

RAD-PROTEIN: Multi-source Knowledge-enriched Retrieval-augmented Protein Structure Searching

Kexin Ding^{1,4,5}, Alison Biester^{1,3}, Brinda Vallat^{1,3}, Chenghua Shao^{1,3},
Dennis W. Piehl^{1,3}, Dongwook Won¹, Joan Segura^{2,3}, Jose M. Duarte^{2,3}, Mu Zhou¹,
Zukang Feng^{1,3}, Yana Rose^{2,3}, Stephen K. Burley^{1,2,3,5}, Dimitris N. Metaxas^{1,4,5}

¹Rutgers, the State University of New Jersey, ²University of California San Diego,

³Research Collaboratory for Structural Bioinformatics Protein Data Bank

⁴Center for Computational Biomedicine, Imaging and Modeling (CBIM)

⁵Rutgers Artificial Intelligence and Data Science (RAD) Collaboratory

Correspondence: dnm@cs.rutgers.edu, sburley@proteomics.rutgers.edu

Abstract

The Research Collaboratory for Structural Bioinformatics Protein Data Bank (RCSB PDB) provides open access to macromolecular structures and offers an extensive structural keyword search infrastructure. However, the current RCSB PDB APIs do not support natural-language semantic search. To overcome this challenge, we propose RAD-PROTEIN, a novel multi-source knowledge-enriched **Retrieval-Augmented Protein** structure search system that answers user natural-language queries by jointly leveraging heterogeneous structural literature and metadata. In addition, we introduce the first benchmark for text-based protein-structure search evaluation, comprising 454 expert-verified queries. RAD-PROTEIN substantially outperforms baselines, achieving up to +19.8% Recall@1 and +16.1% in MRR with hybrid retrieval and multi-source knowledge enrichment. Comprehensive ablation studies further confirm the robustness of our proposed approach. Finally, we develop an interpretable interface to return top-ranked structures with LLM-generated explanations, enabling efficient human verification and downstream scientific analysis. Code: https://github.com/Cassie07/RAD_PROTEIN.

1 Introduction

Driven by rapid advances in structural biology, the Protein Data Bank (PDB) (Berman et al., 2000) has become the largest repository, containing over 254,000 experimentally determined macromolecular structures. The Research Collaboratory for Structural Bioinformatics Protein Data Bank (RCSB PDB) portal enriches these structures with comprehensive biological annotations, including protein function, ligand information, and organism classification, enabling researchers to explore structures within their biological context. This wealth of information has made the RCSB PDB indispensable for drug discovery, enzyme engineering, and disease understanding (Burley et al., 2023).

Despite these resources, protein structure searching remains a significant challenge for biologists. The advanced RCSB PDB search interface (Rose et al., 2020) supports lexical keyword queries over structure similarity, sequence motifs, and chemical descriptors, all of which require users to formulate structured database queries. Consequently, users face three distinct burdens. First, *semantic mismatch*, where the search API cannot process natural-language scientific questions (e.g., "Can you find hemoglobin structures of high-altitude birds?"), forcing manual translation to keyword syntax; Second, *learning load*, as users must learn the API's keyword definitions before conducting a search, significantly reducing the efficiency; Finally, *verification burden*, as search results are not automatically filtered by biological context, requiring researchers to manually validate relevance across heterogeneous data sources.

To address these challenges, we propose RAD-PROTEIN, a multi-source, knowledge-enriched retrieval-augmented protein search workflow supporting natural-language queries by leveraging the capabilities of large language model (LLM). Our major contributions are: (1) **A multi-source knowledge-enriched retrieval-augmented protein searching workflow** enabling semantic protein-structure search via plain-text queries using heterogeneous protein contexts. (2) **The first expert-verified benchmark for text-based protein-structure search evaluation**. This benchmark contains 454 LLM-generated queries that were authentically verified by eight human experts. (3) **An interpretable, user-friendly interface** that presents LLM-generated rationales and judgments, simplifying result understanding and verification.

2 Related Work

Protein Structure Search and Resources. The Protein Data Bank (PDB) (Berman et al., 2000) is

the primary archive that contains large-scale macromolecular structures, of which the majority are protein structures. The US-based Research Collaboratory for Structural Bioinformatics PDB (Burley et al., 2023) allows users to access and explore these structures through the RCSB.org web portal and APIs (Piehl et al., 2025; Rose et al., 2020), while integrating external annotations to enrich structural metadata. Although researchers frequently identify structures relevant to complex scientific questions, the RCSB search API (Rose et al., 2020) operates on a fixed schema that requires users to manually translate natural-language questions into structured keyword queries.

Retrieval-augmented Generation. Retrieval-augmented generation (RAG) (Lewis et al., 2020; Gao et al., 2024) retrieves external relevant documents as additional context for LLM answer generation, reducing hallucination and enhancing knowledge discovery. This strategy has shifted from sparse BM25 (Robertson and Zaragoza, 2009) matching to dense retrieval with bi-encoder transformers (Izacard et al., 2022). The hybrid retrievers that combine the above two further improve the performance on out-of-domain corpora (Ma et al., 2021; Thakur et al., 2021). In biomedicine, RAG methods have been applied across a range of answer generation tasks (Chen et al., 2025; Xiong et al., 2024; Ateia and Kruschwitz, 2025; Jin et al., 2024) and multi-source planning (Chen et al., 2025). However, these approaches focus on synthesizing textual answers from retrieved passages, overlooking the distinct demand of retrieving individual protein structures. To respond, a core contribution of our study is a dual-granularity knowledge enrichment that retrieves and ranks protein-structured entities by leveraging both literature and diverse protein knowledge sources.

LLM-as-a-Judge. LLM-as-a-Judge uses a strong LLM to score the relevance, helpfulness, or factuality of the provided text that can reduce costly manual annotation (Ouyang et al., 2022; Zheng et al., 2023; Liu et al., 2023). Rule-based prompting embeds human-written evaluation criteria and rubrics into the instructions to produce a reliable score, rationale, and judgment for the provided text (Li et al., 2025; Wei et al., 2022). For instance, scoring each candidate using a multi-level rubric has been shown to produce finer-grained scores that better align with human judgments (Zhuang et al., 2024). In contrast, we

adopt LLM-as-a-Judge to strongly improve the interpretability of search results by producing relevance rationale, judgment, and relevance score, enhancing semantic understanding between a searched protein structure and the query.

3 Methodology

Overview. We propose RAD-PROTEIN, as shown in Figure 1, a multi-source knowledge-enriched retrieval-augmented protein structure searching pipeline that integrates evidence from PubMed literature and the RCSB.org portal to support text-based protein structure search (Section 3.1-3.3). Given a natural-language query, the pipeline first leverages a hybrid retriever to select an initial set of candidates. Then, a multi-source knowledge-enhanced reranking module refines relevance estimation by using protein-specific context. To improve interpretability, we developed a user interface that presents top-ranked search results with concise rationales and judgments (Section 3.4).

3.1 Multi-source Protein Knowledge

We include diverse knowledge sources that jointly describe each protein from textual, structural, functional, and chemical perspectives: **(1) PubMed abstracts.** Each protein entry is linked to its abstract, which is published on the PubMed platform with a unique PubMed ID. Abstracts provide the high-level scientific context for a structured biological motivation, method, and key findings. **(2) Structural information and chemical components.** For each entry, we query the RCSB PDB Data API (Piehl et al., 2025) to obtain its protein structure title, free-text description, and PDBx keywords, which capture per-entry attributes, such as the specific ligand bound or organism studied. For each non-polymer ligand or cofactor associated with protein entry, we extract its chemical component identifier and the corresponding name. **(3) UniProt descriptions.** For protein entries, we include their corresponding UniProt curated functional annotation (Consortium, 2023), providing the description of biological roles, catalytic activities, and subcellular localizations.

