# Rethinking Text-based Protein Understanding: Retrieval or LLM?

Anonymous ACL submission


## Abstract

In recent years, protein-text models have gained significant attention for their potential in protein generation and understanding. Current approaches focus on integrating protein-related knowledge into large language models through continued pretraining and multi-modal alignment, enabling simultaneous comprehension of textual descriptions and protein sequences. Through a thorough analysis of existing model architectures and text-based protein understanding benchmarks, we identify significant data leakage issues present in current benchmarks. Moreover, conventional metrics derived from natural language processing fail to accurately assess the model's performance in this domain. To address these limitations, we reorganize existing datasets and introduce a novel evaluation framework based on biological entities. Motivated by our observation, we propose a retrieval-enhanced method, which significantly outperforms fine-tuned LLMs for protein-to-text generation and shows accuracy and efficiency in training-free scenarios. Our code and data will be available.

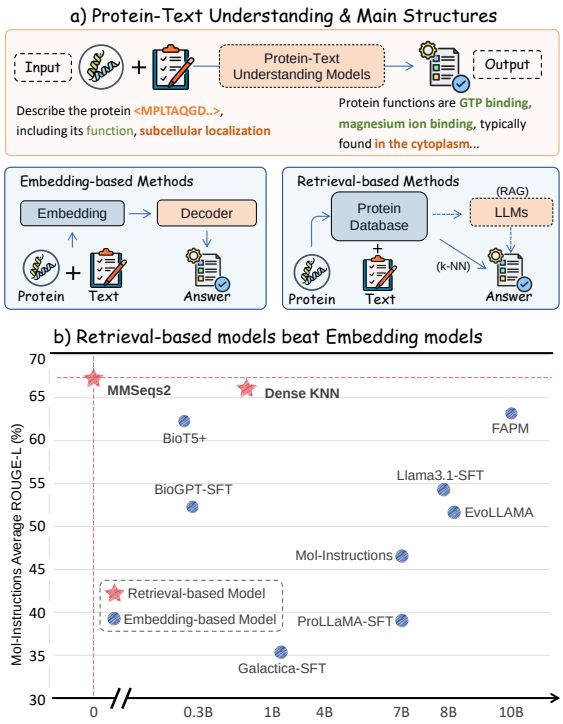




Figure 1: a) Protein understanding tasks, and LLM-based and retrieval-based methods for this task. b) The performance of existing methods in protein understanding tasks. Retrieval methods based on protein embeddings or sequences outperform LLM-based approaches.

## 1 Introduction

In recent years, large language models (LLMs) have achieved remarkable success across diverse domains (Brown et al., 2020; Touvron et al., 2023; Wei et al., 2022; Sallam, 2023). To further enhance the ability of LLMs in understanding domain-specific data (e.g., chemistry, biology), researchers have extended LLMs into the multi-modal domain, giving rise to multi-modal large language models (MLLMs) (Liu et al., 2023a; Cao et al., 2023; Maaz et al., 2023). Unlike traditional LLMs, which process single textual modality, MLLMs integrate multiple modalities, such as images, text, and graphs, by aligning them within a unified framework. This is typically accomplished using unimodal encoders for each input type and cross-modal projectors that map different modalities into a shared embedding space (Zhu et al., 2023). As a result, MLLMs enable sophisticated cross-modal reasoning, paving the way for applications like image-text understanding and molecule-function analysis (Li et al., 2023; Cao et al., 2024; Liu et al., 2023c).

The advances in MLLMs have led to significant developments in text-based protein understanding (Liu et al., 2024c, 2023b; Zhou et al., 2025; Lv et al., 2024; Liu et al., 2024b). The input of most tasks generally consists of a protein sequence paired with natural language text, while the output represents a functional description. Given that proteins can be represented as amino

acid sequences, they are naturally compatible with LLMs and can be processed in two primary ways (Fig. 1a): (1) directly as textual inputs to a language model in a decoder-only or encoder-decoder architechture (Fang et al., 2024; Lv et al., 2024; Luo et al., 2022; Pei et al., 2023), or (2) as an external modality, where specialized encoders first extract high-quality protein representations before alignment with LLMs for downstream tasks (Liu et al., 2024b). To evaluate such protein-text multimodal models, several benchmarks have been introduced, covering key tasks such as protein function prediction, subcellular localization, catalytic activity, and protein design (Fang et al., 2024). The model's performance is assessed by comparing the model's outputs against ground-truth annotations. While promising, existing methods raise key questions:

**Q1:** Can LLMs truly understand protein sequences?
**Q2:** Are current benchmarks suitable for protein understanding tasks?

To answer the two questions, we recall that retrieval methods have long served as fundamental approaches in protein tasks, leveraging sequence alignment and database search techniques to identify functional and structural similarities (Lee et al., 2007; Higdon et al., 2010; Eswar et al., 2006). These well-established methods provide a natural baseline for evaluating whether modern LLMs offer genuine advances in protein understanding or merely replicate retrieval paradigms through alternative mechanisms. We therefore tackle Q1 by first comparing traditional retrieval methods against LLMs. Surprisingly, our analysis reveals that *simple retrieval-based approaches* can match or even outperform current LLMs in protein sequence understanding, challenging the prevailing view that LLMs are inherently superior in this domain.

Through a comprehensive analysis of prevailing protein-text datasets and evaluation metrics, we identify *two key limitations* in current benchmarks: (1) significant data leakage issues that compromise benchmark validity, and (2) metrics that fail to adequately capture model performance on biologically meaningful tasks. We systematically evaluate both fine-tuning-based and alignment-based approaches across existing datasets, revealing that MLLMs primarily generate outputs by memorizing and reproducing similar input features. Motivated by our analysis and findings, we propose a more rigorous benchmark for text-based protein understanding and introduce an efficient protein knowledge retrieval system, which achieves the state-of-the-art performance in protein understanding by Retrieval-Augmented Protein Modeling (RAPM).
Our contributions are summarized as follows:

- We perform detailed evaluations on existing protein-text benchmarks, exposing issues related to data leakage and metric deficiencies.
- We systematically compare fine-tuned LLMs with retrieval-based methods, demonstrating that fine-tuning is unnecessary for specific tasks.
- We propose RAPM, a Retrieval-Augmented Protein Modeling framework with a dual-indexed protein knowledge database for enhancing LLM in protein understanding tasks.

## 2 Related Works

This section provides an overview of prior research focused on three key aspects: (1) applications of language models to protein science, (2) existing benchmarks for protein understanding, and (3) retrieval-based approaches in protein research.

### 2.1 Language Models in Protein

Protein language models (PLMs) have successfully adapted Transformer architectures to represent protein sequences as biological tokens, enabling advances in protein embedding (Hayes et al., 2025; Brandes et al., 2022; Elnaggar et al., 2021; Cao and Shen, 2021; Hu et al., 2024; Chen et al., 2024a,b) and design (Madani et al., 2023; Nijkamp et al., 2023; Lv et al., 2024; Ferruz et al., 2022). However, their inability to integrate textual information limits cross-modal reasoning, prompting recent work to develop mixed protein-text models. These approaches include *Contrastive Learning Methods* (Xu et al., 2023; Wu et al., 2024) that align protein sequences with text, *Bioknowledge-Augmented Pre-training* (Ferruz et al., 2022; Taylor et al., 2022; Lv et al., 2024; Pei et al., 2023; Zhuo et al., 2024; Liu et al., 2024b) that leverage large protein-text corpora, and *Multi-modal LLMs* (Liu et al., 2024c; Abdine et al., 2024; Wang et al., 2024; Chen et al., 2024b; Ma et al., 2025; Xiang et al., 2024) that project protein embeddings into LLM spaces. Despite progress, scaling these methods to larger LLMs remains challenging due to prohibitive retraining costs and catastrophic forgetting (Kirkpatrick et al., 2017), motivating research into parameter-efficient adaptation strategies.

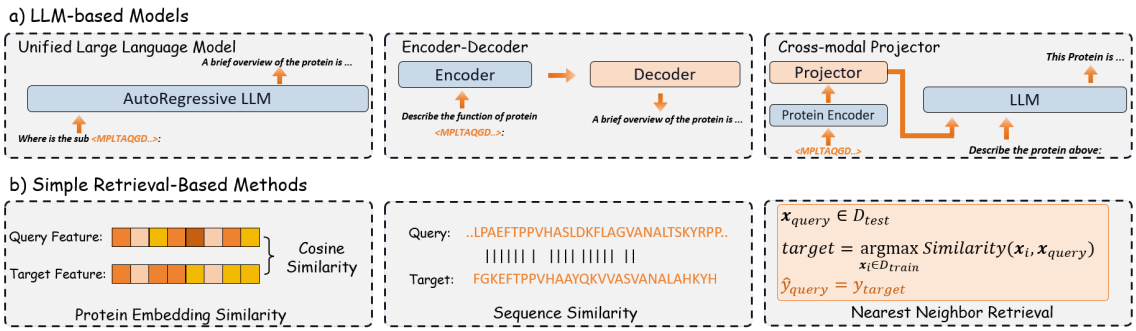



Figure 2: a) Three typical LLM-based approaches for text-based protein understanding. b) Simple nearest-neighbor based retrieval with protein embedding or sequence similarities.

### 2.2 Related Benchmarks

To advance research on protein-text hybrid models, several relevant benchmarks have been proposed. These benchmarks can be categorized into two types: (1) *Protein Captioning Tasks*, where only the protein sequence is input and a corresponding textual description is generated (e.g., the Swiss-Prot (Bairoch and Apweiler, 2000) and ProteinKG datasets (Zhang et al., 2022)), and (2) *Protein Question-Answering Tasks*, where both a protein sequence and a question are provided as input, and the model must generate an answer based on the protein and the query (e.g., Mol-Instructions (Fang et al., 2024), UniProtQA (Luo et al., 2024)), ProteinLMBench (Shen et al., 2024). To evaluate model performance on these benchmarks, researchers typically employ standard NLP metrics such as ROUGE (Lin, 2004), BLEU (Papineni et al., 2002), and METEOR (Banerjee and Lavie, 2005) to measure the similarity between predicted answers and ground-truth references. Models that perform well on these tasks can be applied to automated protein annotation, protein design, and protein property-related QA, thereby facilitating progress in the field.

