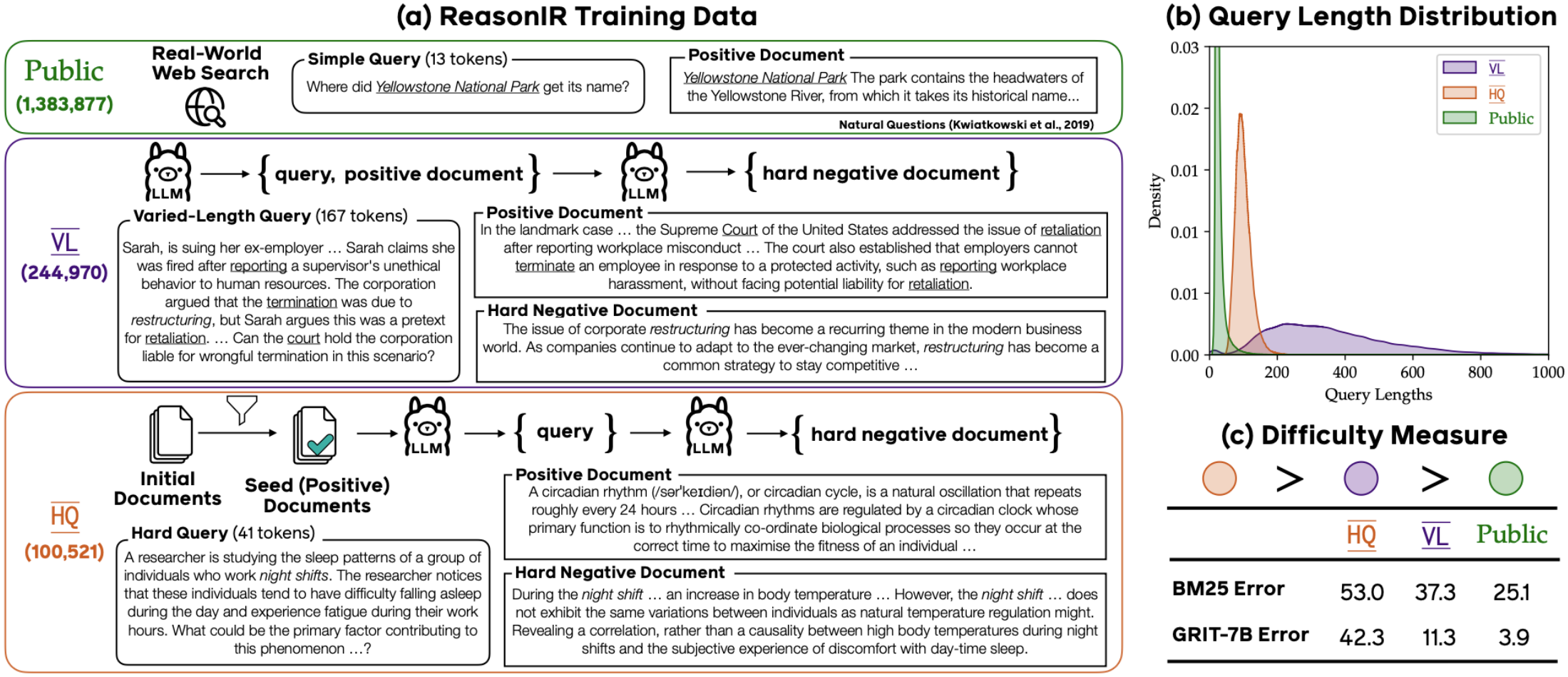
2. REASONIR-SYNTHESIZER

1. **Public** Data – NQ, MSMARCO와 같은 일반 학습 데이터

2. **VL (Varied-Length)** Data
   - (E5-Mistral 차용) 1차적으로 Task Brainstorming
   - 이후 해당 task 기반하여 긴 query와 positive, hard negative를 순차적으로 생성하도록 프롬프팅

