FreshStack: Building Realistic Benchmarks for Evaluating Retrieval on Technical Documents

Nandan Thakur[1] Jimmy Lin[1] Sam Havens[2] Michael Carbin[2] Omar Khattab[2] Andrew Drozdov[2]
University of Waterloo[1] Databricks[2]

FreshStack is a part of RTEB (new MTEB) benchmark!

Leaderboard: fresh-stack.github.io/#leaderboard

Dataset (CC-by-SA-4.0): huggingface.co/freshstack

Code & PyPI: github.com/fresh-stack/freshstack

Presenter: **Jacob Portes** (Research Scientist at Databricks)

Motivation

Most academic RAG benchmarks **suffer** from three things:

(1) They lack **realistic** questions and/or **answer** distributions.
(2) They are **artificially** easy because they are built as **"RAG"** datasets.
(3) They are **static** and **unspecialized**.

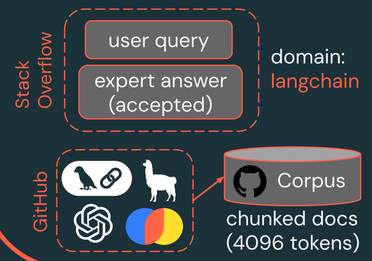
We built an automatic framework to construct realistic RAG evaluation benchmarks!

- **FreshStack** is a technical RAG benchmark with user **queries** on from **StackOverflow** & real-time sourced documents from **GitHub**!
- **FreshStack** includes **five niche** technical domains: (1) ***LangChain*** (2) ***Laravel 10 & 11***, (3) ***Angular 16, 17 & 18***, (4) ***Godot4***, (5) ***YOLO v& & v8***.

Step I: Queries & Corpus

Source queries & answers from niche domains in Stack Overflow. Chunk & combine rerelevant **GitHub repositories** to construct a corpus.

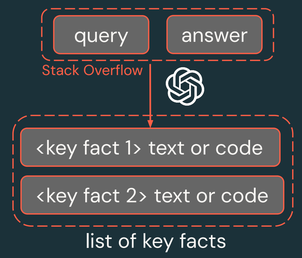
Uses Stack Overflow & GitHub



Step II: Fact Generation

Fact generation breaks down the Stack Overflow answer into multiple ***atomic facts*** which are essential in the RAG answer.

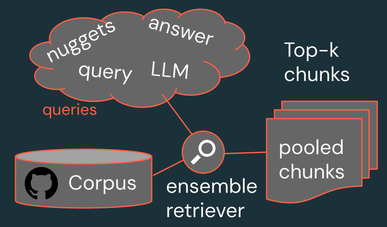
Uses GPT-4o (grading notes)



Step III: Oracle Retrieval

Oracle Retrieval pools relevant *document chunks* from a corpus containing **code-snippets** and documentation.

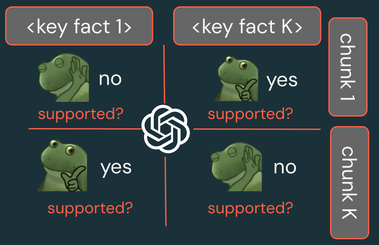
Uses BM25, E5, BGE & Voyage.



Step IV: Support w/ Facts

Grounding techniques provides ***relevance judgements*** of retrieved document chunks for each individual ***key atomic fact***.

Uses GPT-4o as a CoT Judge



Retrieval Results (Avg. 5 domains)