### 2.3 Protein Related Retrieval-Based Methods

Retrieval-based approaches are fundamental to protein science, grounded in the well-established biological principle that sequence homology implies evolutionary conservation and functional similarity (Pearson, 2013). Single-sequence alignment approaches (Altschul et al., 1990; Buchfink et al., 2015; Steinegger and Söding, 2017; van Kempen et al., 2022) and multiple sequence alignment tools (Remmert et al., 2012; Johnson et al., 2010) are extensively used in bioinformatics for identifying highly homologous sequences. Many protein models utilize retrieval methods to assist downstream tasks, where AlphaFold2 (Jumper et al., 2021), MSA-Transformer (Rao et al., 2021), and RosettaFold (Baek et al., 2021) employ multiple sequence alignment results to aid property prediction or structure folding. Furthermore, retrieval-based approaches (Tan et al., 2024; Shaw et al., 2024; Sgarbossa and Bitbol, 2025; Jin et al., 2024; Li et al., 2024) have demonstrated the feasibility of using retrieval tools to enhance LLM-based predictions in protein research.

## 3 Analysis

Despite the broad usage of LLMs for protein understanding tasks, it remains unclear whether they truly understand protein sequences or simply memorize patterns. To answer this question, we conduct a systematic comparison between LLMs and retrieval-based methods, analyzing their performance and studying what LLMs actually learn.

### 3.1 Retrieval vs. LLM in Existing Tasks

We first evaluate both LLM-based and retrieval-based approaches on existing benchmarks, with the following experimental setup:

- **LLM-based approach** (Fig. 2a): After fine-tuning the model on the training dataset, we processed test samples using next-token prediction to generate answers.
- **Retrieval-based approach** (Fig. 2b): For each test sample, we retrieve the most similar protein sequence from the training set and use its annotation as the answer.

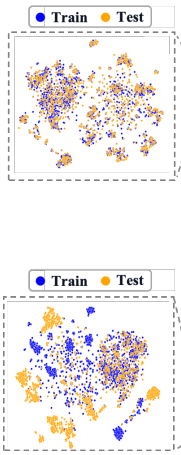
For LLM-based methods, we test a variety of model architectures, including 5 unimodal LLMs (Galactica-1.3B-SFT (Taylor et al., 2022), BioGPT-347M-SFT (Luo et al., 2022), ProLLaMA-7B-SFT (Lv et al., 2024), Mol-Instructions-7B (Fang et al., 2024), and Llama-3.1-8B-SFT (Dubey et al., 2024)), 1 encoder-decoder model (BioT5-Plus (Pei et al., 2023)), and 2 multi-modal LLMs

| Model | Arch. | SFT | Mol-Instructions/Protein (ROUGE-L) | | | | Avg. |
|---|---|---|---|---|---|---|---|
| | | | Function | Description | Domain | Catalytic | |
| Galactica-1.3B-*SFT* | Decoder-only | ✓ | 7.1 | 48.2 | 55.3 | 30.2 | 35.2 |
| BioGPT-347M-*SFT* | Decoder-only | ✓ | 50.9 | 49.7 | **55.4** | 54.2 | 52.5 |
| ProLLaMA-7B-*SFT* | Decoder-only | ✓ | 48.6 | 20.3 | 46.7 | 39.3 | 38.7 |
| Mol-Instructions-7B | Decoder-only | ✓ | 43.0 | 44.0 | 46.0 | 52.0 | 46.2 |
| Llama-3.1-8B-*SFT* | Decoder-only | ✓ | 52.1 | 54.2 | 51.2 | 59.6 | 54.2 |
| BioT5-Plus-252M | Encoder-Decoder | ✓ | 56.6 | 68.0 | 53.4 | 71.8 | 62.4 |
| EvoLLaMA-8.8B | MLP-Projector | ✓ | 48.0 | 50.0 | 50.0 | 60.0 | 52.0 |
| FAPM-10B | Q-Former | ✓ | **60.9** | 64.0 | 52.7 | **76.0** | 63.4 |
| MMSeqs2-Align | Retrieval | × | 60.2 | **76.0** | 55.2 | 75.6 | **66.7** |
| ESM2-Embedding | Retrieval | × | 59.7 | 74.9 | 54.5 | 75.2 | 66.0 |

Table 1: Performance comparison of LLM-based and retrieval-based methods across two text-based protein understanding benchmarks. **Arch.** denotes the model architecture. **SFT** indicates whether the model has undergone supervised fine-tuning on the training set. **Bold** indicates the best performance.

(EvoLLAMA-8.8B (Liu et al., 2024b) and FAPM-10B (Xiang et al., 2024)). For retrieval approaches, we employ MMSeqs2 (Steinegger and Söding, 2017) for sequence retrieval and ESM-2-650M (Lin et al., 2022) as the protein sequence encoder for embedding similarity. Our evaluation results (Table 1) highlight a key finding: **all current deep learning methods underperform retrieval-based approaches on these benchmarks**. We find that multi-modal LLMs merely match the performance of retrieval methods, while unimodal LLMs demonstrate poorer results. More critically, fine-tuning LLMs requires significant GPU resources, whereas ESM2-based retrieval only needs to compute protein embeddings, and MMSeqs2 retrieval completes 100 million comparisons within 1 minute using only one CPU.

To investigate why the retrieval-based methods beat LLMs, we first examine the data leakage in current benchmarks. The t-SNE visualization of the ESM2 embeddings of the proteins reveals samples forming distinct clusters with significant training-test contamination (Figure 3). We then quantify the level of data leakage by the percentage of test samples whose label can be directly obtained by retrieving the most similar sample (right table in Figure 3). It is easy to see that the leakage rates exceed 50% for most tasks, surpass 95% in some extreme cases. Such pervasive leakage suggests that models fine-tuned on these benchmarks predominantly memorize dataset-specific features rather than develop meaningful biological understanding.




| Benchmark | Leakage (%) |
|---|---|
| **Mol-Instructions:** | |
| - Protein Function | 54.5 |
| - General Function | 71.6 |
| - Domain Motif | 82.7 |
| - Catalytic Activity | 91.6 |
| **UniprotQA:** | |
| - Official Names | 69.1 |
| - Subcellular Locations | 86.3 |
| - Function | 75.7 |
| - Protein Family | 97.7 |
| **Protein Caption:** | |
| - Swiss-Prot | 47.2 |
| - OntoProtein | 5.0 |
| **Mol-Instructions-OOD:** | |
| - Protein Function | 0.1 |
| - General Function | 0.1 |
| - Domain Motif | 0.8 |
| - Catalytic Activity | 0.5 |

Figure 3: We evaluate the degree of data leakage in both existing benchmarks and OOD benchmarks. “Leakage" is defined as the probability that test set samples can directly retrieve similar samples with the same label from the training set.

### 3.2 What do LLMs Learn?

A fundamental question in text-based protein understanding is whether LLMs genuinely comprehend protein knowledge or simply act as sophisticated pattern matches based on input similarities. To address this, we perform a fine-grained comparative analysis between LLM-based and retrieval-based approaches. Specifically, we visualize and compare their performance across all test samples in the Protein Function task. This analysis enables us to distinguish whether LLMs predict properties based on protein sequence features or simply learn to replicate labels from similar training samples.

We compare the performance of the retrieval method to that of the LLM method under the measurement of ROUGE-L (Fig. 4). The majority of samples fall below the $y = x$ reference line and naturally separate into three clusters:

- Cluster 1: Both retrieval and LLM methods fail to predict the protein function.
- Cluster 2: The retrieval method correctly predicts the function while the LLM method fails
- Cluster 3: Both methods demonstrate competent performance.

Our analysis demonstrates that LLMs fine-tuned

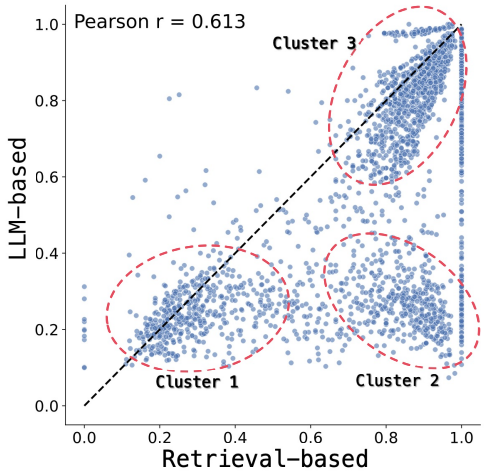



Figure 4: The ROUGE-L score distributions of retrieval-based methods versus LLM-based methods for all test samples in the General Function task.

for protein function prediction fail to surpass the performance of retrieval-based approaches for most test samples, indicating they primarily serve as a less effective substitute for retrieval approaches.

### 3.3 Is Retrieval a Silver Bullet?

We further investigate if traditional retrieval methods are a silver bullet as shown above. Using the entire training set as retrieval candidates, rather than creating separate pools for each subtask, leads to significant performance degradation for different methods (Table 2). Such task-specific pools are impractical in practice, given task diversity and continuous change. Furthermore, traditional methods often return only the top-1 match, preventing multi-source aggregation and lacking flexibility.

| Retrieval | Function | Description | Domain | Catalytic |
|---|---|---|---|---|
| MMSeqs2 | 60.2 | 76.0 | 55.2 | 75.6 |
| MMSeqs2$_{all}$ | 40.0(↓34%) | 25.4(↓67%) | 36.7(↓34%) | 37.6(↓50%) |
| ESM2-Embed | 59.7 | 74.9 | 54.5 | 75.2 |
| ESM2-Embed$_{all}$ | 38.7(↓35%) | 17.6(↓77%) | 36.7(↓33%) | 26.8(↓64%) |

Table 2: Performance degradation of retrieval methods with full corpus as candidates pool.