3.2 Offline Chunking and Indexing

In Step 1(a) of Figure 1, the goal is to produce a vectorstore over abstract chunks and cache all necessary knowledge from multiple sources and avoid repeated API calls to the RCSB-PDB. For

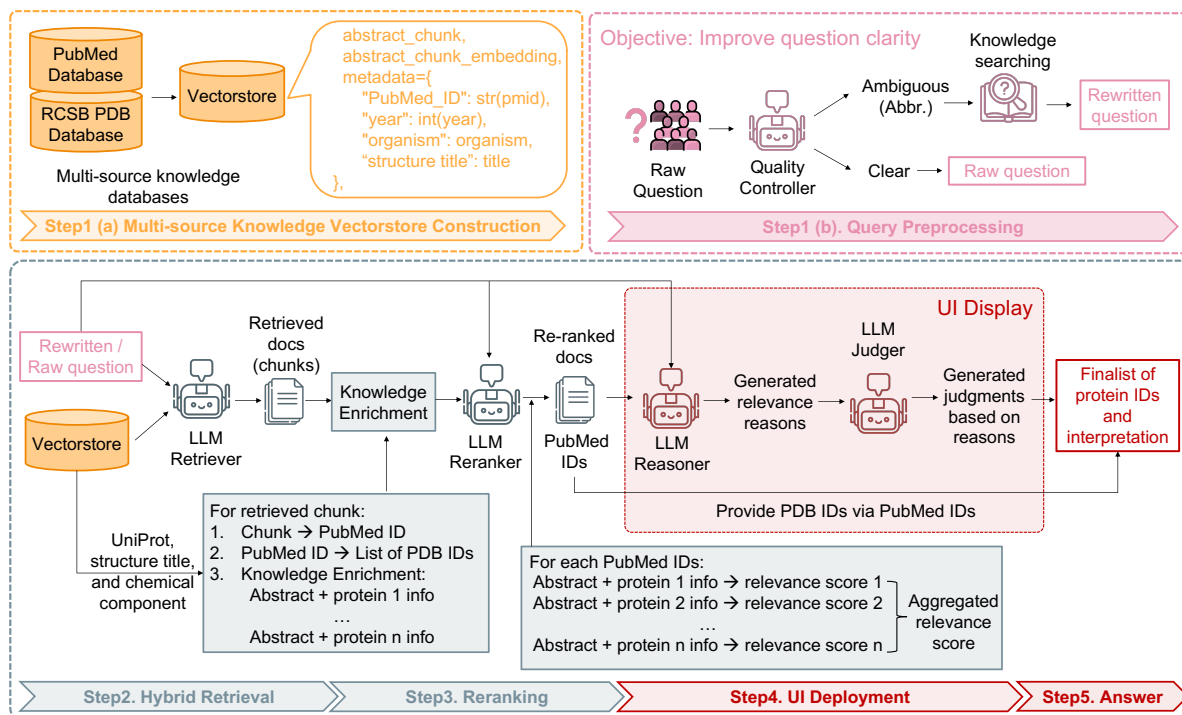


Figure 1: **Overview of RAD-PROTEIN.** First, we implemented multi-source knowledge vectorstore construction (Section 3.1-3.2) and query preprocessing (Section 3.3). We then proposed a knowledge-enriched retrieval-augmented protein structure search workflow together with an interpretable user interface (Section 3.3-3.4).

retrieval efficiency, each abstract is split into overlapping chunks using a recursive character splitter. To retain semantic continuity among chunks, we ensure there is overlapping content between them. Each chunk is stored with its corresponding meta-data and multi-source knowledge (e.g., PubMed ID, year, organism, structure title, chemical components information, UniProt descriptions, and the list of co-deposited PDB IDs). Next, chunks are embedded with a BGE bi-encoder (Xiao et al., 2024) loaded via Sentence-Transformers (Reimers and Gurevych, 2019) and stored in a Chroma vector database (Chroma, 2026). The same chunks are tokenized for the BM25 sparse index (Robertson and Zaragoza, 2009) using the LangChain default tokenizer (LangChain Community, 2026).

3.3 Knowledge-enriched Retrieval-augmented Protein Structure Search

The proposed multi-source knowledge-enriched retrieval-augmented protein structure search pipeline includes query preprocessing, hybrid retrieval, and knowledge-enhanced rerank.

Query Preprocessing. The most common quality issue in real-world user queries is the use of protein-related abbreviations, which introduces ambiguous searching signals. We address this with a

lightweight dictionary-based rewrite to implement query preprocessing towards enhanced clarity of the raw data (Step 1(b) of Figure 1). The abbreviation dictionary is built directly from the PubMed abstracts by extracting (abbreviation, full name) pairs. After deduplication, the dictionary contains 68,000 unique abbreviations, where 50,000 map to a single full name and 18,000 map to two or more. Then, we utilized a query controller to determine whether to rewrite the query. If no abbreviation is found, or if a matched abbreviation has multiple candidate full names, the query is retained as unchanged. If an abbreviation with a unique full name is found, we expand the query by using the full name while keeping the original abbreviation.

Hybrid Retrieval The goal of the initial retrieval is to explore a small set of candidates that potentially contain the answer, prioritizing the recall so that no relevant article is filtered out. We feed the same chunked abstracts into a dense and a sparse retriever in parallel and fuse their ranked lists. Such fusion simultaneously exploits lexical overlap and semantic similarity between query and candidate chunks. To assess the semantic similarity between the query and chunks, dense retrieval uses the same BGE bi-encoder (Xiao et al., 2024) as in the index-

ing phase to encode the query and each chunk independently into fixed-dimensional vectors, which are then used to calculate cosine similarity. Meanwhile, we use BM25 (Robertson and Zaragoza, 2009) as a sparse retrieval that matches the query to abstract chunks via exact keyword overlaps to determine the lexical overlap. Finally, the weighted dense and sparse scores are aggregated to obtain the top-100 chunks, which are then passed to the next stage of reranking.

Multi-source Knowledge-enhanced Reranking.

The goal of reranking is to reorder the retrieved candidates so that the structures most relevant to the query rise to the top. In RAD-PROTEIN, the reranker scores candidates at the protein-entry level rather than directly on the article level (i.e., PubMed ID). The relevance between the query and each co-deposited PDB structure is determined separately, and the per-entry relevance scores are then aggregated to the article level. Concretely, for each retrieved PubMed ID with abstract a and co-deposited PDB entries $\{p_1, \dots, p_m\}$, we construct m knowledge-enriched candidates. The i -th candidate concatenates the query with the multi-source knowledge of entry p_i , assembled with explicit field markers as [Query] q , [Abstract] a , [Structure info] $\text{title}_i; \text{kw}_i; \text{pdbx_kw}_i$, [UniProt] u_i , and [Chemical components] c_i . These markers ensure that the reranker distinguishes the query from each knowledge source and attends to them independently. The reranker thus jointly encodes the query and context of each protein entry to produce an entry-level relevance score. The article-level score is then obtained by aggregating over the co-deposited entries using max aggregation.

3.4 User Interface.

UI Development We developed a UI through a Gradio web application that contains a single-question-answer and batch-submission panel. A Cloudflare Tunnel (Cloudflare, 2026) is launched to provide a public HTTPS endpoint without requiring account configuration. The batch submission accepts a list of questions along with an email address, runs the pipeline in the background, persists per-question entries to JSON, and emails the result file when the run completes.

LLM Reasoning and Judgment. We leverage LLM-as-A-Judge to provide more interpretable search results in the UI, including the retrieved

PubMed IDs, relevance scores, concise relevance reason and judgment. Each PubMed ID is sorted by descending S_i with ties broken by retrieval order. The top-10 reranked PDB IDs are passed to the LLM to produce an interpretable output. We also show the relevance score per PDB entry to guide accurate structure relevance to the user. For reasoning, given the query and the searched structure together with enriched knowledge context, an LLM reasoner first generates a concise text rationale explaining why the candidate addresses the query, conditioned on a chain-of-thought-style prompt (Wei et al., 2022) as shown in Appendix 11. For the judgment, the same LLM further judges the rationale to validate the rationale, using the underlying evidence, and assigns a discrete relevance score in $\{0, 1, 2\}$ based on the prompt (Appendix 12). Both the rationale and the score are displayed in the UI, so researchers can quickly see why a structure was returned and decide whether further verification is needed. The judgment is a concise summary of the relevance based on the rationale.

4 Text-based Protein Structure Searching Benchmark

4.1 Data Source and Preprocessing

The evaluation benchmark is built on two public resources: the RCSB.org (Berman et al., 2000; Burley et al., 2023) and PubMed (Canese and Weis, 2013) databases. From every released PDB entry from the RCSB.org web portal, we extract the citation PubMed ID(s), the structure title, and PDBx keywords, the UniProt descriptions, and the chemical component information. From PubMed, we extract the title and abstract of each cited article. We then apply two preprocessing steps before generating any question. First, we drop PubMed IDs with missing abstracts. Then, we stratify the remaining PubMed IDs by category and by the number of co-deposited PDB entries, and apply category-balanced sampling to avoid a single category dominating the query pool. Each PubMed ID could be linked to one or more PDB entries.

