Retrieval Evaluation Under Temporal Drift in Technical Documentation

Nathan Kuissi*, Suraj Subrahmanyan*, Nandan Thakur*,✉, Jimmy Lin
University of Waterloo, Canada
*Equal Contribution. ✉ Correspondence: nandan.thakur@uwaterloo.ca

Abstract

Information retrieval (IR) benchmarks typically follow the Cranfield paradigm, relying on static and predefined corpora. However, temporal changes in technical corpora, such as API deprecations and code reorganizations, can render existing benchmarks stale. In our work, we investigate how temporal corpus drift affects FreshStack, a retrieval benchmark focused on technical domains. We examine two independent corpus snapshots of FreshStack from October 2024 and October 2025 to answer queries about LangChain. Our analysis shows that all but one query posed in 2024 remains fully supported by the 2025 corpus, as relevant documents *migrate* from LangChain to competitor repositories, such as LlamaIndex. Next, we compare the accuracy of retrieval models on both snapshots and observe only minor shifts in model rankings, with overall strong correlation of up to 0.978 Kendall τ at Recall@50. These results suggest that retrieval benchmarks for technical documentation, e.g., LangChain, can remain reliable when re-judged in the drift regime we observe, where relevant evidence is *relocated* rather than invalidated. All resources (code and data) are publicly available at `https://github.com/fresh-stack/driftbench`.

1 Introduction

Information retrieval (IR) benchmarks rely on static test collections to evaluate and compare retrieval systems (Zobel, 1998; Voorhees, 2009). Under the Cranfield paradigm, test collections are constructed once with a fixed document corpus, a set of queries, and relevance judgments. Popular IR benchmarks such as MS MARCO (Nguyen et al., 2016), TREC Deep Learning (DL) (Craswell et al., 2021, 2022, 2023, 2024), and BEIR (Thakur et al., 2021) follow this paradigm, with corpus snapshots and relevance judgments fixed at the time of construction.

Test collections have helped establish reusable baselines (Voorhees and Tice, 2000; Carterette et al., 2010; Voorhees, 2018), but information changes and corpora continue to evolve and undergo temporal changes (Voorhees et al., 2022; Keller et al., 2024a,b; Parry et al., 2025). Real-world document collections—such as technical documentation, source code and example snippets—are highly dynamic, as code and documentation are frequently created, deleted, or reorganized.

In our work, we evaluate temporal drift in the LangChain domain from FreshStack (Thakur et al., 2025), because it exhibits the greatest degree of document restructuring, making it a challenging domain for evaluation. We independently construct temporal snapshots of the LangChain corpus, providing documentation for retrieval-augmented generation (RAG) frameworks and developer tools, by reproducing the October 2024 snapshot and extending it to October 2025. We reproduce the FreshStack corpus construction and generate relevance judgments at both temporal snapshots for document retrieval evaluation (Thakur et al., 2025). We reuse the FreshStack nuggets (an atomic or minimal unit of factual information) and retrieve potentially relevant documents using a hybrid fusion incorporating BM25 (Robertson and Zaragoza, 2009), BGE (Gemma-2) (Chen et al., 2024), E5 Mistral (7B) (Wang et al., 2024), and Qwen3 (4B) Embedding (Zhang et al., 2025). Finally, we use Cohere's Command A as the judge (Cohere, 2025) for nugget-level support assessment (Pradeep et al., 2025; Abbasiantaeb et al., 2025) to annotate relevance judgments for all queries across both temporal snapshots. We explore the following three research questions in our work:

RQ1 Can existing queries be grounded in a corpus dynamically changing with time?
RQ2 How does the distribution of relevant documents across repositories change over time?
RQ3 Do model rankings remain consistent under temporal drift in technical documentation?

Findings of the Association for Computational Linguistics: EMNLP 2026, pages 37517–37527
October 24-29, 2026 ©2026 Association for Computational Linguistics

The contribution of this work is, to our knowledge, the first to evaluate the effect of temporal corpus changes on a retrieval benchmark in technical documentation. Prior work has studied temporal changes in more generic corpora, such as news (Voorhees et al., 2022), federal documents (Soboroff, 2006), or scientific and web collections in LongEval (Cancellieri et al., 2025; Galuscáková et al., 2023). The LangChain corpus is different from prior work, as technical documentation and source code exhibit API deprecation, file reorganization, and evidence migration across repositories.

Our experimental results demonstrate that surprisingly *all but one query* remains fully grounded and answerable in both temporal corpus snapshots, despite extensive restructuring in repositories such as LangChain. A major reason for this observation lies in the fact that relevant documents present in LangChain in the October 2024 snapshot are distributed across related GitHub repositories (such as LlamaIndex) in the October 2025 snapshot. Finally, model rankings on the document retrieval task remain strongly correlated (Kendall $\tau = 0.978$ at Recall@50), demonstrating that benchmarks can robustly evaluate retrieval systems, despite temporal changes affecting the corpus.