**Summary:** For practical usage, neither retrieval-based methods nor LLMs provide satisfactory protein understanding, which suggests a hybrid framework that synergistically combines the precision of retrieval with the reasoning capacity of LLMs.

## 4 Methods: Combine Retrieval & LLM and New Benchmark

To address this need, we develop Retrieval-Augmented Protein Modeling (RAPM) based on the Retrieval-Augmented Generation (RAG) paradigm. RAG is a proven approach for enhancing LLM factual accuracy and domain knowledge (Lewis et al., 2020). Our method leverages a Bio-Knowledge Database and contextual prompts to provide LLMs with explicit protein evidence during inference, thereby improving their understanding of biological information and addressing the memorization vs. reasoning tradeoff inherent in this problem.

### 4.1 Protein Knowledge Database Construction

For optimal performance, an accurate and efficient domain retrieval system relies on a carefully curated protein knowledge database. In our database, existing biological annotations are standardized into structured `[Protein, Annotation]` tuples, indexed by amino acid sequence and embeddings. The details of the database construction is shown in Fig. 5, including:

- **Protein Annotation Data Collection.** We gather protein-annotation data from InterPro (Blum et al., 2025), EC-GO (Gligorijević et al., 2021), and Mol-Instructions (Fang et al., 2024), extracting biological entity annotations using the method in Sec. 4.3.
- **Dual-key Indexing.** We build database indices with: (1) Sequence-based Indexing with inverted K-mer indices for heuristic retrieval, and (2) Feature-based Indexing using ESM-2 extracted protein features with the HNSW algorithm for efficient indexing.
- **Feature Aggregation.** For an annotation shared by multiple proteins, we compute the mean-pooled embedding of all proteins with that annotation. This aggregated feature ensures retrieval breadth while maintaining biological relevance.

### 4.2 Retrieval-Augmented Protein Modeling

Using a RAG-based approach, we provide LLMs with explicit protein evidence during inference for downstream queries. The overall retrieval and query pipeline is illustrated in Fig. 5. A critical component is our reformatted protein knowledge database, constructed by reorganizing existing protein annotations into standardized `[Protein, Annotation]` tuples. For precise and efficient retrieval, each entry is indexed using a dual-key mechanism, incorporating both amino acid sequence-based indexing with inverted K-mer indices for heuristic retrieval and feature-based index-

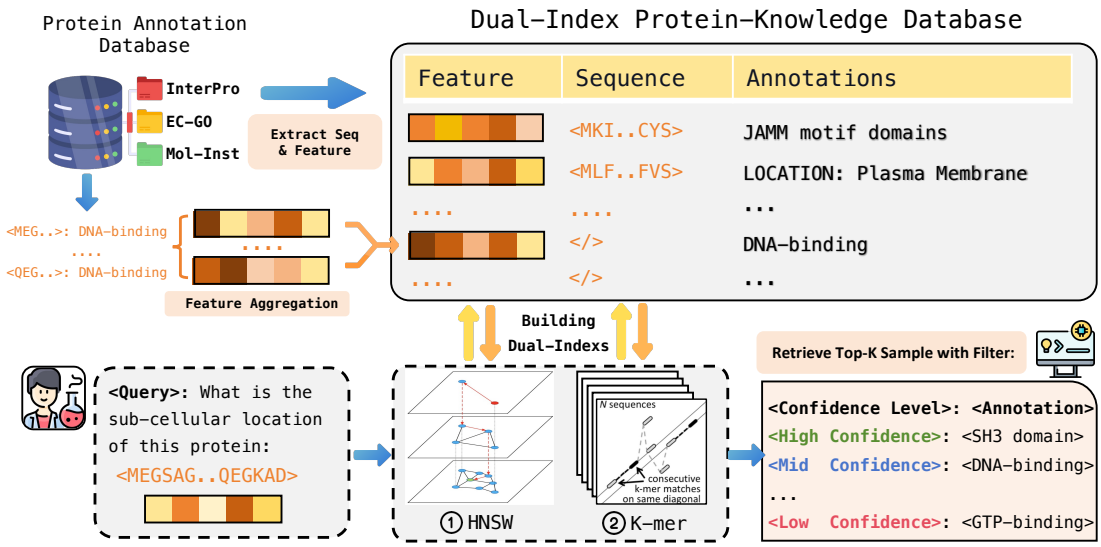



Figure 5: We collect protein-annotation pairs from existing protein annotation databases for the Protein-Knowledge Database construction. We extract dense features of proteins using a Protein Encoder and build database indices using two indexing methods. For entries sharing identical labels, we incorporate meta-features into the database. For downstream queries, we combine scores from both indices to retrieve the Top-K relevant entities, then construct retrieval-augmented prompts after quantizing sequence similarity into **High**, **Mid**, and **Low** confidence levels.

ing using ESM-2 embeddings with the HNSW algorithm for efficient similarity search (details about the indexing refer to Appendix C.3). To further improve retrieval breadth, especially for annotation labels shared by multiple proteins, we aggregate the features by computing the mean-pooled embedding of all proteins associated with a common annotation and indexing these aggregated features.

Formally, given a protein query $\mathcal{Q}$, we retrieve $K$ support data points $\{d_i\}_{i=1}^{K}$ from this structured database by ranking candidate entries based on a similarity score $s_i$. This score is a weighted combination of sequence and embedding similarity: $s_i = \alpha \cdot \text{sim}_{seq}(s, s_i) + (1 - \alpha) \cdot \text{sim}_{emb}(\mathbf{e}, \mathbf{e}_i)$, where $s$ and $s_i$ are protein sequences, $\mathbf{e}$ and $\mathbf{e}_i$ are the corresponding ESM-2 embeddings, and $\alpha$ is a weight parameter, currently set to 0.5. Instead of including full sequences in the prompt, each of the top-$K$ retrieved items is formatted as a concise `[Confidence, Annotation]` tuple. The Confidence level is derived from $s_i$ based on quantiles: $> 90\%$ as **High** , $90\% > s_i > 60\%$ as **Medium**, and $\leq 60\%$ as **Low**. The final input prompt $\mathcal{P}$ for the LLM is constructed by concatenating the query, few-shot examples, and the formatted retrieved items: $\mathcal{P} = \mathcal{Q} \oplus \mathcal{E}_{\text{few-shot}} \oplus \mathcal{R}_{1:k}$, where $\mathcal{R}_{1:k}$ represents the formatted top-$K$ entries and $\mathcal{E}_{\text{few-shot}}$ are demonstrations from the training dataset included to help the LLM understand the task format and reasoning. The LLM is then conditioned on $\mathcal{P}$ to predict the answer: $\hat{y} = \text{LLM}(\mathcal{P})$.

### 4.3 Novel Benchmark Proposal

Existing benchmarks (Sec 2.2) rely on NLP-derived metrics like token or sentence similarity, implicitly assuming equal importance for all answer components. This approach is fundamentally flawed for biological QA tasks. In protein-related questions, responses frequently include standardized template structures while the critical biological information is concentrated in just a few content words, which typically appear in the final portion of the answer. Consider the following example:

**Ground Truth:**
Upon evaluating your submitted sequence, our predictive algorithms suggest the presence of: ABC transporter domains

**Prediction 1 (True Answer):**
The sequence you provided has been analyzed for potential protein domains or motifs. The results are: ABC transporter domains

ROUGE-L = 0.27; BLEU = 0.04

**Prediction 2 (False Answer):**
Upon evaluating your submitted sequence, our predictive algorithms suggest the presence of: GGDEF, MHYT, EAL domains

ROUGE-L = 0.83; BLEU = 0.73

**Blue**: Mached Part **Red**: Mismatched Part

Although **Prediction 2** achieves much higher NLP metric scores, its information is biologically inaccurate. This discrepancy highlights a critical flaw of current evaluation metrics: they prioritize superficial text overlap, particularly in generic template

segments, and are insensitive to errors in the core biological content.

To address the data leakage in Sec 3.1 and metric validity issues identified above, we construct a new protein domain benchmark with novel task partitions to avoid data leakage and a BLEU-like metric specifically designed for biological entities.

**Data Unification and Clustering.** To mitigate data leakage, we restructure the protein-text data from Mol-Instruction through a two-step sequence-based partitioning strategy. (1) *Low similarity (Low-Sim) split*: Based on MMSeqs2 clustering with an 8:2 class split, mitigating general leakage. (2) *Out-of-Distribution (OOD)*: Filters Low-Sim by removing samples with answers present in the training set, preventing reliance on retrieval. We refer to this new benchmark dataset as the Mol-Instructions-OOD dataset. Detailed OOD dataset construction process can be seen in Appendix A.

**Metric Design for Biological QA.** Existing NLP metrics like ROUGE and BLEU are inadequate for biological QA, failing to capture biological nuances such as order-invariant entity lists by treating all tokens equally. To address this, we propose **Entity-BLEU**, a biological entity-focused metric analogous to BLEU. It works by first extracting biological entities from predictions and references using a knowledge base derived from databases like InterPro, EC-GO, and Mol-Instructions labels. A detailed biological entities can be seen in Appendix B.2. The standard BLEU score is then computed on these extracted entity sequences. Formally, Entity-BLEU is given by:

$$\text{Entity-BLEU} = \text{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right) \quad (1)$$

where BP is the brevity penalty term, $w_n$ are the weights for n-gram precision scores $p_n$ (typically $N \in \{1, 2, 3, 4\}$), and all calculations are performed on the extracted Bio-Entity sequences rather than raw text.

## 5 Experiments

This section comprehensively evaluates the proposed RAPM with the benchmark and metric proposed in Sec. 4.3. Experimental results demonstrate that retrieval-based methods show significant performance degradation in OOD-benchmarks. In ablation studies, we assess the impact of the number of items retrieved, database indexing methods, and prompt construction approaches for retrieval.