3. **Hard Query (HQ)** Data ➔ document를 기반으로 query와 hard negative 생성
   - BRIGHT 벤치마크에 있는 document pool을, FineWeb-Edu Classifier로 점수 측정
   - 점수가 2점 이상인 document를 유의미한 document로 판단하며, 해당 document를 기반으로 LLM에게 어려운 query를 생성하도록 함
   - Query 생성 시, prompt로 1) 배경 개념 정리 / 2) 흔한 솔루션 패턴 떠올리기 / 3) 현실 시나리오 상상하기 / 4) 문서의 특정 어휘를 그대로 따오지 말기
   - HN 생성 시, 생성된 query와 positive docs를 주고 LLM에게 "겉보기에는 관련 있어 보이지만 답을 하지는 못하는 장문"을 생성하도록 요청
   - [생성된 Q] – [벤치마크로부터 얻은 D] – [생성된 HN] triplet으로 contrastive learning 수행
   - 근데 이거 치팅 아닌가 ???

2. REASONIR-SYNTHESIZER



(c) Difficulty Measure

HQ > VL > Public

	HQ	VL	Public
BM25 Error	53.0	37.3	25.1
GRIT-7B Error	42.3	11.3	3.9

2. ReasonIR

2. REASONIR-Embedding

- 앞서 확보한 데이터셋으로 모델 학습
- Base model: Llama3.1-8B (changed to bidirectional attention)
- Training Dataset
  - Public: 1,383,877
  - Varied-Length(VL): 244,970
  - Hard Query(HQ): 100,521
- Evaluation: BRIGHT benchmark for retrieval, MMLU, GPQA for RAG

2. ReasonIR

2. REASONIR-RERANKER

- 임베딩 모델과 더불어 아예 reranker까지 한번에 개발
- 기존 Naïve LLM-based Rerankers (query doc 주어지고 0~5점 산출하게 하는 방식)의 성능이 ReasonIR 태스크에서 너무 낮았음
- 그래서 이전에 개발되었던 **Rank1** model← LRM (Large Reasoning Model)들로부터 distillation 받아옴
- 하지만 이건 너무 비싸다 ! - Naïve LLM-based rerankers의 성능 저하 원인 규명

  ➔ 동점이 되는 경우가 너무 많기 때문이다
- 해결하기 위해 retriever의 score로 interpolation 진행
- 기존 Naïve LLM-based reranker의 점수는 그대로 둔 채, retriever의 점수만 더해줌 (Tie-breaking Method)

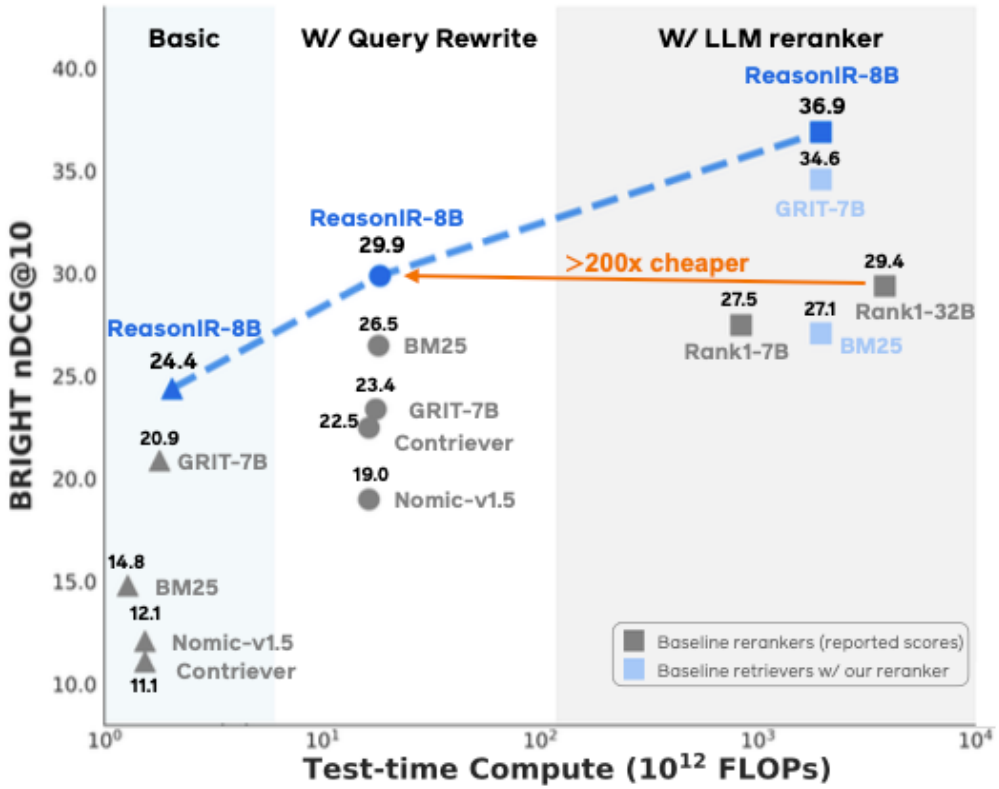
$$\text{최종 점수} = \alpha \cdot s_{\text{retriever}} + (1 - \alpha) \cdot s_{\text{LLM}}$$