4.2 Dataset Construction

We construct the benchmark illustrated in Figure 2. The 454 queries are generated by an LLM from a single abstract, producing a unique mapping between each question and its ground-truth PubMed ID, along with the corresponding abstract and PDB IDs. For queries, we group them into 11 super-

categories, which shows broad coverage across protein space (Appendix 5); Meanwhile, we further classified each query by 9 types of search intent (Appendix 6). To construct, first, we prompt an LLM with a question generation prompt (Appendix 9) that conditions on the abstract and asks for five candidate questions that can be answered from the information in the abstract and are phrased the way a structural biologist might query the PDB. The five candidate questions together with the abstract and a separate filter prompt (Appendix 10) are then passed to the same LLM, which determines the best query candidate that is most specific to the abstract with the least redundancy.

Human-expert Annotation. We recruited eight experts with extensive experience in biology to annotate the generated benchmark. The annotation aims to determine the quality of the generated question and its relevance to the source abstract. Each expert needs to rate on four criteria by being shown the question, the source abstract, and the linked PDB IDs. Each annotation criterion should be rated on a scale of 1 to 5. First, experts assessed the **quality** of the generated question by evaluating whether it was well written. Second, they evaluated the **relevance** of the question to the source abstract by determining whether the abstract could answer the question. Only questions rated high on both quality and relevance are retained in the final benchmark. Third, experts estimated the **difficulty** of the question, indicating how challenging it would be for a human expert to answer. Fourth, they reported the **confidence** level for annotating the question–abstract pair. The annotation interface allows experts to submit revised questions. Any revised question and abstract pair will be included in the final benchmark, even if the originally LLM-generated question is determined as low quality. The annotation statistics details are in Appendix 5.

5 Experiments and Results

5.1 Implementation Setting

Our implementation comprises retrieval, reranking, reasoning, and judgment modules based on the LangChain framework (LangChain Community, 2026). We use BAAI/BGE-SMALL-EN-V1.5 (Xiao et al., 2024) as the dense retriever, while the BM25 sparse retriever uses the LangChain default tokenizer. The hybrid weights for dense and sparse retrieval results aggregation are (0.7, 0.3). We

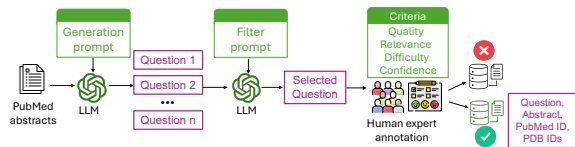


Figure 2: **Benchmark generation.** We developed an LLM-enabled workflow for generating a text-based protein search evaluation dataset with expert verification.

used BAAI/BGE-RERANKER-LARGE as the cross-encoder for reranking, and used Qwen-2.5-7B-Instruct (Team, 2024; Yang et al., 2024) to construct our benchmark and produce rationale and judgment to reveal the relevance reason between the searched results and the query. Regarding operational complexity and response time, each query takes approximately 50–130 seconds to complete the full search across all candidates and generate relevance reasons and judgments for the top 10 candidates, which are displayed in our UI.

We evaluate RAD-PROTEIN on the developed benchmark of 454 queries constructed in Section 4. The ground truth for each query is the PubMed ID, which can serve as a bridge linking user queries to the corresponding PDB IDs (i.e., protein structures). Its corresponding protein structures (i.e., PDB IDs) are provided in the UI as suggestions to accelerate users’ protein structure search. We report three evaluation metrics at the PubMed ID level to assess the search performance. Recall@ k is the fraction of queries whose top k results contain at least one relevant PubMed ID. nDCG@ k (Thakur et al., 2021) measures the rank position of relevant items within the top k , as relevant items appearing at higher ranks contribute more to the score. We report both at $k \in \{1, 5, 10, 20\}$. Mean Reciprocal Rank (MRR) measures how quickly the first relevant item is reached. We report 95% confidence intervals from bootstrap to quantify the uncertainty of each metric and confirm the reliability of the reported results.

5.2 Main Results

We report results on the developed human-verified benchmark ($N = 454$). The search results in the UI, together with the relevance rationale and the judgment between the searched structures and the query, are shown in Appendix 6–Appendix 8. In Table 1, We mainly assess two key components: (i) the contribution of each module in the workflow, by comparing hybrid retrieval alone (row 4) against retrieval combined with cross-encoder

Row	Module	Knowledge	Recall@k (%)				nDCG@k (%)				MRR (%)
			1	5	10	20	1	5	10	20	
1	BM25	A	42.29 [37.67, 46.70]	62.56 [58.14, 66.96]	70.26 [66.08, 74.45]	75.99 [72.03, 79.96]	42.29 [37.67, 46.70]	53.42 [49.30, 57.44]	55.91 [51.98, 59.76]	57.37 [53.56, 61.06]	52.05 [48.03, 55.95]
2	BGE Large	A	53.08 [48.46, 57.71]	71.15 [66.96, 75.33]	76.21 [72.25, 79.96]	80.84 [77.09, 84.36]	53.08 [48.46, 57.71]	62.77 [58.69, 66.75]	64.44 [60.55, 68.28]	65.62 [61.90, 69.33]	61.17 [57.19, 65.13]
3	BGE M3	A	38.77 [34.36, 43.39]	54.85 [50.22, 59.25]	60.13 [55.73, 64.54]	64.32 [59.91, 68.72]	38.77 [34.36, 43.39]	47.21 [42.96, 51.42]	48.93 [44.81, 53.01]	49.98 [45.98, 53.99]	45.83 [41.69, 49.99]
4	Hybrid Retrieval	A	53.74 [49.12, 58.37]	74.23 [70.26, 78.19]	80.18 [76.43, 83.92]	85.68 [82.38, 88.77]	53.74 [49.12, 58.37]	64.75 [60.87, 68.60]	66.70 [63.03, 70.34]	68.11 [64.64, 71.55]	62.91 [59.08, 66.73]
5	+ Rerank	A	62.78 [58.37, 67.18]	82.38 [78.85, 85.90]	86.56 [83.26, 89.65]	89.21 [86.12, 91.86]	62.78 [58.37, 67.18]	73.45 [69.92, 77.01]	74.82 [71.52, 78.18]	75.49 [72.25, 78.71]	71.29 [67.72, 74.91]
6		+ S	<u>73.35</u> [69.16, 77.31]	<u>85.46</u> [82.16, 88.55]	88.33 [85.24, 91.19]	90.09 [87.22, 92.73]	<u>73.35</u> [69.16, 77.31]	<u>80.13</u> [76.77, 83.40]	81.08 [77.88, 84.15]	81.53 [78.40, 84.53]	<u>78.89</u> [75.46, 82.18]
7		+ U	72.69 [68.50, 76.87]	84.36 [81.06, 87.66]	88.55 [85.46, 91.63]	90.09 [87.22, 92.95]	72.69 [68.50, 76.87]	79.21 [75.76, 82.66]	80.57 [77.35, 83.79]	80.96 [77.80, 84.13]	78.20 [74.76, 81.52]
8		+ C	71.59 [67.40, 75.77]	85.24 [81.94, 88.55]	88.33 [85.24, 91.41]	90.31 [87.44, 93.17]	71.59 [67.40, 75.77]	79.21 [75.82, 82.60]	80.21 [76.97, 83.44]	80.72 [77.57, 83.86]	77.78 [74.31, 81.09]
9		+ S + U	72.91 [68.72, 76.87]	85.46 [82.16, 88.55]	87.89 [84.80, 90.75]	90.09 [87.22, 92.73]	72.91 [68.72, 76.87]	79.81 [76.48, 83.06]	80.52 [77.24, 83.66]	81.09 [77.96, 84.17]	78.35 [74.91, 81.68]
10		+ S + C	73.57 [69.38, 77.53]	85.68 [82.38, 88.77]	88.11 [85.02, 90.97]	90.09 [87.22, 92.73]	73.57 [69.38, 77.53]	80.31 [76.94, 83.53]	<u>81.12</u> [77.90, 84.20]	<u>81.64</u> [78.51, 84.63]	79.03 [75.64, 82.32]
11		+ U + C	72.47 [68.28, 76.65]	84.80 [81.50, 88.11]	<u>88.43</u> [85.34, 91.52]	89.87 [87.00, 92.73]	72.47 [68.28, 76.65]	79.33 [75.89, 82.77]	80.51 [77.28, 83.74]	80.84 [77.67, 84.00]	78.09 [74.65, 81.41]
12		+ S + U + C	72.69 [68.50, 76.65]	85.46 [82.16, 88.55]	87.67 [84.58, 90.53]	90.09 [87.22, 92.73]	72.69 [68.50, 76.65]	79.80 [76.44, 83.02]	80.46 [77.22, 83.58]	81.10 [78.00, 84.12]	78.34 [74.91, 81.66]

Table 1: **Main results.** + RERANK uses retrieval together with reranking; A, +S, +U, +C progressively add abstract (A), protein structure (S) metadata, UniProt (U) descriptions, and chemical (C) component names in reranking. In each column, best in **red bold** and second-best in blue underline. Values in brackets are 95% bootstrap CIs.

reranking (rows 5-12); and (ii) the contribution of multi-source knowledge enrichment, by appending protein entry-wise knowledge sources to the abstract (rows 5-12). The PubMed abstract is included in every configuration to provide the primary context of protein structure. We have gained the following major findings:

Finding 1: Hybrid retrieval ensures candidate coverage, while the knowledge-enriched reranking is necessary for precise rank ordering.