2 Background and Related Work

Temporal drift refers to the systematic change in a corpus, its queries, and associated relevance judgments over time, caused by additions, deletions, updates, or reorganization of documents. Recent work suggests that test collections may lose validity over time (Parry et al., 2025); however, those studies primarily address judgment variation or query drift. What remains unclear is how benchmarks behave when the corpus undergoes substantial reorganization. Other works share conceptual similarities with our experiments (Galuscáková et al., 2023; El-Ebshihy et al., 2023; Keller et al., 2024a) by focusing on the corpus or the dataset drift; however, they study scientific or web search data. In contrast, our work focuses on evaluating verbose Stack Overflow queries carrying *multiple* information needs, which motivates nugget-level judgment and poses a strong challenge for the retrieval systems being tested (Thakur et al., 2025). Furthermore, Cancellieri et al. (2025) explore similar topics by varying both the corpus and queries across time, whereas Soboroff (2006) reviews the impact of temporal changes on the dataset itself. In addition, GitChameleon (Misra et al., 2026) is a complementary work evaluating LLM's code generation capabilities across temporal code changes. In this work, we limit ourselves to a dynamic corpus such as technical documentation with LangChain and focus on the usability of the retrieval benchmark.

3 Experimental Setup & Details

Building on previous work in FreshStack (Thakur et al., 2025), we use the Stack Overflow queries originally constructed in October 2024 and extend the test collection in October 2025. Additionally, we focus on the **LangChain** domain, as it is representative of our temporal exploration, e.g., the LangChain GitHub repository reduced from 11,037 to 3,628 documents between October 2024 and October 2025 (as shown ahead in Table 1), the largest restructuring among the five domains in FreshStack, which also include Laravel, Angular, Godot, or Yolo (Figure 3 in the Appendix).

3.1 Corpus Preparation

We combine multiple repositories to form our corpus because a query tagged as LangChain on Stack Overflow might require context from other repositories. For example, a LangChain query could concern the ability to properly load a HuggingFace transformer model, or use ChromaDB.

We follow FreshStack (Thakur et al., 2025) and use the main branch snapshot for all GitHub repositories (listed in Table 1) as of October 2024 and October 2025. After providing the repository name, we search for GitHub commits prior to those dates and use the latest commit hash. Then, we chunk all documentation, code files, Jupyter notebooks, etc., with a maximum of 2048 tokens.

3.2 Nugget Generation

Nuggets are atomic key facts that are essential for constructing an answer (Lin and Demner-Fushman, 2005, 2006; Pradeep et al., 2025; Thakur et al., 2025). We automatically generate nuggets from Stack Overflow queries and answers to break down long-form answers and facilitate document-level assessment. The nuggets also crucially serve as queries in oracle retrieval (Thakur et al., 2025).

In addition, LLMs demonstrate a strong ability to extract relevant and concise information from documents, rivaling the human annotation process, which is cumbersome and time-consuming (Pradeep et al., 2025; Metropolitansky

GitHub Repository	October 2024		October 2025	
	#Docs	Relevant (%)	#Docs	Relevant (%)
langchain	11,037	2,000 (50.9%)	3,628	921 (24.8%)
langchainjs	3,852	733 (18.6%)	3,921	950 (25.5%)
llama_index	11,627	634 (16.1%)	9,751	842 (22.6%)
transformers	13,580	173 (4.4%)	13,798	223 (6.0%)
openai-cookbook	537	160 (4.1%)	970	320 (8.6%)
chroma	1,359	127 (3.2%)	2,949	245 (6.6%)
azure-openai-samples	413	49 (1.2%)	1,717	125 (3.4%)
openai-python	654	20 (0.5%)	1,274	25 (0.7%)
azure-search-openai-demo	406	19 (0.5%)	1,052	36 (1.0%)
LangChain-nextjs-template	93	16 (0.4%)	105	32 (0.9%)
Total	**43,558**	**3,931**	**39,165**	**3,719**

Table 1: LangChain corpus statistics across 2024 and 2025 snapshots. The distribution of relevant documents across each GitHub repository is provided in %.

and Larson, 2025). Therefore, for each query in LangChain, we reuse the nuggets generated from FreshStack in our work (Thakur et al., 2025).

3.3 Oracle Retrieval

Retrieval Techniques. We follow FreshStack (Thakur et al., 2025) and utilize diverse and SoTA retrieval models. Our first model is BM25 (Robertson and Zaragoza, 2009), which retrieves lexically similar documents. Our next model is BGE (Gemma-2) (Chen et al., 2024), a dense retriever model with an embedding size of 3584 and an 8K context length. Next, E5 Mistral (7B) (Wang et al., 2024) is another dense retriever model with an embedding size of 4096. Furthermore, we employ Qwen3 (4B) (Zhang et al., 2025) which has an embedding size of 4096 and a 32K context length. Qwen3 embeddings achieve state-of-the-art accuracy in retrieval on various domains. Finally, we normalize the scores for each model between $[0, 1]$ and sum them up in hybrid fusion.[1]

Retrieval Settings. To further increase diversity in our judgment pools, we use different approaches to construct queries in the oracle setting. We utilize (1) sub-query generation, (2) nuggets, (3) gold answer, and (4) closed-book generated answer as different techniques to retrieve top 50 documents from each technique. We discuss each technique in depth in Appendix A.

3.4 Nugget-Level Assessment

Deciding whether a document is relevant to a given query has traditionally relied on human judges. However, this process can be expensive in terms of both cost and time. Recent studies have shown potential in utilizing LLMs as automatic judges for relevance judgments (Thomas et al., 2024; Rahmani et al., 2024; Farzi and Dietz, 2025; Upadhyay et al., 2025). IR test collections typically contain short queries, but query length itself is not a reliable proxy for task difficulty (Belkin et al., 2003; Wilkie and Azzopardi, 2015). Our motivation is instead to incorporate granularity: a verbose Stack Overflow question bundles several distinct information needs into a single post. We therefore judge relevance at the nugget level, assessing whether a document supports an individual key fact.