2. REASONIR-RERANKER

- 논문에서는 이렇게 score를 합쳐 산출하는 방식을 REASONIR-Rerank라고 칭함 (이걸 발견이라고 할 수 있나 ?)
- 실험에서는 Qwen2.5-32B-IT 모델 기반으로, Retriever score interpolation을 적용하여 **QwenRerank**라고 칭함

2. ReasonIR

3-1. Evaluation Results - IR



(a) IR Performance.

설정	REASONIR-8B	GRIT-7B	Rank1-32B (LLM rerank)
원본 질의	**24.4**	20.9	—
GPT-4 REASON-QUERY	**29.9**	23.4	29.4
+ Qwen Rerank	**36.9**	—	—

2. ReasonIR

3-1. Evaluation Results - IR

	StackExchange							Coding		Theorem-based			Avg.
	Bio.	Earth.	Econ.	Psy.	Rob.	Stack.	Sus.	Leet.	Pony	AoPS	TheoQ.	TheoT.	
Evaluate with original query													
BM25	19.2	27.1	14.9	12.5	13.5	16.5	15.2	24.4	7.9	6.0	13.0	6.9	14.8
Contriever	9.2	13.6	10.5	12.1	9.5	9.6	8.9	24.5	14.7	7.2	10.4	3.2	11.1
GritLM-7B	25.0	32.8	19.0	19.9	17.3	11.6	18.0	29.8	**22.0**	8.8	25.1	21.1	20.9
OpenAI	23.7	26.3	20.0	27.5	12.9	12.5	20.3	23.6	2.5	8.5	23.8	12.3	17.8
Voyage	23.6	25.1	19.8	24.8	11.2	15.0	15.6	30.6	1.5	7.4	26.1	11.1	17.7
Google	23.0	**34.4**	19.5	27.9	16.0	17.9	17.3	29.6	3.6	9.3	21.5	14.3	19.5
ReasonIR-8B	**26.2**	31.4	**23.3**	**30.0**	**18.0**	**23.9**	**20.5**	**35.0**	10.5	**14.7**	**31.9**	**27.2**	**24.4**
Evaluate with LLAMA3.1-8B-INSTRUCT REASON-QUERY													
ReasonIR-8B	37.8	39.6	29.6	35.3	24.1	31.1	27.4	28.8	14.5	9.2	26.6	32.3	28.0
+ BM25 (Hybrid)	51.9	50.6	24.0	40.6	26.9	31.0	28.5	26.2	17.8	9.2	22.3	22.5	29.3
Evaluate with GPT4 REASON-QUERY													
BM25	53.6	53.6	24.3	38.6	18.8	22.7	25.9	19.3	17.7	3.9	20.2	18.9	26.5
Contriever	37.5	40.5	22.6	27.1	15.2	22.6	19.6	22.5	13.8	8.1	24.1	16.2	22.5
GritLM-7B	33.2	33.0	23.3	30.6	15.2	17.5	21.7	**33.2**	11.7	6.8	26.9	28.0	23.4
RankLLaMA-7B (top-100)$	17.5	15.5	13.1	13.6	17.9	6.9	16.9	8.4	**46.8**	2.2	4.5	3.5	13.9
Rank1-7B (top-100)$	48.8	36.7	20.8	35.0	22.0	18.7	**36.2**	12.7	31.2	6.3	23.7	37.8	27.5
Rank1-32B (top-100)$	49.7	35.8	22.0	37.5	22.5	21.7	35.0	18.8	32.5	**10.8**	22.9	43.7	29.4
ReasonIR-8B	43.6	42.9	**32.7**	38.8	20.9	25.8	27.5	31.5	19.6	7.4	33.1	35.7	**29.9**
+ BM25 (Hybrid)	55.9	**54.9**	29.6	42.9	23.0	27.9	29.8	27.9	25.8	7.2	33.7	25.8	**32.0**
+ QwenRerank (top-100)$	**58.2**	53.2	32.0	**43.6**	**28.8**	**37.6**	36.0	**33.2**	34.8	7.9	32.6	**45.0**	**36.9**