We found that the hybrid retrieval module could identify potentially relevant candidates but could not rank them precisely; adding a knowledge-enriched reranker could significantly improve candidate ordering by embedding heterogeneous protein knowledge. In Table 1, the hybrid retriever already places the correct article inside the top 20 of queries (row 4 Recall@20, 85.68), yet at recall@1 is only 53.74, indicating the necessity of further precise reranking. By adding a reranker using abstract only, the performance achieved a remarkable improvement (row 5 vs. row 4: MRR 71.29 vs. 62.91; Recall@1 62.78 vs. 53.74). Further-

more, leveraging protein entry-wise knowledge in the reranker could complement the domain-specific information to distinguish co-deposited structures that share an abstract, significantly enhancing the search accuracy (row 10 vs. row 5: MRR 79.03 vs. 71.29; Recall@1 73.57 vs. 62.78), revealing the importance of domain-specific knowledge accumulation. As shown in Figure 3, the proposed workflow can accurately identify the ground-truth PubMed ID and the corresponding protein entries for the given query, providing a concise rationale and judgment of relevance.

Finding 2: The combination of abstract and protein entry-level knowledge consistently improves protein searching performance.

Protein entry-wise context provides a unique description for each entry and fine-grained information specific to each protein structure, while PubMed abstracts contain coarse information at the article level, shared across all co-deposited protein structures. These two granularities of knowledge complement each other, enabling a comprehensive relevance assessment for improving search perfor-

PubMed ID: 38425922 PDB ID: 8K7P 7YOC 7YP0 8K2O Reason: Yes, the provided information is directly relevant as it describes the structure of the Fhb7 gene and its variants, along with their role in degrading DON, which addresses the user's query about FHB resistance gene structures and variants involved in DON degradation. The detailed catalytic mechanism and stability improvements of certain variants also support the user's interest. Judgement: The abstract is directly relevant to the question because it describes the structure of the Fhb7 gene and its variants, along with their role in degrading DON, which addresses the user's query about FHB resistance gene structures and variants involved in DON degradation. Relevance score (0-2): 2
PubMed ID: 30112765 PDB ID: 6GCB 6GC9 6HTA 6GCC 6GCA Reason: The provided information about the structural analysis of TvGSTX1 and TvGSTX3 from <i>Trametes versicolor</i> is not directly relevant to the FHB resistance gene Fhb7 or its variants involved in degrading deoxynivalenol (DON). The focus on Xi and Omega GSTs does not address the specific query regarding Fhb7. Therefore, it cannot be used to answer the user's question. Judgement: The abstract is irrelevant to the question because the provided information about the structural analysis of TvGSTX1 and TvGSTX3 from <i>Trametes versicolor</i> does not address the specific query regarding the Fhb7 gene or its variants involved in degrading deoxynivalenol. Relevance score (0-2): 0

Figure 3: **UI output** for query: "Can you find structures of Fhb7 and its variants involved in degrading deoxynivalenol?". Green box shows the structures with the highest relevance score, while the grey box shows the lowest.

mance. As shown in Table 1, each single-source knowledge-enriched configuration (rows 6-8) consistently outperforms abstract-only reranking (row 5) by a large margin (recall@1 row 6 vs. row 5 73.35 vs. 62.78). Such per-entry protein-structural knowledge allows the reranker to discriminate entries at the structure level rather than the article level, yielding more query-relevant results. Furthermore, we observe that multi-source protein structure knowledge can complementarily enhance search performance in a large margin (row 10 vs. row 5: MRR 79.03 vs. 71.29; Recall@1 73.57 vs. 62.78). Chemical components could be effectively appended to content beyond the abstract and structure title, enabling the most comprehensive knowledge enrichment. As shown in Table 1, +A+S+C (row 10) achieved the best MRR (79.03) among all combinations of multi-source knowledge.

Finding 3: Protein-structural knowledge consistently improves searching performance, whereas protein-functional knowledge does not.

Multiple protein-structural knowledge sources can consistently improve protein searching performance; however, continually increasing protein-functional sources does not lead to significant improvements. As shown in Table 1, combining all four sources (row 12, MRR 78.34) did not introduce further benefits compared with either +A+S (row 6, MRR 78.89) or +A+S+C (row 10, MRR 79.03). UniProt functional annotations describe the protein at the entity level, such as catalytic activity,

biological role, and subcellular localization, with less structural context. Such functional annotations offer limited benefits for protein structure search, as most useful structural information is commonly described in abstracts, titles, and search queries by their widely recognized common names rather than the standardized UniProt names of their individual components. Also, multiple knowledge sources may contain query-irrelevant content, inevitably introducing ambiguity into the search results.

5.3 Ablation study

To assess the robustness of the proposed workflow, we conduct three ablation studies: (i) the effect of the abbreviation rewrite module that preprocesses ambiguous queries before retrieval, (ii) the choice of bi-encoder retriever and cross-encoder reranker, and (iii) the effect of chunk and its overlap size.

Effect of the Abbreviation rewrite We evaluate the effect of the query preprocessing (Step 2 in Figure 1) on the subset of the proposed benchmark. To construct the subset, we convert each initial query, which originally contains both the abbreviation (abbr) and its full name, into an abbr-only form. As shown in Table 2, the workflow with abbr rewrite outperforms the no-rewrite baseline (row 3, w/ (Ours) vs. w/o Recall@10 87.44 vs. 86.12) by leveraging external knowledge to expand abbreviations to their full names, resulting in a more precise retrieval and knowledge-enriched reranking. Meanwhile, we observe that the rewrite query can efficiently improve the coverage of initial candidates in the retrieval stage to ensure the

Row	Module	Knowledge	Recall@10			NDCG@10			MRR		
			w/o	w/ (ours)	Δ	w/o	w/ (ours)	Δ	w/o	w/ (ours)	Δ
1	Retrieval only	Abstract	75.99 [72.03, 79.74]	78.63 [74.67, 82.38]	+2.64	62.96 [59.08, 66.69]	64.76 [60.98, 68.45]	+1.80	59.28 [55.31, 63.19]	60.80 [56.81, 64.70]	+1.52
2		Abstract	83.26 [79.74, 86.56]	84.58 [81.28, 87.89]	+1.32	71.13 [67.52, 74.65]	72.65 [69.22, 76.13]	+1.52	67.54 [63.71, 71.31]	69.10 [65.40, 72.83]	+1.56
3		+ S	86.12 [82.82, 89.21]	87.44 [84.36, 90.31]	+1.32	78.46 [75.04, 81.75]	79.62 [76.33, 82.78]	+1.16	76.10 [72.49, 79.60]	77.21 [73.68, 80.65]	+1.11
4	+ Rerank	+ S + U	85.68 [82.38, 88.77]	87.00 [83.92, 90.09]	+1.32	77.87 [74.44, 81.17]	78.95 [75.64, 82.19]	+1.08	75.53 [71.94, 79.07]	76.52 [73.01, 80.00]	+0.99
5		+ S + C	85.90 [82.60, 88.99]	87.22 [84.14, 90.09]	+1.32	78.51 [75.14, 81.75]	79.82 [76.57, 83.01]	+1.31	76.24 [72.66, 79.69]	77.54 [74.09, 80.96]	+1.30
6		+ S + U + C	85.46 [82.16, 88.55]	86.78 [83.48, 89.87]	+1.32	77.81 [74.42, 81.09]	79.02 [75.78, 82.24]	+1.21	75.51 [71.97, 78.98]	76.68 [73.20, 80.11]	+1.17

Table 2: **The effect of the abbreviation rewritten module** (Step 1(b) of Figure 1). *w/o* feeds the raw query to the retriever; *w/ (ours)* feeds the rewritten query. Δ denotes the absolute improvement (*w/* − *w/o*). The better result in each pair is in **bold**. Values in brackets are 95% bootstrap CIs. All metrics are reported as percentages (%).

performance of the subsequent steps. The abbreviation expansion significantly enhances the inclusion of correct candidates, improving Recall@10 from 75.99 to 78.63 as shown in row 1.