[1]Hybrid fusion is used *only* for oracle pooling to retrieve diverse categories of documents and not as an evaluated run.

Judgment	Full Support	Partial Support	No Support
Ph.D. Student	48 (80.0%)	5 (8.3%)	7 (11.7%)

Table 2: Human validation measuring Command A's support judgments on 60 randomly sampled nugget–document pairs.

We utilize **Cohere's Command A** as the judge, with 111B parameters and a 256K context length, providing strong performance rivaling GPT-4o on long-context reasoning tasks (Cohere, 2025). We provide Command A with the query and answer (for context), generated nuggets and a set of retrieved documents. Finally, we task it with identifying which documents are relevant and support a nugget following FreshStack (Thakur et al., 2025).

To validate the reliability of Command A's judgments, we conduct a human evaluation study with a Ph.D. student in machine learning on 60 randomly sampled relevant nugget–document pairs. Scores in Table 2 indicate 80% human agreement.

4 Experimental Results & Analysis

4.1 Temporal Support of Queries

In this section, we explore **RQ1**, whether existing queries continue to be grounded in a document corpus with temporal shifts. For example, *agents* used to be part of the LangChain repository in 2024. However, since 2025, the implementation of agentic systems has been moved to LangGraph (Wang and Duan, 2024). Therefore, queries made on the subject of agentic systems *would fail* to retrieve relevant documents from the 2025 snapshot.

Experimental Setup. To determine whether dynamic changes in the LangChain corpus limit the set of queries asked previously in time, we conduct the following experiment: If all nuggets for a query are supported by at least a single relevant document in the corpus, there is sufficient information in the corpus to ground the key facts necessary to construct the Stack Overflow answer.

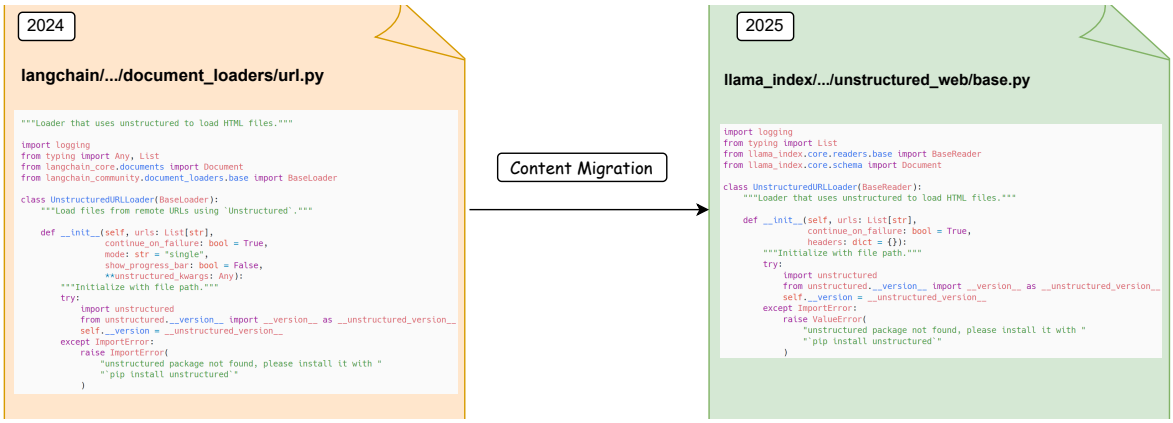

Figure 1: `UnstructuredURLLoader` class migrated for LangChain query `75864073` from LangChain (October 2024) and integrated into LlamaIndex (October 2025).

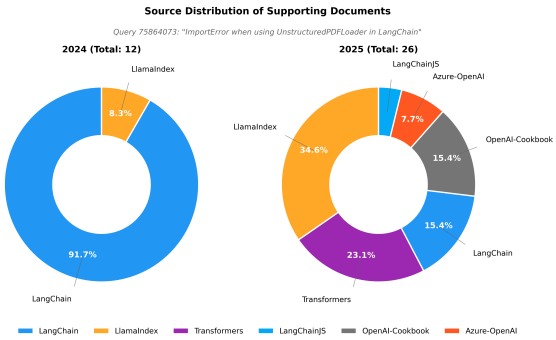

Figure 2: Source distribution shift for LangChain query `75864073` between October 2024 and October 2025.

Results. We run the experiment on both snapshots of our corpus, and observe that only 1 nugget is not supported in the 2025 corpus. We explore this nugget in Appendix B, where the reason is due to the deprecation of the FAISS vector store integration in LangChain, without a re-organization to other repositories. Therefore, for the remaining 639 out of 640 nuggets (across 203 queries), there exists at least one relevant document in the 2025 corpus to support the nugget information. As the single unsupported nugget belongs to one query, 202 of 203 queries (99.5%) remain fully grounded. We can thus conclude that the October 2025 corpus snapshot can ground older queries (until October 2024) and provide necessary relevant documents.