### 5.1 Experimental Setup

We evaluate four representative methodological approaches under our novel dataset splitting strategy described in Sec. 4.3:

1. **Fine-tuned LLMs:** These approaches fine-tune pre-trained language models (BioGPT, BioT5+, Llama) on the training set and inference on the test set.
2. **Retrieval-based methods:** For each test input, these approaches use the label of the retrieved most similar training sample as the predictions.
3. **Task-Prompted LLMs:** These approaches employ few-shot prompting frameworks with general-purpose LLMs (Llama-3.3, DeepSeek-V3, GPT-4.1), denoted by the subscript "few-shot", to generate predictions without retrieval augmentation.
4. **RAPM methods (Ours):** Our method retrieves top-K relevant samples from a protein knowledge database, constructs augmented prompts with these samples, and leverages general LLMs to generate context-aware responses.

We test four subtasks in Mol-Instructions-OOD, including "Protein Function", "Functional Description", "Domain/Motif" and "Catalytic Activity", using the standard NLP metrics (ROUGE-L) and our proposed metric, Entity-BLEU ($N = 2$). Detailed fine-tuning hyperparameters, retrieval settings, and RAPM prompt can be seen in Appendix C.2 and C.3. Note that for a fair comparison between RAPM methods and fine-tuned LLMs, we exclude all extra-training-set data during retrieval to prevent potential data leakage.

### 5.2 OOD Benchmark Performance

Table 3 summarizes the main results on existing benchmarks for four representative methodological approaches, and we observe the following key results: (1) When evaluated in OOD settings, the RAPM method achieves the highest Entity-BLEU scores, outperforming retrieval-based methods and demonstrating substantial improvements over fine-tuned and task-prompted LLMs. Beyond the superior performance, RAPM requires substantially fewer computational resources than fine-tuned LLMs and demonstrates a stronger capability to handle diverse tasks than retrieval-based methods. (2) When comparing ROUGE-L and Entity-BLEU scores of different methods, we observe a poor correlation between them, particularly

| Model | #Train Params | Retrieval | Function | | Description | | Domain | | Catalytic | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | E-BLEU | RG-L | E-BLEU | RG-L | E-BLEU | RG-L | E-BLEU | RG-L |
| **Fine-tuned LLM** | | | | | | | | | | |
| BioT5+ | 252M | None | 3.7 | 36.3 | 0.1 | 30.4 | 0.1 | 37.9 | 0.2 | 39.6 |
| Llama-3.2-1B | 1.0B | None | 16.2 | 43.4 | 4.4 | **43.0** | 3.9 | 43.0 | 1.8 | 44.0 |
| BioGPT | 347M | None | 6.8 | 41.6 | 0.3 | 34.3 | 1.1 | 44.3 | 0.6 | 43.9 |
| Galactica-1.3B | 1.3B | None | 14.4 | 43.2 | 1.2 | 32.6 | 3.8 | **45.3** | 0.9 | 43.3 |
| ProLLama-7B* | 19M | None | 6.3 | 39.8 | 0.5 | 30.2 | 1.0 | 41.7 | 0.4 | 41.8 |
| **Retrieval-based** | | | | | | | | | | |
| MMSeqs2 | N/A | Seq | 45.2 | 42.5 | 16.5 | 39.4 | 12.7 | 43.4 | 12.6 | 48.9 |
| ESM-2-650M | 650M | Emb. | 39.9 | **43.5** | 16.8 | 40.8 | 17.5 | 44.1 | 15.7 | **49.7** |
| **Task-prompted LLM** | | | | | | | | | | |
| Llama-3.3-70B$_{\text{w/ Few-shot}}$ | N/A | None | 0.3 | 29.5 | 1.0 | 27.1 | 0.1 | 36.8 | 0.1 | 44.5 |
| DeepSeek-V3$_{\text{w/ Few-shot}}$ | N/A | None | 0.2 | 28.3 | 0.3 | 25.5 | 0.0 | 35.3 | 0.9 | 19.6 |
| GPT-4.1$_{\text{w/ Few-shot}}$ | N/A | None | 0.1 | 31.9 | 0.2 | 26.1 | 0.1 | 38.7 | 0.1 | 40.8 |
| **RAPM-based** | | | | | | | | | | |
| Llama-3.3-70B$_{\text{w/ RAPM}}$ | N/A | Seq+Emb. | 41.5 | 37.5 | 16.9 | 25.4 | 7.3 | 11.1 | 23.5 | 44.6 |
| DeepSeek-V3$_{\text{w/ RAPM}}$ | N/A | Seq+Emb. | 35.3 | 31.2 | 13.8 | 24.4 | 8.8 | 17.9 | 16.3 | 21.0 |
| GPT-4.1$_{\text{w/ RAPM}}$ | N/A | Seq+Emb. | **46.6** | 27.4 | **20.9** | 30.1 | **32.0** | 22.5 | **38.9** | 46.4 |

Table 3: Performance across 4 protein understanding tasks, each evaluated with E-BLEU(Entity-BLEU) and RG-L(ROUGE-L). "*" means using LoRA (Hu et al., 2022) fine-tuning. **Bold** for best, underline for second best.

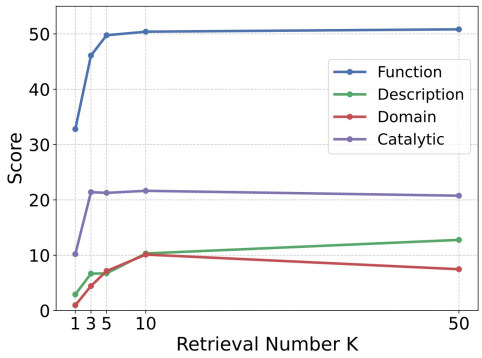



Figure 6: Impact of Retrieval Number $K$ in Entity-BLEU-2, randomly selected 256 samples for each task.

for fine-tuned LLMs which have high ROUGE-L scores and low Entity-BLEU scores. As discussed in Sec 3.2, we owe this to the fact that fine-tuned LLMs primarily focus on learning irrelevant response patterns rather than understanding protein sequences.

### 5.3 Ablation Studies

**Effect of Retrieved Sample Number.** We investigate the impact of varying the number of samples retrieved ($K \in \{1, 3, 5, 10, 50\}$) in the RAG pipeline. As shown in Figure 6, increasing K can improve model performance to some extent when the number of retrieved items is small. However, a larger K may introduce low-confidence incorrect samples, thereby degrading model performance.

**Effect of Database Index Methods.** We conduct ablation studies on different database components, specifically analyzing the impact of removing: (1) the Sequence Index, (2) the HNSW Index, (3) Feature aggregation, and (4) the Few-shot component in prompts. Note: When removing the HNSW Index, Meta-Features are also eliminated. The results (Table 4) show that removing any index significantly affects retrieval accuracy, while the observed ROUGE-L degradation confirms the importance of Few-Shot examples for guiding LLMs to learn proper response formats.

| RAG setting | E-BLEU-2 | E-BLEU-4 | Rouge-L |
|---|---|---|---|
| GPT-4.1 w. RAPM | 56.2 | 51.4 | 34.7 |
| - w/o. Seq Index | 50.0 (↓11%) | 44.0 (↓14%) | 31.1 (↓10%) |
| - w/o. HNSW Index | 44.8 (↓20%) | 41.2 (↓20%) | 32.3 (↓7%) |
| - w/o. Feature Aggr. | 51.7 (↓8%) | 47.2 (↓8%) | 31.0 (↓11%) |
| - w/o. Few-shot | 46.7 (↓17%) | 41.4 (↓19%) | 26.4 (↓24%) |

Table 4: Impact of Retrieval Indexing Methods. We randomly selected 256 samples from the "Protein Function" task and used GPT-4.1 to generate responses.

## 6 Conclusion and Future Works

In this work, we conduct a comprehensive analysis of existing text-based protein understanding benchmarks and methods, revealing that current benchmarks suffer from severe data leakage and that training-free retrieval-based approaches outperform fine-tuned LLM methods. Building on this, we introduce a novel hybrid benchmark and propose retrieval-augmented protein modelling. Our RAG method leverages both retrieval capabilities and LLMs' strengths to synthesize instruction-specific answers from retrieved evidence, achieving impressive results on OOD datasets.

Our findings highlight the effectiveness of retrieval methods for protein understanding and the need for rigorous benchmark and metric design. Future work will focus on deeper integration of retrieval and LLM methods, continuous improvements to benchmarks and metrics, and extension to other bio-entities (e.g., molecules, DNA, RNA).

## Limitations

This work primarily addresses text-based protein understanding. Extending our proposed RAG framework to other protein science tasks, such as de novo design or complex structure prediction, will require further investigation. The framework's effectiveness also heavily depends on the quality, coverage, and timeliness of the underlying protein knowledge database; incomplete or biased information in this resource can hinder performance, and maintaining an up-to-date database is an ongoing challenge. While our new benchmark and Entity-BLEU metric aim to improve evaluation rigor by mitigating data leakage and focusing on biological entities, assessing true biological understanding remains a multifaceted problem. Consequently, these tools, like any evaluation method, will benefit from continued validation, community adoption, and refinement. Furthermore, we plan to explore retrieval-augmented finetuning in future work, particularly with efficient LLMs, to further enhance domain-specific performance, an approach not investigated in this study.

## Potential Risks

A primary risk is that our framework could generate inaccurate biological insights. If unverified, these could misdirect research efforts. Over-reliance might also diminish critical human oversight. Furthermore, biases in the underlying data or LLMs could be amplified, leading to skewed predictions, especially for novel or less-studied proteins. The opaque nature of some LLMs can also make it hard to audit results or identify the root of errors. Finally, ensuring broad and equitable access to these powerful tools remains a challenge.