Effect of Module Selection We evaluate the effect of different retriever-reranker combinations to identify the optimal pairing. Specifically, we cross two bi-encoder retrievers (BGE-SMALL-EN-V1.5 and BGE-LARGE-EN-V1.5) with two cross-encoder rerankers (BGE-RERANKER-LARGE and BGE-RERANKER-V2-M3 (Chen et al., 2024)), yielding four pairings evaluated under five knowledge-enrichment combinations. As shown in Figure 4a, with a fixed retriever, RERANKER-LARGE consistently outperforms V2-M3 with knowledge enrichment applied (average Recall@1 73.13 vs. 71.86; light blue vs. light pink bars in Figure 4a). We then compare the two retrievers under a fixed reranker. Using BGE-SMALL as the retriever achieves performance comparable to BGE-LARGE across all knowledge combinations (average Recall@1 73.13 vs. 73.07; light vs. dark blue bars in Figure 4a). Because the smaller retriever yields substantially lower query-time costs while achieving the comparable accuracy of the larger one, we use BGE-SMALL as the retriever and BGE-RERANKER-LARGE as the reranker, allowing a trade-off between search efficiency and accuracy.

Effect of Chunk and Overlap Size We evaluate the effects of chunk size and chunk overlap on search performance. The chunk size refers to the maximum number of characters per abstract, while the chunk overlap is designed to preserve cross-chunk context by sharing a small number of

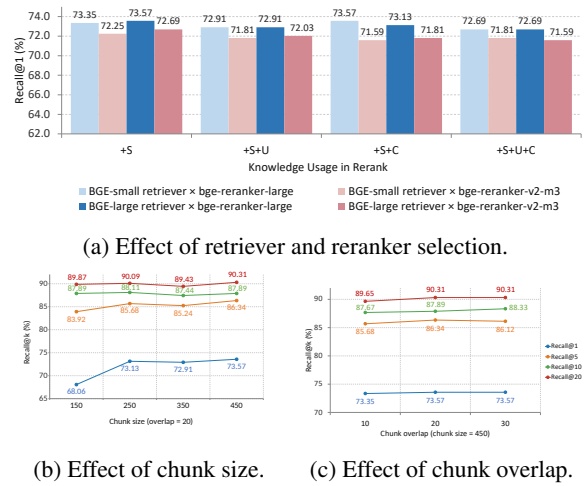


Figure 4: **Ablation studies.** (a) The effect of the module selection. (b) The effect of chunk size. (c) The effect of chunk overlap size. configuration.

characters between adjacent chunks to indicate the semantic relationship. A smaller chunk size yields finer-grained content but breaks each abstract into pieces, while a larger chunk size preserves more context. As shown in Figure 4b, we found that using a larger chunk size substantially improves retrieval performance over the smallest one (Recall@1 150 vs. 450, 68.06 vs. 73.57). In contrast, given the trade-off between search accuracy and cost, we set the overlap size to 20, which yields the same performance as a larger overlap.

Human evaluation for LLM-as-a-Judge In the UI, the “LLM-as-a-Judge” component serves only to provide complementary information and suggestions to help users with their decision-making, rather than reaching a conclusion. Four experts

Expert	Quality	Correctness	Helpfulness
1	5	4	5
2	5	4.67	5
3	5	4.67	5
4	4.67	4.33	4.33
Overall Average	4.92	4.42	4.83

Table 3: **Human evaluation for LLM-as-a-Judge**

conducted user studies on 12 samples by rating the “LLM-as-a-Judge” explanations on Quality, Correctness, and Helpfulness. The rubric scores each dimension independently on the 1-5 scale. For each metric, we report the average score for each expert across three samples in Table 3. The overall average scores show that the LLM-generated explanations in the UI are fluent and highly readable (Quality: 4.92) and clearly useful for accelerating RCSB-PDB user searches (Helpfulness: 4.83). Factual correctness with only minor inaccuracies (Correctness: 4.42); for example, the LLM mislabeled the structure as the Fhb7 gene, while it is the Fhb7 enzyme (Appendix 13).

For the rubrics provided to the experts, *Quality* measures readability and fluency (5: reads like well-written human content; 3: understandable but awkward; 1: incoherent). *Correctness* measures factual accuracy (5: no errors; 3: mostly accurate with a minor error or unsupported claim; 1: clear factual errors). *Helpfulness* measures usefulness for accelerating RCSB-PDB searches (5: clearly explains relevance for quick assessment; 3: some useful but vague signal; 1: no useful interpretation).

Comparison with real-world search systems. We conduct the external comparison with two real-world search systems, i.e., RCSB PDB Search API (Piehl et al., 2025) and NCBI Entrez API (Sayers, 2022). For full-text search, we submitted each question to the RCSB PDB Search API after stripping conversational prefixes and parenthetical abbreviations. For PubMed E-Search, we submitted the same queries to the NCBI Entrez API. As shown in Table 4, we found that our proposed RAD-Protein substantially outperforms all existing real-world search systems across every metric:

Method	Recall@10	MRR	NDCG@10
RCSB Full-text Search	22.56	20.28	20.83
PubMed E-Search	51.54	36.17	39.74
Ours	88.11	79.03	81.12

Table 4: **Comparison with real-world search systems.**

6 Conclusion

RCSB Protein Data Bank users routinely seek to identify structures relevant to scientific questions, yet existing search interfaces remain limited to attribute- or keyword-based queries. To bridge this gap, we presented RAD-PROTEIN, an efficient multi-source, knowledge-enriched retrieval-augmented search framework that synthesizes heterogeneous literature and metadata to enable text-driven protein structure search. We also contribute a human-verified dataset that authentically reflects real-world scientific inquiry in this domain. The benchmark mimics real-world user demand for searching protein structures using plain-text queries. Finally, we introduced an interpretable interface that allows RCSB PDB users to conduct protein searches via plain-text queries, supporting users understand the search process and accelerating scientific verification.

Limitations

We identify following limitations of this study. First, all experiments use the curated PubMed subset and the RCSB GraphQL snapshot packaged with the project; we did not include full-text articles due to open-access limitations. Second, both the evaluation benchmark and the reason-judge stage rely on the Qwen-2.5 family of models used for inference, which may introduce correlated bias across question generation and scoring. We used GPT-4 to produce rationale and judgement (Appendix 14), which shows performance similar to the Qwen-2.5 family; a more comprehensive comparison is expected to be explored in future work. Third, the pipeline scores only experimentally determined PDB entries with an associated PubMed ID. Extending the per-PDB enrichment scheme to AlphaFold-predicted structures (Jumper et al., 2021) that are not yet linked to a publication is a natural next step to broaden the search space. Finally, our benchmark queries specifically focus on protein structure searching. Since RCSB PDB also contains macromolecular structures other than proteins, a larger, more comprehensive benchmark could further broaden the scope of the proposed method.

Ethical Considerations

The proposed RAD-PROTEIN workflow and benchmark are intended solely for research and educational purposes. All external resources used in this

work were used in accordance with their publicly available research access conditions and licenses. The developed benchmark contains only expert-verified queries, the corresponding ground-truth PubMed IDs, and expert annotation scores, rather than redistributed copyrighted full-text biomedical articles. AI assistants were used to support prompt example figure refinement (i.e., [Appendix 9](#), [Appendix 10](#), [Appendix 11](#), and [Appendix 12](#)). All technical content and final manuscript materials were reviewed and verified by the authors.

Acknowledgments

We would like to thank the reviewers, area chairs, and program chairs for their evaluation and feedback. This work was completed as part of the Rutgers Artificial Intelligence and Data Science (RAD) Collaboratory and made possible with funding and support provided by the Rutgers-New Brunswick Office of the Chancellor, the Rutgers-New Brunswick Office for Research, and the Rutgers University Office of the Executive Vice President for Academic Affairs. Also, this work is part of RCSB Protein Data Bank core operations, which are jointly funded by the US National Science Foundation (NSF) (DBI-2321666, PI: S.K. Burley), the US Department of Energy (DE-SC0019749, PI: S.K. Burley), and the National Cancer Institute, the National Institute of Allergy and Infectious Diseases, and the National Institute of General Medical Sciences of the NIH (R01GM157729, PI: S.K. Burley). Furthermore, this research was supported by research grants to Dimitris N. Metaxas through NSF grants 2212302, 2212301, 2212303, AFOSR 23RT0630, and NIH 2R01HL127661.