4.2 Corpus Temporal Changes

In this section, we explore **RQ2** by analyzing the differences in relevant documents across both snapshots. Previously, we outlined a potential scenario where implementation details (e.g., agents) migrate from one repository to another. Therefore, it is important to consider the impact of such shifts occurring within the selected set of repositories.

Experimental Setup & Results. We examine the distribution of documents assessed as relevant by Command A. Table 1 highlights the underlying distribution shift in relevant documents across GitHub repositories in both snapshots. First, we observe a reduction in relevant documents originating from the LangChain repository. In 2024, LangChain was the source of 50.9% of all relevant documents. However, in 2025, this drops to 24.8% (just below LangChainJS with 25.5%). Furthermore, in 2025, we observe greater diversity in the sources of relevant documents. This shift is consistent with prior work (Malviya et al., 2024), which characterizes LangChain as a modular ecosystem that relies on related repositories such as LlamaIndex, with an example in Figure 1.

Case Study on Query `75864073`. As a case study, we examine query `75864073` (title: *ImportError with UnstructuredPDFLoader in LangChain*). From the source distribution of relevant documents across both snapshots in Figure 2, we observe that the relevant documents increase from 12 to 26, demonstrating that related frameworks increasingly overlap in functionality, resulting in redundancy rather than deferring to the original source. In October 2024, LangChain is the primary source for relevant documents (91.7%). However, in October 2025, the relevant documents are spread across six repositories, with LlamaIndex (34.6%) being the largest source. As illustrated earlier in Figure 1, the `UnstructuredURLLoader` class, a relevant document for query 75864073, was migrated from LangChain to LlamaIndex.

We suspect two reasons for this observation: (1) LangChain reorganized its documentation and removed content about third-party integrations in

Retrieval Model	October 2024			October 2025		
	αN@10	C@20	R@50	αN@10	C@20	R@50
BM25	0.228	0.576	0.170	0.187	0.489	0.154
Qwen3 (8B)	**0.503**	**0.877**	0.432	0.480	0.868	0.424
Qwen3 (4B)	0.501	0.868	**0.436**	**0.482**	**0.873**	**0.437**
Qwen3 (0.6B)	0.442	0.821	0.362	0.411	0.782	0.372
Stella (1.5B)	0.444	0.824	0.374	0.430	0.830	0.371
Stella (400M)	0.386	0.812	0.340	0.388	0.805	0.345
Granite-Emb.-R2	0.396	0.767	0.318	0.359	0.766	0.325
Granite-Emb.-R2 (Small)	0.393	0.768	0.311	0.375	0.752	0.311
GTE (ModernBERT-base)	0.365	0.716	0.271	0.343	0.728	0.279
GTE (large) v1.5	0.324	0.709	0.239	0.276	0.683	0.242
E5 Mistral (7B)	0.464	0.851	0.395	0.434	0.828	0.388
BGE (Gemma-2)	0.423	0.794	0.341	0.401	0.828	0.348
Jina V4	0.408	0.811	0.381	0.417	0.821	0.379
Voyage-4-nano	0.441	0.811	0.399	0.426	0.824	0.396

Table 3: Document retrieval on LangChain using the 2024 and 2025 temporal snapshots. Metrics listed are α-nDCG@10, Coverage@20, and Recall@50.

2025, and (2) competing frameworks like LlamaIndex and Transformers expanded their documentation to cover similar processing workflows in 2025.

These observations reiterate the inherent complexity of the retrieval task on technical documentation. The retrieval system cannot simply find the exact same file across both corpus snapshots to retrieve relevant information. Due to temporal changes in the repository, a robust retrieval system must focus on the underlying document structure and content presented at runtime, despite the lack of temporal consistency.

4.3 Benchmark Analysis

In this section, we attempt to answer **RQ3**. We benchmark diverse embedding models across both snapshots and evaluate changes in model ranking.

Experimental Setup. In addition to the retrieval models used in Section 3.3, we benchmark models from the following series: Qwen3 embeddings (Zhang et al., 2025), Stella (Zhang and Wang, 2024), Granite-R2 (Awasthy et al., 2025), GTE (Li et al., 2023), Jina V4 (Günther et al., 2025), and Voyage-4-nano (VoyageAI, 2026). For evaluation, we follow FreshStack and use α-nDCG@10 (αN@10), Coverage@20 (C@20), and Recall@50 (R@50) to measure retrieval diversity and relevance. Metric definitions are listed in Thakur et al. (2025).

Results. From Table 3, we observe that Qwen3 (4B) and (8B) achieve the highest scores. The gap between Qwen3 (4B) and (8B) is small and metric-dependent. We suspect this because Qwen3 (4B) is also used as a pooling model, so it likely suffers less from unjudged documents. For α-nDCG@10 and Recall@50, we generally observe a decrease in scores from 2024 to the 2025 snapshot, likely due to relevant documents becoming more distributed (as shown in Table 1), increasing retrieval difficulty and reducing lexical alignment with LangChain documentation, hurting BM25.

Metric	α-nDCG@10	Coverage@20	Recall@50
Kendall τ	0.846	0.692	0.978

Table 4: Kendall τ correlation between model rankings across the October 2024 and 2025 temporal snapshots.