## References


Hadi Abdine, Michail Chatzianastasis, Costas Bouyioukos, and Michalis Vazirgiannis. 2024. Prot2text: Multimodal protein's function generation with gnns and transformers. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 10757–10765.

Stephen F Altschul, Warren Gish, Webb Miller, Eugene W Myers, and David J Lipman. 1990. Basic local alignment search tool. *Journal of molecular biology*, 215(3):403–410.

Minkyung Baek, Frank DiMaio, Ivan Anishchenko, Justas Dauparas, Sergey Ovchinnikov, Gyu Rie Lee, Jue Wang, Qian Cong, Lisa N Kinch, R Dustin Schaeffer, and 1 others. 2021. Accurate prediction of protein structures and interactions using a three-track neural network. *Science*, 373(6557):871–876.

Amos Bairoch and Rolf Apweiler. 2000. The swiss-prot protein sequence database and its supplement trembl in 2000. *Nucleic acids research*, 28(1):45–48.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Matthias Blum, Antonina Andreeva, Laise Cavalcanti Florentino, Sara Rocio Chuguransky, Tiago Grego, Emma Hobbs, Beatriz Lazaro Pinto, Ailsa Orr, Typhaine Paysan-Lafosse, Irina Ponamareva, and 1 others. 2025. Interpro: the protein sequence classification resource in 2025<? mode longmeta?>. *Nucleic Acids Research*, 53(D1):D444–D456.

Nadav Brandes, Dan Ofer, Yam Peleg, Nadav Rappoport, and Michal Linial. 2022. Proteinbert: a universal deep-learning model of protein sequence and function. *Bioinformatics*, 38(8):2102–2110.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Benjamin Buchfink, Chao Xie, and Daniel H Huson. 2015. Fast and sensitive protein alignment using diamond. *Nature methods*, 12(1):59–60.

He Cao, Zijing Liu, Xingyu Lu, Yuan Yao, and Yu Li. 2023. Instructmol: Multi-modal integration for building a versatile and reliable molecular assistant in drug discovery. *arXiv preprint arXiv:2311.16208*.

He Cao, Yanjun Shao, Zhiyuan Liu, Zijing Liu, Xiangru Tang, Yuan Yao, and Yu Li. 2024. Presto: progressive pretraining enhances synthetic chemistry outcomes. *arXiv preprint arXiv:2406.13193*.

Yue Cao and Yang Shen. 2021. Tale: Transformer-based protein function annotation with joint sequence–label embedding. *Bioinformatics*, 37(18):2825–2833.

Bo Chen, Xingyi Cheng, Pan Li, Yangli-ao Geng, Jing Gong, Shen Li, Zhilei Bei, Xu Tan, Boyan Wang, Xin Zeng, and 1 others. 2024a. xtrimopglm: unified 100b-scale pre-trained transformer for deciphering the language of protein. *ArXiv preprint*, abs/2401.06199.

Zhiyuan Chen, Tianhao Chen, Chenggang Xie, Yang Xue, Xiaonan Zhang, Jingbo Zhou, and Xiaomin Fang. 2024b. Unifying sequences, structures, and descriptions for any-to-any protein generation with the large multimodal model helixprotx. *arXiv preprint arXiv:2407.09274*.

UniProt Consortium. 2019. Uniprot: a worldwide hub of protein knowledge. *Nucleic acids research*, 47(D1):D506–D515.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, and 1 others. 2024. The llama 3 herd of models. *ArXiv preprint*, abs/2407.21783.

Ahmed Elnaggar, Michael Heinzinger, Christian Dallago, Ghalia Rehawi, Yu Wang, Llion Jones, Tom Gibbs, Tamas Feher, Christoph Angerer, Martin Steinegger, and 1 others. 2021. Prottrans: Toward understanding the language of life through self-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 44(10):7112–7127.

ESM Team. 2024. Esm cambrian: Revealing the mysteries of proteins with unsupervised learning.

Narayanan Eswar, Ben Webb, Marc A Marti-Renom, MS Madhusudhan, David Eramian, Min-yi Shen, Ursula Pieper, and Andrej Sali. 2006. Comparative protein structure modeling using modeller. *Current protocols in bioinformatics*, 15(1):5–6.

Yin Fang, Xiaozhuan Liang, Ningyu Zhang, Kangwei Liu, Rui Huang, Zhuo Chen, Xiaohui Fan, and Huajun Chen. 2024. Mol-instructions: A large-scale biomolecular instruction dataset for large language models. In *The Twelfth International Conference on Learning Representations*. OpenReview.net.

Noelia Ferruz, Steffen Schmidt, and Birte Höcker. 2022. Protgpt2 is a deep unsupervised language model for protein design. *Nature communications*, 13(1):4348.

Vladimir Gligorijević, P Douglas Renfrew, Tomasz Kosciolek, Julia Koehler Leman, Daniel Berenberg, Tommi Vatanen, Chris Chandler, Bryn C Taylor, Ian M Fisk, Hera Vlamakis, and 1 others. 2021. Structure-based protein function prediction using graph convolutional networks. *Nature communications*, 12(1):3168.

Thomas Hayes, Roshan Rao, Halil Akin, Nicholas J Sofroniew, Deniz Oktay, Zeming Lin, Robert Verkuil, Vincent Q Tran, Jonathan Deaton, Marius Wiggert, and 1 others. 2025. Simulating 500 million years of evolution with a language model. *Science*, page eads0018.

Roger Higdon, Brenton Louie, and Eugene Kolker. 2010. Modeling sequence and function similarity between proteins for protein functional annotation. In *Proceedings of the 19th ACM International Symposium on High Performance Distributed Computing*, pages 499–502.

Bozhen Hu, Cheng Tan, Yongjie Xu, Zhangyang Gao, Jun Xia, Lirong Wu, and Stan Z. Li. 2024. Protgo: Function-guided protein modeling for unified representation learning. In *Advances in Neural Information Processing Systems*, volume 37, pages 88581–88604. Curran Associates, Inc.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. Lora: Low-rank adaptation of large language models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Mingyu Jin, Haochen Xue, Zhenting Wang, Boming Kang, Ruosong Ye, Kaixiong Zhou, Mengnan Du, and Yongfeng Zhang. 2024. ProLLM: Protein chain-of-thoughts enhanced LLM for protein-protein interaction prediction. In *First Conference on Language Modeling*.

L Steven Johnson, Sean R Eddy, and Elon Portugaly. 2010. Hidden markov model speed heuristic and iterative hmm search procedure. *BMC bioinformatics*, 11:1–8.

John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, and 1 others. 2021. Highly accurate protein structure prediction with alphafold. *nature*, 596(7873):583–589.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, and 1 others. 2017. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526.

David Lee, Oliver Redfern, and Christine Orengo. 2007. Predicting protein function from sequence and structure. *Nature reviews molecular cell biology*, 8(12):995–1005.

Patrick S. H. 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. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR.

Pan Li, Xingyi Cheng, Le Song, and Eric P Xing. 2024. Retrieval augmented protein language models for protein structure prediction. *bioRxiv*, pages 2024–12.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Allan dos Santos Costa, Maryam Fazel-Zarandi, Tom Sercu, Sal Candido, and 1 others. 2022. Language models of protein sequences at the scale of evolution enable accurate structure prediction. *bioRxiv*.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, and 1 others. 2024a. Deepseek-v3 technical report. *ArXiv preprint*, abs/2412.19437.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023a. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916.

Nuowei Liu, Changzhi Sun, Tao Ji, Junfeng Tian, Jianxin Tang, Yuanbin Wu, and Man Lan. 2024b. Evollama: Enhancing llms' understanding of proteins via multimodal structure and sequence representations. *arXiv preprint arXiv:2412.11618*.

Shengchao Liu, Yanjing Li, Zhuoxinran Li, Anthony Gitter, Yutao Zhu, Jiarui Lu, Zhao Xu, Weili Nie, Arvind Ramanathan, Chaowei Xiao, and 1 others. 2023b. A text-guided protein design framework. *ArXiv preprint*, abs/2302.04611.

Zhiyuan Liu, Sihang Li, Yanchen Luo, Hao Fei, Yixin Cao, Kenji Kawaguchi, Xiang Wang, and Tat-Seng Chua. 2023c. MolCA: Molecular graph-language modeling with cross-modal projector and uni-modal adapter. *arXiv preprint arXiv:2310.12798*.

Zhiyuan Liu, An Zhang, Hao Fei, Enzhi Zhang, Xiang Wang, Kenji Kawaguchi, and Tat-Seng Chua. 2024c. ProtT3: Protein-to-text generation for text-based protein understanding. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5949–5966, Bangkok, Thailand. Association for Computational Linguistics.

Renqian Luo, Liai Sun, Yingce Xia, Tao Qin, Sheng Zhang, Hoifung Poon, and Tie-Yan Liu. 2022. Biogpt: generative pre-trained transformer for biomedical text generation and mining. *Briefings in bioinformatics*, 23(6):bbac409.

Yizhen Luo, Jiahuan Zhang, Siqi Fan, Kai Yang, Massimo Hong, Yushuai Wu, Mu Qiao, and Zaiqing Nie. 2024. Biomedgpt: An open multimodal large language model for biomedicine. *IEEE Journal of Biomedical and Health Informatics*.

Liuzhenghao Lv, Zongying Lin, Hao Li, Yuyang Liu, Jiaxi Cui, Calvin Yu-Chian Chen, Li Yuan, and Yonghong Tian. 2024. Prollama: A protein large language model for multi-task protein language processing. *arXiv e-prints*, pages arXiv–2402.

Zicheng Ma, Chuanliu Fan, Zhicong Wang, Zhenyu Chen, Xiaohan Lin, Yanheng Li, Shihao Feng, Jun Zhang, Ziqiang Cao, and Yi Qin Gao. 2025. Prottex: Structure-in-context reasoning and editing of proteins with large language models. *arXiv preprint arXiv:2503.08179*.

Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Shahbaz Khan. 2023. Video-chatgpt: Towards detailed video understanding via large vision and language models. *arXiv preprint arXiv:2306.05424*.