References

- Samy Ateia and Udo Kruschwitz. 2025. Bioragent: a retrieval-augmented generation system for showcasing generative query expansion and domain-specific search for scientific q&a. In *European Conference on Information Retrieval*, pages 1–5. Springer.
- Helen M Berman, John Westbrook, Zukang Feng, Gary Gilliland, Talapady N Bhat, Helge Weissig, Ilya N Shindyalov, and Philip E Bourne. 2000. The protein data bank. *Nucleic acids research*, 28(1):235–242.
- Stephen K Burley, Charmi Bhikadiya, Chunxiao Bi, Sebastian Bittrich, Henry Chao, Li Chen, Paul A Craig, Gregg V Crichlow, Kenneth Dalenberg, Jose M Duarte, Shuchismita Dutta, Maryam Fayazi, Zukang Feng, Justin W Flatt, Sai Ganesan, Sutapa Ghosh, David S Goodsell, Rachel Kramer Green, Vladimir Guranovic, and 22 others. 2023. Rcsb protein data bank (rcsb.org): delivery of experimentally-determined pdb structures alongside one million computed structure models of proteins from artificial intelligence/machine learning. *Nucleic acids research*, 51(D1):D488–D508.
- Kathi Canese and Sarah Weis. 2013. Pubmed: the bibliographic database. *The NCBI handbook*, 2(1):2013.
- 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.
- Zhe Chen, Yusheng Liao, Shuyang Jiang, Pingjie Wang, Yiqiu Guo, Yanfeng Wang, and Yu Wang. 2025. Towards omni-rag: Comprehensive retrieval-augmented generation for large language models in medical applications. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15285–15309.
- Chroma. 2026. Chroma. <https://www.trychroma.com/>. Online resource, accessed May 25, 2026.
- Cloudflare. 2026. Cloudflare Tunnel. <https://developers.cloudflare.com/cloudflare-one/networks/connectors/cloudflare-tunnel/>. Online documentation, accessed May 25, 2026.
- UniProt Consortium. 2023. Uniprot: the universal protein knowledgebase in 2023. *Nucleic acids research*, 51(D1):D523–D531.
- Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Qianyu Guo, Meng Wang, and Haofen Wang. 2024. [Retrieval-augmented generation for large language models: A survey](#). Preprint, arXiv:2312.10997.
- Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2022. [Unsupervised dense information retrieval with contrastive learning](#). *Transactions on Machine Learning Research*.
- Qiao Jin, Yifan Yang, Qingyu Chen, and Zhiyong Lu. 2024. Genegpt: augmenting large language models with domain tools for improved access to biomedical information. *Bioinformatics*, 40(2):btae075.
- John M. Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, Alex Bridgland, Clemens Meyer, Simon A A Kohl, Andy Ballard, Andrew Cowie, Bernardino Romera-Paredes, Stanislav Nikolov, Rishub Jain, Jonas Adler, and 15 others. 2021. Highly accurate protein structure prediction with alphafold. *nature*, 596(7873):583–589.

- LangChain Community. 2026. LangChain: The agent engineering platform. <https://github.com/langchain-ai/langchain>. GitHub repository, accessed May 25, 2026.
- Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474.
- Dawei Li, Bohan Jiang, Liangjie Huang, Alimohammad Beigi, Chengshuai Zhao, Zhen Tan, Amrita Bhat-tacharjee, Yuxuan Jiang, Canyu Chen, Tianhao Wu, Kai Shu, Lu Cheng, and Huan Liu. 2025. From generation to judgment: Opportunities and challenges of llm-as-a-judge. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 2757–2791.
- Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: Nlg evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pages 2511–2522.
- Xueguang Ma, Kai Sun, Ronak Pradeep, and Jimmy J. Lin. 2021. A replication study of dense passage retriever. *ArXiv*, abs/2104.05740.
- Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke E. Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Francis Christiano, Jan Leike, and Ryan J. Lowe. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.
- Dennis W. Piehl, Brinda Vallat, Ivana Truong, Habiba Morsy, Rusham Bhatt, Santiago Blaumann, Pratyoy Biswas, Yana Rose, Sebastian Bittrich, Jose M. Duarte, Joan Segura, Chunxiao Bi, Douglas Myers-Turnbull, Brian P. Hudson, Christine Zardecki, and Stephen K. Burley. 2025. rcsb-api: python toolkit for streamlining access to rcsb protein data bank apis. *Journal of molecular biology*, 437(15):168970.
- Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, pages 3982–3992.
- Stephen E. Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: Bm25 and beyond. *Found. Trends Inf. Retr.*, 3:333–389.
- Yana Rose, Jose M. Duarte, Robert Lowe, Joan Segura, Chunxiao Bi, Charmi Bhikadiya, Li Chen, Alexander S. Rose, Sebastian Bittrich, Stephen K. Burley, and John D. Westbrook. 2020. Rcsb protein data bank: Architectural advances towards integrated searching and efficient access to macromolecular structure data from the pdb archive. *Journal of molecular biology*, 433:166704 – 166704.
- Eric Sayers. 2022. A general introduction to the e-utilities. In *Entrez® Programming Utilities Help [Internet]*. National Center for Biotechnology Information (US).
- Qwen Team. 2024. Qwen2.5: A party of foundation models.
- 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 *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc.
- Shitao Xiao, Zheng Liu, Peitian Zhang, Niklas Muen-nighoff, Defu Lian, and Jian-Yun Nie. 2024. C-pack: Packed resources for general chinese embeddings. In *Proceedings of the 47th international ACM SIGIR conference on research and development in information retrieval*, pages 641–649.
- Guangzhi Xiong, Qiao Jin, Zhiyong Lu, and Aidong Zhang. 2024. Benchmarking retrieval-augmented generation for medicine. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 6233–6251.
- An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, and 40 others. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.
- Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623.
- Honglei Zhuang, Zhen Qin, Kai Hui, Junru Wu, Le Yan, Xuanhui Wang, and Michael Bendersky. 2024. Beyond yes and no: Improving zero-shot llm rankers via scoring fine-grained relevance labels. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 358–370.

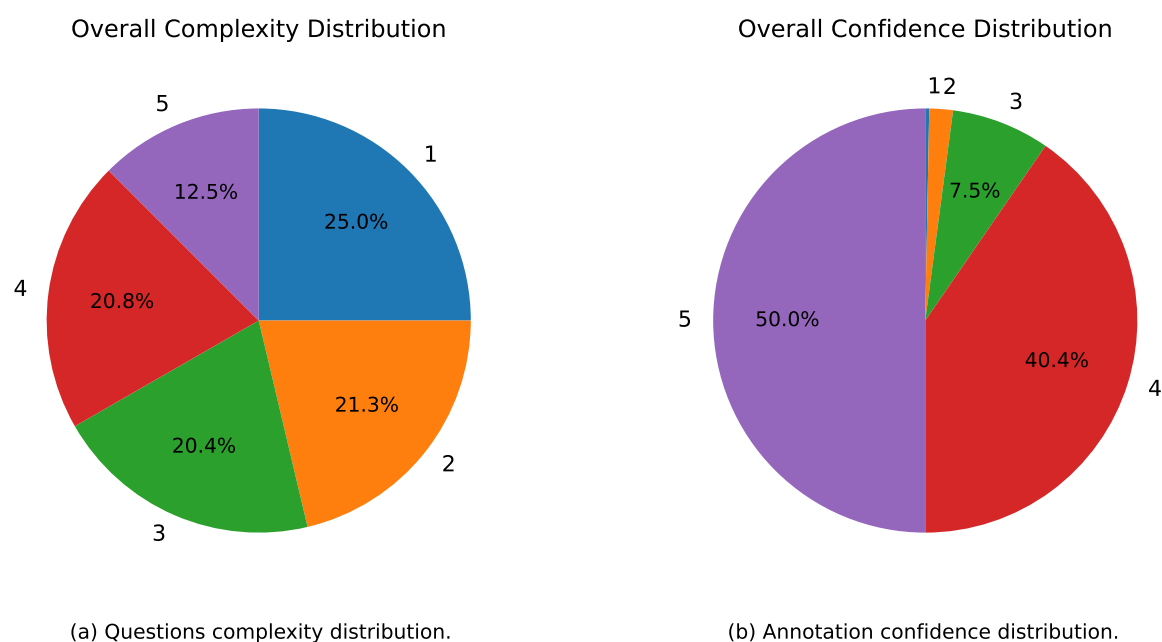


Figure 5: **Human annotation statistics in complexity and confidence.** Colors indicate the rating level: ■ 1, ■ 2, ■ 3, ■ 4, ■ 5. For complexity (a), 1 indicates questions a human can easily solve with their own knowledge or by translating into search API keywords, while 5 indicates questions difficult for a human to solve. For confidence (b), 1 indicates the annotator was unsure of the annotation, and 5 indicates they were very confident in the annotation.

Ask a question about the protein/structure corpus used in this project.
About 2-3 minutes to generate an answer.

Current status

Ready.

Question

Can you find structures of Bmml in complex with malonylated/succinylated macrolactins?