Model Ranking Correlation. We utilize the Kendall τ correlation metric to compare changes in model rankings between both temporal snapshots. From Table 4, we observe a very strong correlation, particularly for Recall@50 (Kendall τ = 0.978) and α-nDCG@10 (Kendall τ = 0.846). However, the correlation is relatively weaker for Coverage@20 (Kendall τ = 0.692). The positive correlation indicates that models across different temporal snapshots remain fairly consistent in their ability to retrieve relevant information, despite dynamic changes in the corpus.

5 Conclusion

In this work, we investigate how temporal corpus drift affects FreshStack on the LangChain subset. We first observe that all but one query (202 out of 203) continue to have supported documents (i.e., relevant documents) for each nugget in the October 2025 corpus snapshot. Next, we observe that functions and classes often migrate to other competitor frameworks, e.g., LangChain $\rightarrow$ LlamaIndex. Finally, we observe that model rankings across both temporal snapshots of the corpus are strongly correlated, indicating that retrieval systems are consistent in retrieving relevant documents despite changes in the corpus over time. These findings suggest that FreshStack (LangChain) is robust to dynamic changes in technical code documentation, where documents are *relocated* to other repositories rather than being invalidated. In future, we can motivate an end-to-end study linking retrieval drift to generated-code correctness.

Limitations

Our work is not perfect and has limitations. We describe our limitations below and leave extending this study to future work.

Limited Scope. Our work evaluates LangChain, one of five domains available in FreshStack, alongside Angular, Laravel, Godot, and Yolo (Thakur

et al., 2025). We pick LangChain as it represents the most rapidly evolving domain in FreshStack, with the highest degree of corpus restructuring between October 2024 and 2025, motivating the study of temporal drift. However, the extent to which our findings generalize to other domains, such as Laravel or Angular, remains an open question. Domains with more stable documentation or without competing repositories may exhibit different behaviors under temporal drift.

Fixed Queries and Answers. Our analysis holds the Stack Overflow queries and answers fixed from October 2024 while varying the document corpus across two temporal snapshots (October 2024 & 2025). However, Stack Overflow answers are themselves subject to temporal change: accepted answers may become outdated as APIs evolve, code examples may break with new library versions, and community-accepted solutions may shift over time. This introduces query and answer staleness that is distinct from corpus drift and is not addressed in this work. We focus exclusively on how a changing corpus affects existing retrieval benchmarks, and leave answer-level temporal drift for future work.

Fixed Repository Set. Our LangChain corpus is constructed from a fixed set of ten GitHub repositories selected at the time of the original FreshStack construction (Thakur et al., 2025). New frameworks or repositories that gained relevance to LangChain queries between October 2024 and October 2025 (e.g., newly emerged RAG toolkits or documentation sites) are not included in the October 2025 temporal snapshot.

Temporal Drift Measured as Code Migration. Our central finding that benchmarks remain reliable under temporal corpus drift is contingent on a property specific to open-source software ecosystems: relevant information is rarely removed but rather relocated across interdependent repositories. This *migration-without-loss* pattern is fundamentally different from temporal drift in domains where facts are superseded or invalidated entirely, such as economics and finance (where numerical values are continuously replaced), geopolitics (where territorial or political facts change), or legal domains (where statutes and rulings invalidate prior findings). In such settings, a corpus update does not merely redistribute relevant documents but introduces *factual staleness*, where previously relevant documents become actively misleading to retrieve.

Automatically Generated Relevance Judgments. A crucial aspect of our work involves using the Command-A model to decide whether the retrieved text is relevant to a specific nugget, ultimately determining if the retrieved text can help answer the query. Automating this process allows for easy experimentation, and its strong alignment with a Ph.D. student's judgments gives us a degree of certainty in the approach. However, the extent to which this alignment generalizes across all judgments can only be estimated, not conclusively proven. We acknowledge that human assessment provides essential validation in our setup and would help further support our conclusion.

Acknowledgments

This research was supported in part by the Natural Sciences and Engineering Research Council of Canada (NSERC).

References

Zahra Abbasiantaeb, Simon Lupart, Leif Azzopardi, Jeffrey Dalton, and Mohammad Aliannejadi. 2025. Conversational gold: Evaluating personalized conversational search system using gold nuggets. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '25, page 3455–3465, New York, NY, USA. Association for Computing Machinery.

Parul Awasthy, Aashka Trivedi, Yulong Li, Meet Doshi, Riyaz A. Bhat, Vignesh P, Vishwajeet Kumar, Yushu Yang, Bhavani Iyer, Abraham Daniels, Rudra Murthy, Ken Barker, Martin Franz, Madison Lee, Todd Ward, Salim Roukos, David Cox, Luis A. Lastras, Jaydeep Sen, and Radu Florian. 2025. Granite Embedding R2 Models. *CoRR*, abs/2508.21085.

N. J. Belkin, D. Kelly, G. Kim, J.-Y. Kim, H.-J. Lee, G. Muresan, M.-C. Tang, X.-J. Yuan, and C. Cool. 2003. Query length in interactive information retrieval. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*, SIGIR '03, page 205–212, New York, NY, USA. Association for Computing Machinery.

Matteo Cancellieri, Alaa El-Ebshihy, Tobias Fink, Petra Galuščáková, Gabriela Gonzalez-Saez, Lorraine Goeuriot, David Iommi, Jüri Keller, Petr Knoth, Philippe Mulhem, Florina Piroi, David Pride, and Philipp Schaer. 2025. LongEval at CLEF 2025: Longitudinal Evaluation of IR Model Performance. In *Advances in Information Retrieval: 47th European Conference on Information Retrieval, ECIR 2025, Lucca, Italy, April 6–10, 2025, Proceedings, Part V*, page 382–388, Berlin, Heidelberg. Springer-Verlag.