Ali Madani, Ben Krause, Eric R Greene, Subu Subramanian, Benjamin P Mohr, James M Holton, Jose Luis Olmos, Caiming Xiong, Zachary Z Sun, Richard Socher, and 1 others. 2023. Large language models generate functional protein sequences across diverse families. *Nature echnology*, 41(8):1099–1106.

Erik Nijkamp, Jeffrey A Ruffolo, Eli N Weinstein, Nikhil Naik, and Ali Madani. 2023. Progen2: exploring the boundaries of protein language models. *Cell systems*, 14(11):968–978.

OpenAI. 2025. Introducing GPT-4.1 in the API. `https://openai.com/blog/gpt-4-1-api`.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

William R Pearson. 2013. An introduction to sequence similarity ("homology") searching. *Current protocols in bioinformatics*, 42(1):3–1.

Qizhi Pei, Lijun Wu, Kaiyuan Gao, Xiaozhuan Liang, Yin Fang, Jinhua Zhu, Shufang Xie, Tao Qin, and Rui Yan. 2024. BioT5+: Towards generalized biological understanding with IUPAC integration and multi-task tuning. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 1216–1240, Bangkok, Thailand. Association for Computational Linguistics.

Qizhi Pei, Wei Zhang, Jinhua Zhu, Kehan Wu, Kaiyuan Gao, Lijun Wu, Yingce Xia, and Rui Yan. 2023. BioT5: Enriching cross-modal integration in biology with chemical knowledge and natural language associations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 1102–1123, Singapore. Association for Computational Linguistics.

Roshan Rao, Jason Liu, Robert Verkuil, Joshua Meier, John F. Canny, Pieter Abbeel, Tom Sercu, and Alexander Rives. 2021. MSA transformer. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 8844–8856. PMLR.

Michael Remmert, Andreas Biegert, Andreas Hauser, and Johannes Söding. 2012. Hhblits: lightning-fast iterative protein sequence searching by hmm-hmm alignment. *Nature methods*, 9(2):173–175.

Malik Sallam. 2023. The utility of ChatGPT as an example of large language models in healthcare education, research and practice: Systematic review on the future perspectives and potential limitations. *MedRxiv*, pages 2023–02.

Damiano Sgarbossa and Anne-Florence Bitbol. 2025. Rag-esm: Improving pretrained protein language models via sequence retrieval. *bioRxiv*, pages 2025–04.

Peter Shaw, Bhaskar Gurram, David Belanger, Andreea Gane, Maxwell L Bileschi, Lucy J Colwell, Kristina Toutanova, and Ankur P Parikh. 2024. Protex: A retrieval-augmented approach for protein function prediction. *bioRxiv*, pages 2024–05.

Yiqing Shen, Zan Chen, Michail Mamalakis, Luhan He, Haiyang Xia, Tianbin Li, Yanzhou Su, Junjun He, and Yu Guang Wang. 2024. A fine-tuning dataset and benchmark for large language models for protein understanding. In *2024 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 2390–2395. IEEE.

Martin Steinegger and Johannes Söding. 2017. Mmseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nature biotechnology*, 35(11):1026–1028.

Yang Tan, Ruilin Wang, Banghao Wu, Liang Hong, and Bingxin Zhou. 2024. Retrieval-enhanced mutation mastery: Augmenting zero-shot prediction of protein language model. *ArXiv preprint*, abs/2410.21127.

Ross Taylor, Marcin Kardas, Guillem Cucurull, Thomas Scialom, Anthony Hartshorn, Elvis Saravia, Andrew Poulton, Viktor Kerkez, and Robert Stojnic. 2022. Galactica: A large language model for science. *ArXiv preprint*, abs/2211.09085.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, and 1 others. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Michel van Kempen, Stephanie S Kim, Charlotte Tumescheit, Milot Mirdita, Cameron LM Gilchrist, Johannes Söding, and Martin Steinegger. 2022. Foldseek: fast and accurate protein structure search. *Biorxiv*, pages 2022–02.

Chao Wang, Hehe Fan, Ruijie Quan, and Yi Yang. 2024. Protchatgpt: Towards understanding proteins with large language models. *ArXiv preprint*, abs/2402.09649.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, and 1 others. 2022. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*.

Kevin E Wu, Howard Chang, and James Zou. 2024. Proteinclip: enhancing protein language models with natural language. *bioRxiv*, pages 2024–05.

Wenkai Xiang, Zhaoping Xiong, Huan Chen, Jiacheng Xiong, Wei Zhang, Zunyun Fu, Mingyue Zheng, Bing Liu, and Qian Shi. 2024. Fapm: functional annotation of proteins using multimodal models beyond structural modeling. *Bioinformatics*, 40(12):btae680.

Minghao Xu, Xinyu Yuan, Santiago Miret, and Jian Tang. 2023. Protst: Multi-modality learning of protein sequences and biomedical texts. In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 38749–38767. PMLR.

Ningyu Zhang, Zhen Bi, Xiaozhuan Liang, Siyuan Cheng, Haosen Hong, Shumin Deng, Qiang Zhang, Jiazhang Lian, and Huajun Chen. 2022. Ontoprotein: Protein pretraining with gene ontology embedding. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Xibin Zhou, Chenchen Han, Yingqi Zhang, Jin Su, Kai Zhuang, Shiyu Jiang, Zichen Yuan, Wei Zheng, Fengyuan Dai, Yuyang Zhou, and 1 others. 2025. Decoding the molecular language of proteins with evolla. *bioRxiv*, pages 2025–01.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. 2023. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*.

Le Zhuo, Zewen Chi, Minghao Xu, Heyan Huang, Jianan Zhao, Heqi Zheng, Conghui He, Xian-Ling Mao, and Wentao Zhang. 2024. ProtLLM: An interleaved protein-language LLM with protein-as-word pre-training. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8950–8963, Bangkok, Thailand. Association for Computational Linguistics.

## A Additional Dataset and Details

**Dataset Statistics** In this section, we introduce detailed dataset statistics information, such as data splitting and scale in Table 5, including the existing dataset and the newly proposed OOD dataset.

**OOD dataset constructions** The construction of OOD datasets involves three key steps:

| Dataset / Task | Train | Validation | Test |
|---|---|---|---|
| **Swiss-Prot** | 430,595 | 10,000 | 10,000 |
| **ProteinKG25** | 422,315 | 10,000 | 10,000 |
| **Mol-Instructons (Protein)** | | | |
| - Protein Function | 110,689 | – | 3,494 |
| - Catalytic Activity | 51,573 | – | 1,601 |
| - Domain / Motif | 43,700 | – | 1,400 |
| - Functional Desc. | 83,939 | – | 2,633 |
| **OOD Datasets** | | | |
| - Protein Function | 108,696 | – | 5,487 |
| - Domain/Motif | 42,368 | – | 2,732 |
| - Catalytic Activity | 51,187 | – | 1,987 |
| - General Function | 82,275 | – | 4,297 |

Table 5: Dataset statistics: number of samples for each task in the three corpora.

1. Sequence Clustering: We cluster all sequences from both training and test sets using MMseqs2 with the command "`mmseqs easy-cluster -cluster-mode 0 -c 0 -e 1e5 -single-step-clustering -min-seq-id 0 [all_seqs]`", generating distinct sequence clusters.

2. Cluster Partitioning: All clusters are randomly split into training (80%) and test (20%) clusters, with sequences from these clusters forming the respective training and test sets.

3. Leakage Elimination: To prevent test-set samples from having direct training-set answers, we use "`mmseqs easy-search -max-accept 1 [query_db] [target_db]`" to query the most similar training-set protein for each test-set protein. If a test-set protein shares a label with its retrieved training-set counterpart, it is reallocated to the training set.

We repeat Step 3 twice to ensure minimal label overlap between the test set and retrieval results, yielding the final OOD dataset.

# B Methods Details

This section presents the methodology details, including the implementation of simple retrieval baselines and the construction of the protein knowledge database.

## B.1 Details of Simple Retrieval Methods

To establish a straightforward baseline for the Text-based Protein Understanding task, we employ the simple retrieval approach using MMSeqs2, a widely adopted sequence alignment toolkit. Specifically, we utilize the `easy-search` mode of MMSeqs2 with the parameters `-e 1e5 -max-accept 1`. Here, `-e 1e5` sets a permissive E-value threshold to ensure the retrieval mechanism is recall-oriented, and `-max-accept 1` restricts the output to only the top candidate for each query. For every sample in the test dataset, we retrieve from the training dataset the most similar protein sequence based on alignment scores. The functional annotation (label) of the retrieved protein is then assigned as the predicted label for the query. This simple nearest-neighbor baseline is effective for assessing the upper bound of sequence-based function transfer.

## B.2 Details of Protein Knowledge Database Construction

To construct a comprehensive protein knowledge database to support downstream tasks, we divide our methodology into data collection and efficient indexing phases.

**Data Collection** We integrated annotations from three prominent sources: InterPro, EC-GO, and Mol-Instructions. For InterPro, we selectively used only sequences annotated via Swiss-Prot curation, resulting in a high-quality subset with 573,230 sequences. In the EC-GO database, labels corresponding to the enzyme classification (EC) and gene ontology (GO) were merged into a unified text-based annotation to capture functional and process aspects simultaneously. For Mol-Instructions, only the "meta-data" field is retained as the annotation, disregarding the original class labels, to emphasize naturalistic, descriptive phrasing of protein functions.