Ask

Async batch submission

Submit multiple questions with your email. Processing runs in the background; results are saved to JSON and emailed when complete.

Questions (one per line)

Can you find structures of Bmml in complex with malonylated/succinylated macrolactins?
Can you find structures of Fhb7 and its variants involved in degrading deoxynivalenol?
Can you find structures of aldo-keto reductases showing positive cooperativity with substrates?

Email for completion notification

name@example.com

Submit Batch

Batch submission status

No batch job submitted yet.

Figure 6: **User Interface.** The UI interface supports both single and batch question submission.

Question

Can you find the hemoglobin structure of high-altitude birds?

Ask

Retrieved articles; showing the top 7 sorted by relevance score (0-2), with ties in reranker order. Below: each article's PMID, PDB IDs, reason, judgement, score, and full abstract (relevant entries highlighted).

PubMed ID: 11375496

PDB ID: 1C40

Reason: The provided information can be used to infer that the hemoglobin structure of high-altitude birds, specifically the bar-headed goose, has unique characteristics that contribute to higher oxygen affinity, which may be relevant to understanding similar structures in other high-altitude bird species. However, specific structures for other high-altitude bird species are not detailed in the given literature. Therefore, while informative, the exact structures for all high-altitude birds cannot be fully deduced from this data alone.

Judgement: The abstract is directly relevant to the question because it provides insights into the unique hemoglobin structure of high-altitude birds, particularly the bar-headed goose, which may inform similar structures in other high-altitude bird species.

Relevance score (0-2): 2

Full abstract: Haemoglobin from the bar-headed goose (*Anser indicus*) has higher oxygen affinity than that from its lowland relatives such as greylag goose (*A. anser*). The crystal structure of bar-headed goose aquomet haemoglobin was determined at 2.3 Å resolution and compared with the structures of the goose oxy, human, horse and other avian haemoglobins and the sequences of other avian haemoglobins. Four amino-acid residues differ between greylag goose and bar-headed goose haemoglobins, among which Alaα119 and Aspβ125 in bar-headed goose haemoglobin reduces the contacts between the α1 and β1 subunits compared with Pro and Glu, respectively, and therefore may increase the oxygen affinity by loosening the α1β1 interface. Compared with human oxy haemoglobin, the relative orientation of two αβ dimers in the bar-headed goose aquomet and oxy Hbs are rotated by about 4 degrees, indicating a unique quaternary structural difference from the typical R state. This new 'R(H)' state is probably correlated with the higher oxygen affinity of bar-headed goose haemoglobin.

PubMed ID: 23545644

PDB ID: 4H2L

Reason: Yes, the provided information can be helpful as it describes the structural differences between high-altitude deer mouse hemoglobin and other species, which may provide insights into the adaptations of high-altitude birds. However, specific data on high-altitude bird hemoglobin structures is not directly available in the given context.

Judgement: The abstract is somewhat related to the question because it provides structural insights into high-altitude deer mouse hemoglobin, which might inform about similar adaptations in high-altitude birds, though specific data on bird hemoglobin is not provided.

Relevance score (0-2): 1

Full abstract: The deer mouse, *Peromyscus maniculatus*, exhibits altitude-associated variation in hemoglobin oxygen affinity. To examine the structural basis of this functional variation, the structure of the hemoglobin was solved. Recombinant hemoglobin was expressed in *Escherichia coli* and was purified by ion-exchange chromatography. Recombinant hemoglobin was crystallized by the hanging-drop vapor-diffusion method using polyethylene glycol as a precipitant. The obtained orthorhombic crystal contained two subunits in the asymmetric unit. The refined structure was interpreted as the aquo-met form. Structural comparisons were performed among hemoglobins from deer mouse, house mouse and human. In contrast to human hemoglobin, deer mouse hemoglobin lacks the hydrogen bond between α1Trp14 in the A helix and α1Thr67 in the E helix owing to the Thr67Ala substitution. In addition, deer mouse hemoglobin has a unique hydrogen bond at the α1β1 interface between residues α1Cys34 and β1Ser128.

PubMed ID: 25061917

PDB ID: 4NI0 4NI1

Reason: The provided information does not directly address the hemoglobin structure of high-altitude birds. The literature focuses on human hemoglobin and its modulation by small molecules, which is not relevant to the specific query about high-altitude birds. Therefore, this information cannot be used to answer the user's question.

Judgement: The abstract is irrelevant to the question because the provided information focuses on human hemoglobin and does not address the hemoglobin structure of high-altitude birds.

Relevance score (0-2): 0

Full abstract: Small molecules that increase the oxygen affinity of human hemoglobin may reduce sickling of red blood cells in patients with sickle cell disease. We screened 38,700 compounds using small molecule microarrays and identified 427 molecules that bind to hemoglobin. We developed a high-throughput assay for evaluating the ability of the 427 small molecules to modulate the oxygen affinity of hemoglobin. We identified a novel allosteric effector of hemoglobin, di[5-(2,3-dihydro-1,4-benzodioxin-2-yl)-4H-1,2,4-triazol-3-yl]disulfide (TD-1). TD-1 induced a greater increase in oxygen affinity of human hemoglobin in solution and in red blood cells than did 5-hydroxymethyl-2-furfural (5-HMF), N-ethylmaleimide (NEM), or diformamidine disulfide. The three-dimensional structure of hemoglobin complexed with TD-1 revealed that monomeric units of TD-1 bound covalently to β-Cys93 and β-Cys112, as well as noncovalently to the central water cavity of the hemoglobin tetramer. The binding of TD-1 to hemoglobin stabilized the relaxed state (R3-state) of hemoglobin. TD-1 increased the oxygen affinity of sickle hemoglobin and inhibited in vitro hypoxia-induced sickling of red blood cells in patients with sickle cell disease without causing hemolysis. Our study indicates that TD-1 represents a novel lead molecule for the treatment of patients with sickle cell disease.

Figure 7: **Example of UI search result.** Green box shows the searched article and corresponding structures are relevant to the query (relevance score is 2), blue box shows the searched article and corresponding structures are partially relevant to the query (relevance score is 1), and the grey box shows the searched article and corresponding structures are not relevant to the query (relevance score is 0).

Question

Can you find structures of Bmml in complex with malonylated/succinylated macrolactins?

Ask

Retrieved articles; showing the top 9 sorted by relevance score (0-2), with ties in reranker order. Below: each article's PMID, PDB IDs, reason, judgement, score, and full abstract (relevant entries highlighted).

PubMed ID: [32803979](#)

PDB ID:

6KJT

6KJQ

6KJJ

6KJP

6KJH

6KJR

Reason: The provided information includes the crystal structure of Bmml in complex with a succinyl group and the structure of Bmml with acyl acceptor MLN A, which is relevant to the query about β -lactamase homologue Bmml in complex with malonylated/succinylated macrolactins. However, specific structures of Bmml in complex with malonylated/succinylated macrolactins are not directly mentioned in the given literature. Therefore, while the existing data provides a foundation, it does not explicitly address the exact complexes sought after in the user's question. The information can provide context but not a definitive answer to the specific complexes requested.

Judgement: The abstract is somewhat related to the question because it provides structural information on Bmml in complex with similar acyl groups, which can offer context but does not explicitly address the specific complexes with malonylated/succinylated macrolactins requested.

Relevance score (0-2): 1

Full abstract: Macrolactins (MLNs) are a class of important antimaculacul degeneration and antitumor agents. Malonylated/succinylated MLNs are even more important due to their efficacy in overcoming multi-drug-resistant bacteria. However, which enzyme catalyzes this reaction remains enigmatic. Herein, we deciphered a β -lactamase homologue Bmml to be responsible for this step. Bmml could specifically attach C3-C5 alkyl acid thioesters onto 7-OH of MLN A and also exhibits substrate promiscuity toward acyl acceptors with different scaffolds. The crystal structure of Bmml covalently linked to the succinyl group and systematic mutagenesis highlighted the role of oxyanion holelike geometry in the recognition of carboxyl-terminated acyl donors. The engineering of this geometry expanded its substrate scope, with the R166A/G/Q variants recognizing up to C12 alkyl acid thioester. The structure of Bmml with acyl acceptor MLN A revealed the importance of Arg292 in the recognition of macrolide substrates. Moreover, the mechanism of the Bmml-catalyzed acyltransfer reaction was established, unmasking the deft role of Lys76 in governing acyl donors as well as catalysis. Our studies uncover the delicate mechanism underlying the substrate selectivity of acyltransferases, which would guide rational enzyme engineering for drug development.