Ben Carterette, Evgeniy Gabrilovich, Vanja Josifovski, and Donald Metzler. 2010. Measuring the reusability of test collections. In *Proceedings of the Third ACM International Conference on Web Search and Data Mining*, WSDM '10, page 231–240, New York, NY, USA. Association for Computing Machinery.

Jianlyu Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024. M3-embedding: Multi-linguality, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 2318–2335, Bangkok, Thailand. Association for Computational Linguistics.

Cohere. 2025. Command A: An Enterprise-Ready Large Language Model. *CoRR*, abs/2504.00698.

Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Jimmy Lin. 2022. Overview of the TREC 2021 Deep Learning Track. In *Text REtrieval Conference (TREC)*. NIST, TREC.

Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, Jimmy Lin, Ellen M. Voorhees, and Ian Soboroff. 2023. Overview of the TREC 2022 Deep Learning Track. In *Text REtrieval Conference (TREC)*. NIST, TREC.

Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, Ellen M. Voorhees, and Ian Soboroff. 2021. TREC Deep Learning Track: Reusable Test Collections in the Large Data Regime. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, page 2369–2375, New York, NY, USA. Association for Computing Machinery.

Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Hossein A. Rahmani, Daniel Campos, Jimmy Lin, Ellen M. Voorhees, and Ian Soboroff. 2024. Overview of the TREC 2023 Deep Learning Track. In *Text REtrieval Conference (TREC)*. NIST, TREC.

Alaa El-Ebshihy, Tobias Fink, Gabriela Gonzalez-Saez, Petra Galuščáková, Florina Piroi, David Iommi, Lorraine Goeuriot, and Philippe Mulhem. 2023. Predicting Retrieval Performance Changes in Evolving Evaluation Environments. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, pages 21–33, Cham. Springer Nature Switzerland.

Naghmeh Farzi and Laura Dietz. 2025. Criteria-Based LLM Relevance Judgments. In *Proceedings of the 2025 International ACM SIGIR Conference on Innovative Concepts and Theories in Information Retrieval (ICTIR)*, ICTIR '25, page 254–263, New York, NY, USA. Association for Computing Machinery.

Petra Galuscáková, Romain Deveaud, Gabriela González Sáez, Philippe Mulhem, Lorraine Goeuriot, Florina Piroi, and Martin Popel. 2023. LongEval-Retrieval: French-English Dynamic Test Collection for Continuous Web Search Evaluation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '23, page 3086–3094, New York, NY, USA. Association for Computing Machinery.

Michael Günther, Saba Sturua, Mohammad Kalim Akram, Isabelle Mohr, Andrei Ungureanu, Bo Wang, Sedigheh Eslami, Scott Martens, Maximilian Werk, Nan Wang, and Han Xiao. 2025. jina-embeddings-v4: Universal Embeddings for Multimodal Multilingual Retrieval. *CoRR*, abs/2506.18902.

Jüri Keller, Timo Breuer, and Philipp Schaer. 2024a. Evaluation of Temporal Change in IR Test Collections. In *Proceedings of the 2024 ACM SIGIR International Conference on Theory of Information Retrieval*, ICTIR '24, page 3–13, New York, NY, USA. Association for Computing Machinery.

Jüri Keller, Timo Breuer, and Philipp Schaer. 2024b. Replicability Measures for Longitudinal Information Retrieval Evaluation. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction: 15th International Conference of the CLEF Association, CLEF 2024, Grenoble, France, September 9–12, 2024, Proceedings, Part I*, page 215–226, Berlin, Heidelberg. Springer-Verlag.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. 2023. Towards general text embeddings with multi-stage contrastive learning. *CoRR*, abs/2308.03281.

Jimmy Lin and Dina Demner-Fushman. 2005. Automatically evaluating answers to definition questions. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 931–938, Vancouver, British Columbia, Canada. Association for Computational Linguistics.

Jimmy Lin and Dina Demner-Fushman. 2006. Methods for automatically evaluating answers to complex questions. *Inf. Retr.*, 9(5):565–587.

Rajesh Kumar Malviya, Vishal Javalkar, and Rano Malviya. 2024. Scalability and Performance Benchmarking of LangChain, LlamaIndex, and Haystack for Enterprise AI Customer Support Systems. In *IJGIS Fall of 2024 Conference*. The New World Foundation.

Dasha Metropolitansky and Jonathan Larson. 2025. Towards effective extraction and evaluation of factual claims. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6996–7045, Vienna, Austria. Association for Computational Linguistics.

Diganta Misra, Nizar Islah, Victor May, Brice Rauby, Zihan Wang, Justine Gehring, Antonio Orvieto, Muawiz Sajjad Chaudhary, Eilif B. Muller, Irina Rish, Samira Ebrahimi Kahou, and Massimo Caccia. 2026. GitChameleon 2.0: Evaluating AI code generation against Python library version incompatibilities. In

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 46792–46831, San Diego, California, United States. Association for Computational Linguistics.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A human generated machine reading comprehension dataset. In *Proceedings of the Workshop on Cognitive Computation: Integrating neural and symbolic approaches 2016 co-located with the 30th Annual Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain, December 9, 2016*, volume 1773 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Andrew Parry, Maik Fröbe, Harrisen Scells, Ferdinand Schlatt, Guglielmo Faggioli, Saber Zerhoudi, Sean MacAvaney, and Eugene Yang. 2025. Variations in relevance judgments and the shelf life of test collections. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '25, page 3387–3397, New York, NY, USA. Association for Computing Machinery.