**HNSW-Index Construction** We implement HNSW for efficient ANN search over protein sequence embeddings $\mathcal{V} = \{\mathbf{v}_i\}_{i=1}^{N} \subset \mathbb{R}^{1280}$ from ESM2-650M. For sequences sharing functional annotations, we compute aggregated embeddings:

$$\mathbf{v}_y^{\text{agg}} = \frac{1}{|S_y|} \sum_{x \in S_y} \mathbf{v}_x, \quad S_y = \{x | \text{label}(x) = y\}$$

The index construction involves three key steps:

1. **Layer Assignment**: Each vector $\mathbf{v}$ is assigned to layer $l$ via:

$$l = \lfloor -\ln(\text{rand}(0,1)) \cdot m_L \rfloor, \quad m_L = 1/\ln(M)$$

2. **Hierarchical Insertion**: For each layer $l$ from top to bottom:

$$\mathcal{E}_l(\mathbf{v}) = \text{argmin}_{\mathbf{u} \in \mathcal{N}_l(\mathbf{v}), |\mathcal{E}|=M} \|\mathbf{v} - \mathbf{u}\|_2$$

where $\mathcal{N}_l(\mathbf{v})$ contains $ef_construction$ nearest neighbors.

3. **Small-World Guarantee**: Connections maintain:

$$\|\mathbf{v} - \mathbf{u}\|_2 \leq r_l(\mathbf{v}), \quad \forall \mathbf{u} \in \mathcal{E}_l(\mathbf{v})$$

The resulting structure achieves $O(\log N)$ search time with $O(N \cdot M)$ space complexity, balancing accuracy and efficiency for protein function retrieval. Key advantages include multi-layer acceleration, optimized neighborhood connectivity, and adaptive radius control.

**MMSeqs-Index Construction** For indexing at the sequence level, we utilize the k-mer based inverted indexing scheme provided by MMSeqs2. Each protein sequence is decomposed into overlapping k-mers (subsequences of fixed length $k$). The index is then constructed as a mapping from each unique k-mer to the list of all sequences containing it. The search query is similarly tokenized and candidate sequences are retrieved by aggregating all records sharing at least one k-mer with the query. Formally, letting $\mathcal{K}(q)$ denote the set of $k$-mers in query $q$, the candidate set is given by

$$C(q) = \bigcup_{k' \in \mathcal{K}(q)} \text{Index}[k']$$

This approach provides a highly efficient solution for large-scale substring and approximate matching, and is particularly effective for detecting local similarities.

**Bio-Entity List Collection** For the evaluation metric Entity-BLEU, we construct a domain-specific entity list. This list is derived by extracting all distinctive biological terms from annotations in EC-GO, InterPro, and the "metadata" field of Mol-Instructions. The curated entities span numerous key biological domains, meticulously categorized into areas such as *Molecular Biology and Biochemistry*(including nucleic acids like *DNA*, proteins like *polymerase*, and metabolites like *ATP*), *Cell Biology* (covering organelles like *mitochondrion*, processes like *apoptosis*), *Bioenergetics and Metabolism* (e.g., *glycolysis*, *ATP synthase*), *Genetics and Genomics* (terms like *gene*, *codon*, *RNA polymerase*), *Molecular Interactions and Signaling* (e.g., *receptor*, *MAPK pathway*), *Developmental Biology*, *Immunology*, *Plant Biology*, and *Microbiology*. Resulting in 11,341 unique terms across 10 different main categories. This ensures a broad yet granular representation of biological concepts.

To enhance specificity, we remove ambiguous general words (e.g., "domain") and common stopwords (e.g., "for", "to"). This list underlies the Entity-BLEU metric, which rewards lexical overlap specifically on content-relevant biomedical entities, providing a fine-grained measurement of functional description quality.

# C Experimental Details

## C.1 Comparison Baselines

For fair comparison, we select the following baselines:

- **BioT5+** (Pei et al., 2024): A T5 architecture model for biological and chemical tasks, improving on BioT5 (Luo et al., 2022) with IUPAC integration, multi-task tuning, and better numerical processing.

- **Llama-3.2-1B** (Dubey et al., 2024): A recent multilingual LLM with strong generalization ability, tested in both zero-shot and fine-tuned configurations.

- **BioGPT** (Luo et al., 2022): A domain-specific generative Transformer language model pretrained on large-scale biomedical literature. BioGPT achieves strong performance on six biomedical NLP tasks. Case studies demonstrate BioGPT's ability to generate fluent biomedical text descriptions.

- **Galactica** (Taylor et al., 2022): The Galactica models are trained on a large-scale scientific corpus. The models are designed to perform scientific tasks, including but not limited to citation prediction, scientific QA, mathematical reasoning, summarization, document generation, molecular property prediction, and entity extraction. The models were developed by the Papers with Code team at Meta AI to study the use of language models for the automatic organization of science.

- **ProLlama** (Lv et al., 2024): ProLLaMA is a protein large language model, designed for multi-task protein language processing. It employs a two-stage training framework, incorporating Low-Rank Adaptation (LoRA) and Protein Vocabulary Pruning (PVP) to enhance efficiency.

ProLLaMA achieves strong performance in protein sequence generation and property prediction tasks.

- **MMSeqs2 Retrieval** (Steinegger and Söding, 2017): MMseqs2 (Many-against-Many sequence searching) is a high-performance software suite designed for the rapid and sensitive retrieval of homologous protein or nucleotide sequences from large-scale databases. Its retrieval module employs a multi-stage search pipeline—comprising fast k-mer matching, ungapped alignment, and vectorized Smith-Waterman alignment—to efficiently identify relevant sequences while minimizing computational overhead. This approach enables MMseqs2 to achieve sensitivity comparable to BLAST, but with significantly enhanced speed.

- **ESM-2 Embedding KNN** (ESM Team, 2024): This method retrieves homologous proteins by performing K-Nearest Neighbors (KNN) search on fixed-length embeddings generated by the ESM-2 language model. By averaging residue-level embeddings, each protein sequence is represented as a single vector, enabling efficient similarity searches using cosine distance metrics. This embedding-based approach facilitates rapid identification of functionally similar proteins, even in cases of low sequence identity.

### C.2 Hyper-parameters

**Finetune Settings.** The LLM fine-tuning process utilizes hyperparameters shown in Table 6. Training is conducted with DeepSpeed-enabled distributed GPUs, utilizing mixed-precision (bf16) and memory optimization techniques. For LLMs over 7 billion parameters, LoRA is used to significantly reduce memory requirements by freezing the majority of model weights and introducing lightweight low-rank updates. The cosine learning rate schedule with warm-up ensures stable convergence.

### C.3 RAG Inference Settings

We standardize the inference hyperparameters across all evaluated LLMs (GPT-4.1, LLaMA3-70B, and DeepSeek-V3) to ensure fair comparison. The configuration is optimized for retrieval-augmented generation tasks:

| Hyper-parameter | Value |
|---|---|
| Learning rate for LoRA | 1e-4 |
| Learning rate for full parameter | 4e-5 |
| Batch size per device | 2 |
| Gradient accumulation steps | 8 |
| LoRA rank | 8 |
| LoRA $\alpha$ | 32 |
| LoRA dropout | 0.05 |
| Max sequence length | 2048 |
| Number of epochs | 2 |
| Optimizer | AdamW |
| LR scheduler type | Cosine |
| Warm-up ratio | 0.1 |
| Weight decay | 1e-2 |
| Mixed precision | bf16 |
| Gradient checkpointing | Enabled |
| Devices | 4 * RTX-A6000 |
| Approximate training duration | 15 hours /task |
| DeepSpeed config | Zero-2 |

Table 6: Hyper-parameter settings for finetuning.

| Hyper-Parameter | Value |
|---|---|
| Temperature | 0.7 |
| Top-p | 0.9 |
| Max tokens | 2048 |
| Frequency penalty | 0 |
| Presence penalty | 0 |

Table 7: Inference parameters for all evaluated LLMs. Identical settings are maintained across models except where architectural differences require variation.

## D Case Study

### D.1 Case Study for Data Leakage

This part provides specific examples illustrating the data leakage observed in existing protein-text benchmarks, as discussed in Sec 3.1. Table 8 presents two representative pairs of entries, each consisting of a protein from a test dataset and a highly similar protein from its corresponding training dataset. Crucially, for both pairs, the associated functional or domain information is identical.

For instance, the first pair shows test protein UniProt A4WLK4 and training protein UniProt A0A823T310 possessing significantly similar amino acid sequences and precisely the same detailed functional annotation (6,7-dimethyl-8-ribityllumazine synthase activity, etc.). Similarly, the second pair, UniProt Q27996 (test) and UniProt P51782 (training), exhibits high sequence homology directly correlated with an identical domain annotation ("Contains C-type lysozyme domains").

This direct correspondence between highly simi-

## Prompt example for inference (K=10)

You are given a protein sequence and two lists of related proteins retrieved from a database.
Instruction: Examine the protein sequence below and provide a prediction on its subcellular localization within the cell:

**Protein sequence: APQEPNQFQLLKYH**

**Retrieved proteins by feature similarity:**

<High Confidence>: 'chloroplast thylakoid',
<High Confidence>: 'carbohydrate binding'
<High Confidence>: 'FMRFamides and FMRFamide-like peptides are neuropeptides.'
<High Confidence>: 'chloroplast thylakoid'
<High Confidence>: 'extracellular region | toxin activity',
<High Confidence>: 'Has antibacterial activity against the Gram-positive bacteria L.monocytogenes, L.lactis subsp lactis and L.curvatus H28, but not against the Gram-positive bacteria L.curvatus CWBI-B28, L.brevis and L.plantarum or the Gram-negative bacteria E.coli and Pseudomonas sp 55. Has no antifungal activity against S.cerevisiae, Penicillium sp BKS- TAN2 or A.niger.'
<Medium Confidence>: '(R)-amygdalin + H2O = (R)-prunasin + D-glucose'
<Medium Confidence>: 'extracellular region | neuropeptide signaling pathway'
<Medium Confidence>: '2 a phenolic donor + H2O2 = 2 a phenolic radical donor + 2 H2O'
<Medium Confidence>: 'extracellular space | serine-type endopeptidase activity, toxin activity | proteolysis'