PubMed ID: [23148776](#)

PDB ID:

4GNU

4GOG

4H8R

Reason: The provided information does not contain any details about the structures of β -lactamase homologue Bmml in complex with malonylated/succinylated macrolactins, thus it cannot be used to answer the user's question. The literature focuses on GES-1 and GES-5 β -lactamases and their complexes with imipenem, which are different from Bmml and the specific macrolactins mentioned.

Judgement: The abstract is irrelevant to the question because the provided information pertains to different β -lactamases (GES-1 and GES-5) and macrolactins (imipenem) than those specified in the question (Bmml and malonylated/succinylated macrolactins).

Relevance score (0-2): 0

Full abstract: Carbapenem antibiotics have become therapeutics of last resort for the treatment of difficult infections. The emergence of class-A β -lactamases that have the ability to inactivate carbapenems in the past few years is a disconcerting clinical development in light of the diminished options for treatment of infections. A member of the GES-type β -lactamase family, GES-1, turns over imipenem poorly, but the GES-5 β -lactamase is an avid catalyst for turnover of this antibiotic. We report herein high-resolution X-ray structures of the apo GES-5 β -lactamase and the GES-1 and GES-5 β -lactamases in complex with imipenem. The latter are the first structures of native class-A carbapenemases with a clinically used carbapenem antibiotic in the active site. The structural information is supplemented by information from molecular dynamics simulations, which collectively for the first time discloses how the second step of catalysis by these enzymes, namely, hydrolytic deacylation of the acyl-enzyme species, takes place effectively in the case of the GES-5 β -lactamase and significantly less so in GES-1. This information illuminates one evolutionary path that nature has taken in the direction of the inexorable emergence of resistance to carbapenem antibiotics.

Figure 8: **Example of UI search result.** Green box shows the searched article and corresponding structures are relevant to the query (relevance score is 2), blue box shows the searched article and corresponding structures are partially relevant to the query (relevance score is 1), and the grey box shows the searched article and corresponding structures are not relevant to the query (relevance score is 0).

System Prompt:

You are a PDB user who might ask to retrieve specific structures related to a wide variety of biological relevance and functions. (E.g., “Can you find a protein with 2 domains connected by a flexible link” or “Can you give me structures (from homo sapiens) containing a disulfide bond”, etc.).

Task Prompt:

I have an abstract describing the structure and function of a protein or protein complex. I want you to generate a list of high-level, biologically meaningful questions that a PDB user might ask an AI search tool on RCSB.org to find relevant structures. The questions should:

1. Be focused on searching for structures in the PDB.
2. Be phrased like “Can you find X” or “Can you give me structures of Y” — concise, clear, and actionable.
3. Highlight biological relevance or functional aspects, not overly specific residue-level details.
4. Be answerable based on the information in the abstract.
5. Avoid unnecessary jargon; they should read like real search queries a scientist might enter.
6. Either the full name or abbreviation. Do not let them occur at the same time.

For example, if the abstract is about hemagglutinin from influenza viruses, suitable questions could be:

- Can you find structures of H3 hemagglutinin from avian influenza viruses?
- Can you show me hemagglutinin structures bound to sialic acid-containing sugars?
- Can you give me the structures of HA relevant to cross-species influenza transmission?

Response format:

Question: {generated question}

Reference: {a list of reference sentences}

Input:

Abstract: ABSTRACT TO INDEX

Figure 9: The LLM prompts of question generation.

Table 5: Distribution of questions across 11 super-categories.

Super-Category	%
Enzymes	45.7%
Viral / Pathogen	14.1%
Immune System	8.9%
Membrane / Transport	8.3%
Nucleic Acid / Gene Regulation	5.6%
Photosynthesis / Energy	4.7%
Structural / Miscellaneous	4.1%
Signaling / Cell Biology	3.7%
Binding Proteins	3.1%
Others	1.7%

Table 6: Distribution of queries by search intent.

Query Type	%
General Protein Structure	71.5%
Complex / Binding Partner	13.4%
Organism-Specific	6.4%
Structural Feature (conformation, domain, motif)	3.7%
Protein Family / Class	1.9%
Functional / Mechanistic	1.6%
Inhibitor-Bound	1.0%
Oligomeric State	0.4%
Apo State	0.2%

System Prompt:

You are good at selecting the most useful question based on the provided instruction for a typical PDB user.

Task Prompt:

From the list of search questions you just generated, select the ones that are most useful for a typical PDB user. Choose questions that:

1. Are high- or mid-level (not overly specific or residue-level).
2. Are biologically meaningful and focused on finding structures relevant to a biological or biochemical concept.
3. Are broad enough that the search results would include the structure described in the abstract **as well as** other related structures from different studies.
4. Are purely about **searching for structures**, and **do not** involve performing or suggesting any kind of comparison, interpretation, or analysis (e.g., avoid phrases like “for comparison,” “to analyze,” “to study,” etc.).

In response, only include one best question that meets these criteria and directly copy the list of reference sentences. If needed, lightly rephrase the question to make it clearer, more general, and more typical of what a PDB user would ask.

Response format:

Question: {the selected question}

Reference: {a list of reference sentences}

Input:

Generated questions: GENERATED QUESTIONS TO FILTER

Figure 10: The LLM prompts of question filtering.

System / Role:

You are an experienced biomedical analyst for biomedical retrieval.

User question:

{question}

Information of protein structure and related literature:

{info}

Task:

Analyzing the information of protein structure and related literature, whether these information could be helpful to answer the user question or not? And why? The output should clearly indicate whether the info could be used to answer the question or not using exactly 2–3 concise sentences.

Rules:

Do NOT output markdown, bullets, JSON, or extra sections. Keep the reason factual, direct. Do not offer further help.

Figure 11: The LLM prompt for relevance reasoning.

Input:

A reason has been generated for relevance of the abstract and protein structure info to the question: {reason}

Task:

Summarize the above reason to determine whether it indicates that the abstract and protein structure info is relevant to the question or not? Based on the summary of the reason, assign a relevance level:

- 0 = Irrelevant to the question.
- 1 = Relevant but only indirectly related. e.g., the reason indicates that the abstract and protein structure info could be adapted to answer the question.
- 2 = Directly relevant to answering the question.

Output format:

In the answer, first, write exactly 1–2 concise sentences that starts with either “The abstract is irrelevant to the question because ...”, “The abstract is somewhat related to the question because ...”, or “The abstract is directly relevant to the question because ...”. Then, on a new line, output exactly one of “SCORE: 0”, “SCORE: 1”, or “SCORE: 2” that matches your judgement.

Figure 12: **The LLM prompt for relevance judgment.**

Factual error in LLM-generated rationale

Question: Can you find the structures of Fhb7 and its variants involved in degrading deoxynivalenol?

LLM-generated Reason: Yes, the provided information is directly relevant as it describes the structure of the Fhb7 gene and its variants, along with their catalytic mechanisms and potential applications in degrading deoxynivalenol. The details on the structure and variants can help address the user’s query about Fhb7 and its variants involved in degrading deoxynivalenol.

LLM-generated Judgement: The abstract is directly relevant to the question because it describes the structure of the Fhb7 gene and its variants, along with their catalytic mechanisms and potential applications in degrading deoxynivalenol.

Human expert feedback: Although the answer is overall correct, there is an error in the terminology in the LLM-generated reason and judgement. The structure is of the Fhb7 enzyme, not the Fhb7 gene.

Figure 13: **Factual error in LLM-generated rationale on Human evaluation.**

Comparison between Qwen- and GPT4-generated rationale

Question: Can you find structures of O-methyltransferases that modify EGCG in tea?

Reason:

(Qwen-generated) The provided information includes the crystal structures of CsFAOMT1 and CsFAOMT2, which are O-methyltransferases involved in modifying EGCG, as well as details about their activities and key residues. This data can be directly used to answer the user's question regarding the structures of O-methyltransferases that modify EGCG in tea. However, the literature does not provide additional structures beyond CsFAOMT1 and CsFAOMT2. Therefore, while sufficient for the specific query, the scope is limited to these two enzymes.

(GPT4-generated) Yes, this information is directly useful to answer the question. It explicitly identifies two O-methyltransferases (CsFAOMT1 and CsFAOMT2) that modify EGCG and states that their crystal structures have been determined, along with functional insights. Therefore, it provides both the relevant protein identities and structural evidence needed to address the query.

Judgement:

(Qwen-generated) The information is directly relevant to the question because it provides the crystal structures of CsFAOMT1 and CsFAOMT2, which are O-methyltransferases involved in modifying EGCG.

(GPT4-generated) The information is directly relevant to the question because it identifies specific EGCG-modifying O-methyltransferases and provides their crystal structure information.

Relevance score:

(Qwen-generated) 2

(GPT4-generated) 2

Figure 14: Comparison between Qwen- and GPT4-generated rationale.