Ronak Pradeep, Nandan Thakur, Shivani Upadhyay, Daniel Campos, Nick Craswell, Ian Soboroff, Hoa Trang Dang, and Jimmy Lin. 2025. The great nugget recall: Automating fact extraction and RAG evaluation with large language models. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '25, page 180–190, New York, NY, USA. Association for Computing Machinery.

Hossein A. Rahmani, Emine Yilmaz, Nick Craswell, Bhaskar Mitra, Paul Thomas, Charles L. A. Clarke, Mohammad Aliannejadi, Clemencia Siro, and Guglielmo Faggioli. 2024. LLMJudge: LLMs for Relevance Judgments. In *Proceedings of The First Workshop on Large Language Models for Evaluation in Information Retrieval (LLM4Eval 2024) co-located with 10th International Conference on Online Publishing (SIGIR 2024), Washington D.C., USA, July 18, 2024*, volume 3752 of *CEUR Workshop Proceedings*, pages 1–3. CEUR-WS.org.

Stephen Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389.

Ian Soboroff. 2006. Dynamic test collections: measuring search effectiveness on the live web. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '06, page 276–283, New York, NY, USA. Association for Computing Machinery.

Nandan Thakur, Jimmy Lin, Sam Havens, Michael Carbin, Omar Khattab, and Andrew Drozdov. 2025. FreshStack: Building Realistic Benchmarks for Evaluating Retrieval on Technical Documents. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. BEIR: A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*.

Paul Thomas, Seth Spielman, Nick Craswell, and Bhaskar Mitra. 2024. Large language models can accurately predict searcher preferences. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '24, page 1930–1940, New York, NY, USA. Association for Computing Machinery.

Shivani Upadhyay, Ronak Pradeep, Nandan Thakur, Daniel Campos, Nick Craswell, Ian Soboroff, and Jimmy Lin. 2025. A large-scale study of relevance assessments with large language models using UMBRELA. In *Proceedings of the 2025 International ACM SIGIR Conference on Innovative Concepts and Theories in Information Retrieval (ICTIR)*, ICTIR '25, page 358–368, New York, NY, USA. Association for Computing Machinery.

Ellen Voorhees. 2009. I Come Not To Bury Cranfield, but to Praise It. HCIR 2009: Bridging Human Computer Interaction and Information Retrieval, Washington DC, USA.

Ellen M. Voorhees. 2018. On building fair and reusable test collections using bandit techniques. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, CIKM '18, page 407–416, New York, NY, USA. Association for Computing Machinery.

Ellen M. Voorhees, Ian Soboroff, and Jimmy Lin. 2022. Can Old TREC Collections Reliably Evaluate Modern Neural Retrieval Models? *CoRR*, abs/2201.11086.

Ellen M. Voorhees and Dawn M. Tice. 2000. Building a question answering test collection. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '00, page 200–207, New York, NY, USA. Association for Computing Machinery.

VoyageAI. 2026. The Voyage 4 model family: shared embedding space with MoE architecture.

Jialin Wang and Zhihua Duan. 2024. Intelligent Spark Agents: A Modular LangGraph Framework for Scalable, Visualized, and Enhanced Big Data Machine Learning Workflows. *CoRR*, abs/2412.01490.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024. Improving text embeddings with large language models. In

Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024, pages 11897–11916. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

Colin Wilkie and Leif Azzopardi. 2015. Query length, retrievability bias and performance. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, CIKM '15, page 1787–1790, New York, NY, USA. Association for Computing Machinery.

Dun Zhang and Fulong Wang. 2024. Jasper and Stella: distillation of SOTA embedding models. *CoRR*, abs/2412.19048.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. 2025. Qwen3 embedding: Advancing text embedding and reranking through foundation models. *CoRR*, abs/2506.05176.

Justin Zobel. 1998. How reliable are the results of large-scale information retrieval experiments? In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, page 307–314, New York, NY, USA. Association for Computing Machinery.

A Retrieval Settings in Oracle Retrieval

To retrieve potentially relevant documents, we follow FreshStack and try to retrieve diverse documents using four different retrieval settings: (1) We use the gold Stack Overflow answer as the query to retrieve relevant documents. (2) We use the generated nuggets by GPT-4o, which contain concise and structured information. (3) We generate sub-queries by decomposing the original verbose Stack Overflow query into shorter queries, enabling effective retrieval through chain-of-thought prompting (Wei et al., 2022) using the Qwen3-4B-Instruct model.[2] Qwen3-4B-Instruct is a compact, open-weight model with strong instruction-following and chain-of-thought reasoning capabilities (Zhang et al., 2025), making it a cost-effective choice for sub-question decomposition compared to closed-source alternatives like GPT-4o. (4) We use Qwen3-4B-Instruct to generate the closed-book answer to the Stack Overflow query. Finally, we collect the top 50 documents using fusion for each technique described above, and pass them along to the next stage for nugget-level support assessment.