**Retrieved proteins by MMSeqs2:**

<High Confidence>: 'cytosol | carbohydrate derivative binding, glucose-6-phosphate isomerase activity, monosaccharide binding | gluconeogenesis, glucose 6-phosphate metabolic process, glycolytic process'

**Here are some example input-output pairs for this task:**

'example answer': \"The protein characterized by the amino acid sequence demonstrates kynurenines activity, pyridoxal phosphate binding and is implicated in the 'de novo' NAD biosynthetic process from tryptophan, anthranilate metabolic process, L-kynurenine catabolic process, quinolinate biosynthetic process, tryptophan catabolic process. Its subcellular localization is primarily within the cytoplasm.
'example answer': \"Upon analysis of the specified amino acid sequence, it is evident that the protein performs 3-hydroxykynureninase activity, kynurenines activity, pyridoxal phosphate binding, participating in the 'de novo' NAD biosynthetic process from tryptophan, anthranilate metabolic process, L-kynurenine catabolic process, quinolinate biosynthetic process, tryptophan catabolic process. This protein is generally localized to the cytoplasm within the cell.
Based on the instruction, the protein sequence, the retrieved information, and the examples, output ONLY the functional description of this protein in the following JSON format:
{\"description\": \"...\"}

**Do not output any other text or explanation. Only output the JSON answer.**

## Ground Truth:

"Based on the given amino acid sequence, the protein appears to have a primary function of metal ion binding. It is likely involved in the regulation of transcription by RNA polymerase II, and its subcellular localization is within the MOZ/MORF histone acetyltransferase complex, nucleus."
**"Bio-Entity": "MOZ/MORF histone acetyltransferase complex, nucleus | metal ion binding | regulation of transcription by RNA polymerase II"**

## Response:

The protein with the provided amino acid sequence is predicted to function as a histone acetyltransferase. It likely **binds histones and metal ions**, and acts as a histone reader. Biologically, it is involved in processes including regulation of **transcription by RNA polymerase II**, DNA-templated transcription, histone acetylation, erythrocyte maturation, regulation of hemopoiesis, and the regulation of developmental processes. The protein localizes predominantly to the nucleus, including nuclear specks, and **is a component of the MOZ/MORF histone acetyltransferase complex**.

Figure 7: Example of retrieval-augmented prompt for protein knowledge injection at inference. We highlight the matching part with the ground truth in green.

lar sequences and identical labels across the training and test sets demonstrates significant data contamination, allowing models to perform well by pattern-matching or retrieving based on superficial sequence similarity rather than developing genuine biological understanding. These cases underscore the severity of the leakage issue and motivate the need for our proposed benchmark splits.

### D.2 Case Study for RAG methods

Fig. 7 demonstrates our prompting structure, illustrating how we augment the protein sequence and query with explicit biological information retrieved from our dual-indexed database (including feature and sequence similarity results with confidence). Few-shot examples are also incorporated to guide the LLM's response format. The LLM synthesizes this retrieved evidence to generate a detailed answer about the protein. As shown, this augmented approach guides the LLM to accurately identify key biological entities and functional details compared to the ground truth, demonstrating how retrieving relevant biological context improves performance.

## E Details on Metrics

We evaluate the model using several commonly used evaluation metrics adapted to protein description generation and understanding tasks. Here, we detail these metrics, including their calculation method, significance, and specific usage.

**BLEU:** (Papineni et al., 2002) BLEU, or BiLingual Evaluation Understudy, is a metric often used to measure the fluency and correspondence of machine-generated sequences against reference descriptions. Employing $n$-grams, we compute the overlap:

$$\text{BLEU} = \text{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right),$$

where BP is a brevity penalty, $w_n$ are the weights typically equal for all $n$-grams, $\sum_{n=1}^{N} w_n = 1$, and $p_n$ is the precision for $n$-grams.

**ROUGE:** (Lin, 2004) Recall-Oriented Understudy for Gisting Evaluation (ROUGE) measures the quality of machine-generated text by comparing its overlap with a reference set of word sequences. Specifically, it evaluates:

- ROUGE-N (e.g., ROUGE-1, ROUGE-2): Measures $n$-gram overlap.
- ROUGE-L: Based on the longest common subsequence, it considers both recall and precision to compute an F1 score.

## F Discussion on Licensing

### F.1 Pretrained Models and Codes.

**Llama 3** (Dubey et al., 2024) The Llama 3 model is released under the Llama Community License. This license permits use, modification, and distribution, with specific conditions such as prohibitions against using the model for training other language models. For commercial use, compliance with Meta's Acceptable Use Policy is mandatory, and entities with over 700 million monthly active users must obtain a separate license from Meta.

**GPT-4.1** (OpenAI, 2025) The content and models are provided under OpenAI's Terms of Use and API License Agreement. Commercial use, redistribution, or modification of GPT-4.1 models via the API requires compliance with OpenAI's policies, including attribution and adherence to usage restrictions. For full details, review OpenAI's official terms.

**DeepSeek-V3** (Liu et al., 2024a) DeepSeek V3 is distributed under the DeepSeek License (v1.0, Oct 23, 2023). It grants a free, global, irrevocable license for modification and distribution, with strict restrictions on military use, harm, misinformation, discrimination, and unauthorized data processing. Users must enforce these limits in derivative works. Disclaimers of warranties and liability are included, and any legal matters are subject to the jurisdiction of Chinese law, specifically in Hangzhou.

**BioGPT** (Luo et al., 2022) BioGPT is released under the MIT License, permitting unrestricted use, modification, and distribution of the software and its pre-trained models, provided that the original copyright notice and license terms are included in all copies or substantial portions of the software. The software is provided "as is," without warranty of any kind, express or implied.

**BioT5+ Model** (Pei et al., 2024) The BioT5+ model is available under the MIT License. This allows for free use, modification, and distribution, including for commercial purposes, as long as the original copyright notice and permission notice are retained. The software is provided "as is," with no warranties or guarantees, and the authors disclaim liability for any issues arising from its use.

| Dataset | Protein Sequence | Functional/Domain Information |
|---|---|---|
| **Test Dataset** (UniProt A4WLK4) | MVRIAVVVSEFNYDVTQLMLQKALE HAKFLGAEVTYVVKVPGVYDIPTLL RDLVAKEEVDAVVTLGAVIQGATKH DEVVAHQAARKILDISVESGKPITL GIIGPGANRMQALERVEEYAKRAVE AAVKLARRKKTLREAKYAGSTVFID | 6,7-dimethyl-8-ribityllumazine synthase activity; involved in riboflavin biosynthetic process; subcellular localization: riboflavin synthase complex. |
| **Training Dataset** (UniProt A0A832T310) | MVRLAIVVAEFNYDITQLMLQKAVE HAKFLGAEITYIVKTPGVYDIPMIL KELVAKEEVDAVATLGAVIQGATKH DELVATQAARKILDIAVESGKPITL GIIGHGANRIQALERVEEYARRAVE AAVKMARRKKALREAKYNGSTVYID | 6,7-dimethyl-8-ribityllumazine synthase activity; involved in riboflavin biosynthetic process; subcellular localization: riboflavin synthase complex. |
| **Test Dataset** (UniProt Q27996) | MKALLILGLLLLSVAVQGKTFKRCE LAKTLKNLGLAGYKGVSLANWMCLA KGESNYNTQAKNYNPGSKSTDYGIF QINSKWWCNDGKTPKAVNGCGVSCS ALLKDDITQAVACAKKIVSQQGITA WVAWKNKCRNRDLTSYVKGCGV | Contains C-type lysozyme domains (based on computational analysis). |
| **Training Dataset** (UniProt P51782) | MKVLLLLGFIFCSMAAHGKRMERCE FARRIKQLHLDGYHQISLANWVCLA QWESGFDTKATNYNPGDQSTDYGIL QINSHYWCDDGKTPHAANECKVRCS ELQEDDLVKAVNCAKKIVDQQGIRA WVAWRNKCEGKDLSKYLEGCHL | Contains C-type lysozyme domains (based on computational analysis). |

Table 8: Comparison of Protein Sequences and Functional/Domain Annotations in Test and Training Datasets.

**Galactica** (Taylor et al., 2022) The model is licensed under a non-commercial research license. This license permits use of the model and its derivatives solely for non-commercial research and evaluation purposes. Commercial use, including but not limited to using the model or its derivatives in a product or service, is strictly prohibited. Redistribution of the model weights or modifications thereof is allowed only with appropriate attribution and under the same terms. The model is provided "as is," without warranty of any kind, express or implied, including but not limited to warranties of merchantability, fitness for a particular purpose, and noninfringement.

**ProLlama** (Lv et al., 2024) The model is released for research and educational purposes only. Redistribution and use in source and binary forms, with or without modification, are permitted for non-commercial use provided that the original authors are properly cited. Any commercial use or use of the model or its derivatives in a commercial product or service is strictly prohibited.

**ESM-2** (Lin et al., 2022) ESM Metagenomic Atlas (also referred to as "ESM Metagenomic Structure Atlas" or "ESM Atlas") data is available under a CC BY 4.0 license for academic and commercial use. Copyright (c) Meta Platforms, Inc. All Rights Reserved.

**MMSeqs2** (Steinegger and Söding, 2017) MMseqs2 is licensed under the MIT License, permitting free use, modification, and distribution of the software, provided that the original copyright notice and license terms are included in all copies or substantial portions of the software. The software is provided "as is," without warranty of any kind, express or implied.

### F.2 Datasets

**UniProt Database** (Consortium, 2019) The UniProt Database is available under the Creative Commons Attribution (CC BY 4.0) License. This license permits users to share and adapt the data for any purpose, provided appropriate credit is given, a link to the license is provided, and an indication of any changes made is specified.

**Mol-Instructions Dataset** (Fang et al., 2024) Released under the Creative Commons Attribution-NonCommercial 4.0 International License (CC BY-NC 4.0). This license permits use, sharing, and adaptation of the dataset for non-commercial purposes, with appropriate attribution and indication of changes. Commercial use requires additional permissions.