2. ReasonIR

3-2. Evaluation Results - RAG

Retriever	Query Type	MMLU	GPQA
Closed-book	-	71.1	31.3
Contriever	Original question	72.0	36.4
GRIT-7B	Original question	74.1	32.3
Search Engine	Original question	-	33.8
ReasonIR-8B	Original question	75.0	**38.4**
Contriever	REASON-QUERY	72.8	31.3
GRIT-7B	REASON-QUERY	74.7	30.8
Search Engine	REASON-QUERY	-	36.4
ReasonIR-8B	REASON-QUERY	**75.6**	35.4

2. ReasonIR

3-3. Evaluation Results - Ablation

	StackExchange							Coding		Theorem-based			Avg.
	Bio.	Earth.	Econ.	Psy.	Rob.	Stack.	Sus.	Leet.	Pony	AoPS	TheoQ.	TheoT.	
Evaluate with original query													
LLAMA3.1-8B	12.5	6.5	7.7	7.7	3.9	7.5	8.6	22.0	17.1	10.5	7.4	2.0	9.5
Public	21.4	30.3	17.8	24.7	18.6	18.8	18.8	30.0	6.7	12.1	21.4	15.2	19.6
Public+HQ	21.0	31.3	18.4	25.1	15.7	18.4	14.3	34.1	5.2	9.5	33.7	24.4	20.9
Public+VL	28.4	35.8	22.5	28.4	18.4	19.5	18.7	34.5	12.3	11.4	24.4	23.6	23.2
Public+EQVL	26.8	33.8	23.4	30.1	21.1	21.9	21.5	31.0	6.5	10.1	20.9	20.2	22.3
Public+HQVL	26.2	31.4	23.3	30.0	18.0	23.9	20.5	35.0	10.5	14.7	31.9	27.2	**24.4**
Evaluate with GPT4 REASON-QUERY													
LLAMA3.1-8B	41.3	25.1	16.8	17.3	8.7	10.7	15.7	6.8	32.3	0.9	12.3	4.0	16.0
Public	40.3	42.1	26.0	37.7	20.8	22.6	22.7	32.3	13.5	7.0	29.5	30.4	27.1
Public+HQ	37.4	42.7	26.8	35.3	18.2	22.1	20.0	35.0	14.7	6.7	34.1	32.7	27.1
Public+VL	33.8	41.3	28.9	40.2	20.6	24.2	25.9	34.7	19.6	4.8	32.5	29.2	28.0
Public+EQVL	37.9	42.1	30.6	40.0	22.1	25.6	27.4	31.8	15.5	6.1	27.3	28.7	27.9
Public+HQVL	43.6	42.9	32.7	38.8	20.9	25.8	27.5	31.5	19.6	7.4	33.1	35.7	**29.9**

세개 다 쓴거 > Public + VL > Public + HQ > Public

2. ReasonIR

4. Conclusion

- **"검색기가 Reasoning-Intensive IR 벤치마크에서 낮은 성능을 보이는 이유는 그러한 데이터를 보지 않았기 때문이다!"** 주장
- 이를 해결하기 위해 2가지 데이터 (VL, HQ) 합성 파이프라인 개발
- 개인적으로 HQ는 치팅이라고 생각. 좋게 말해 In-domain 데이터라고 치부할 수 있겠지만, 그럴거면 Reasoning-Intensive OOD 벤치마크에 대한 실험도 했었어야 하는 것 같음
- 이런 데이터를 encoder 모델이 학습해도 괜찮은 성능을 낼지 궁금

Thank you

Q&A