B Nugget with No Relevant Documents

During our evaluation, we identify only a single nugget (out of 640 nuggets), belonging to query `76400798` (title: *Langchain cannot create index when running inside Django server*) that has zero supporting documents in the October 2025 snapshot. The nugget contains the following text: "*The issue occurs when trying to create a FAISS index within a Django server using Langchain.*" The nugget has two relevant documents in the October 2024 snapshot, both of which are official LangChain documentation notebooks covering synchronous and asynchronous FAISS vector store integration (‘langchain/.../vectorstores/faiss.ipynb‘ & ‘langchain/.../vectorstores/faiss_async.ipynb‘).

Between October 2024 and October 2025, these notebooks were removed from the LangChain documentation as part of LangChain’s broader reorganization of third-party vector store integrations. Crucially, unlike other deprecated components such as agents (which migrated to LangGraph) or loaders (which migrated to LlamaIndex), the FAISS integration documentation was not reproduced or mirrored in any of the other repositories in our corpus. This represents a true deprecation without re-organization: FAISS, while still usable programmatically, lost its dedicated documentation coverage within the LangChain ecosystem by October 2025, leaving the only Stack Overflow nugget unsupported, out of 640 nuggets.

[2] Qwen3-4B-Instruct: `https://huggingface.co/Qwen/Qwen3-4B-Instruct-2507`

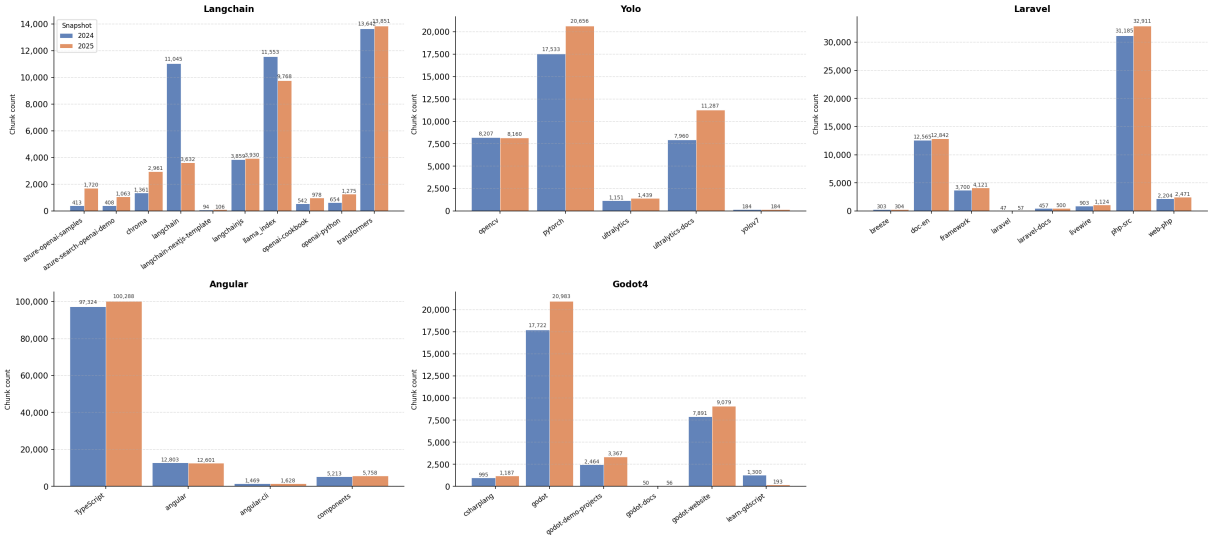

Figure 3: Change in source documents across GitHub repositories between October 2024 (blue) and October 2025 (orange) across all five domains (LangChain, Yolo, Laravel, Angular, and Godot) in FreshStack. The maximum temporal drift or change in document counts is observed in the LangChain domain.

Retrieval Model	**Model URL**
BM25	https://github.com/castorini/anserini
Qwen3 (8B)	https://huggingface.co/Qwen/Qwen3-Embedding-8B
Qwen3 (4B)	https://huggingface.co/Qwen/Qwen3-Embedding-4B
Qwen3 (0.6B)	https://huggingface.co/Qwen/Qwen3-Embedding-0.6B
Stella (1.5B)	https://huggingface.co/NovaSearch/stella_en_1.5B_v5
Stella (400M)	https://huggingface.co/NovaSearch/stella_en_400M_v5
Granite-Emb.-R2	https://huggingface.co/ibm-granite/granite-embedding-english-r2
Granite-Emb.-R2 (Small)	https://huggingface.co/ibm-granite/granite-embedding-small-english-r2
GTE (ModernBERT-base)	https://huggingface.co/Alibaba-NLP/gte-modernbert-base
GTE (large) v1.5	https://huggingface.co/Alibaba-NLP/gte-large-en-v1.5
BGE (Gemma-2)	https://huggingface.co/BAAI/bge-multilingual-gemma2
E5 Mistral 7B	https://huggingface.co/intfloat/e5-mistral-7b-instruct
Jina V4	https://huggingface.co/jinaai/jina-embeddings-v4
Voyage-4-nano	https://huggingface.co/voyageai/voyage-4-nano

Table 5: Retrieval models used in our evaluation, along with URLs to their corresponding model repositories.