	Model	Size	α-nDCG @10	Coverage@20	Recall@50 (↓)
★	Fusion (BM25, BGE, E5, Voyage)	-	0.343	0.669	**0.539**
★	Qwen3-8B (embedding)	8B	**0.365**	**0.689**	0.525
	Qwen3-4B (embedding)	4B	0.347	0.656	0.490
★	Stella-1.5B v5	1.5B	0.317	0.615	0.479
	Voyage Large 2	-	0.289	0.589	0.438
	BGE (Gemma-2)	9B	0.269	0.569	0.427
★	Stella-400M v5	400M	0.276	0.578	0.422
	Jina V4 (embedding)	3.8B	0.282	0.584	0.425
	E5 (Mistral-7B)	7B	0.255	0.553	0.397
	Qwen3-0.6B (embedding)	596M	0.262	0.543	0.394
	OpenAI text-embedding-3-large	-	0.248	0.537	0.373
	Nomic Embed (code)	7B	0.218	0.488	0.348
	Jina V3 (embedding)	570M	0.227	0.515	0.344
	EmbeddingGemma-300M	300M	0.219	0.508	0.336
	OpenAI text-embedding-3-small	-	0.208	0.480	0.330
	GTE (large) v1.5	434M	0.226	0.494	0.318
	BM25	-	0.218	0.448	0.316
	CodeRankEmbed	137M	0.104	0.279	0.162
	*Oracle setting for upper-baseline (*uses the gold answer/key facts)*				
	Stack Overflow answer + Fusion	-	0.503	0.823	0.721
	Stack Overflow key facts + Fusion	-	**0.541**	**0.868**	**0.755**

Retrieval & RAG Evaluation Metrics

Freshstack uses three evaluation metrics for retrieval-based evaluation:

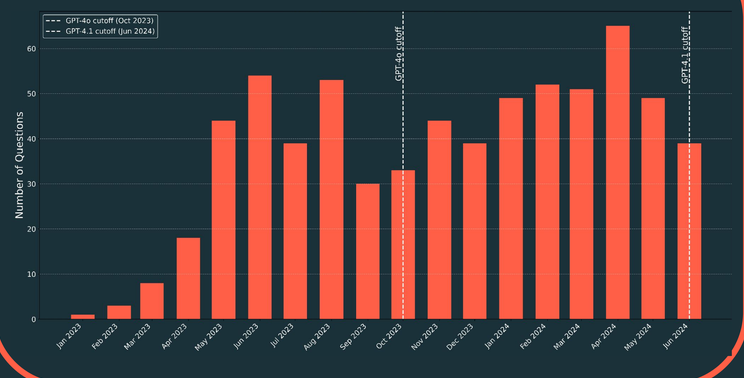
(1) **α-nDCG@10**: A diversity oriented nDCG@10, penalizes redundancy!
(2) **Coverage@20**: % of unique key facts supported by retrieved doc chunks!
(3) **Recall@50:** % of relevant docs retrieved in top 50 / all relevant docs!

RAG evaluation: Measure whether the RAG answer supports each key fact in a three-way judgement: full, partial or no support. Compute the ***All Strict*** metric: ***Count***(fully supported facts by the answer) / **Count**(all facts).

RAG Results (Avg. 5 domains)

Technique	Retrieval	Generator	*"nano"*	*"mini"*	*"full"*
Closed book	-	GPT-4o	-	0.454	0.555
	-	GPT-4.1	0.492	0.609	0.600
Inference (query)	Fusion	GPT-4o	-	0.497	0.601
	Fusion	GPT-4.1	**0.530**	**0.628**	**0.633**
*Oracle setting for upper-baseline (*uses the gold answer/key facts)*					
Stack Overflow key facts	Fusion	GPT-4o	-	0.532	0.640
	Fusion	GPT-4.1	**0.569**	**0.669**	**0.678**

FreshStack Question Distribution



Key Takeaways & Lessons!

(1) **FreshStack** is unlike previous academic RAG benchmarks: (a) longer and complex queries, (b) niche domains, (c) focuses on a realistic setup.

(2) **Off the shelf retrievers** struggle on FreshStack queries, with improvement being observed in latest models such as Qwen3-embeddings.

(3) **Oracle setting** still score much higher than inference setting — indicating less saturation and a plenty of headroom to improve models in FreshStack.

(4) **RAG results** indicate the quality of retrieval leads to a better RAG answer, with a strong closed-book with GPT-4.1 due to recent knowledge cutoff.

Future Work: FreshStack will again be susceptible to contamination & leaderboard overfitting in the future. We will expand FreshStack to newer domains & update the benchmark to limit the pre-training contamination.