OPI: An Open Instruction Dataset for Adapting Large Language Models to Protein-Related Tasks

Hongwang Xiao[*†1,2], Wenjun Lin[*1], Hui Wang[*1], Zheng Liu[1], and Qiwei Ye[‡1]

[1]Beijing Academy of Artificial Intelligence, Beijing, China
[2]Peking University, Beijing, China
{hwxiao, wjlin, wanghui, qwye}@baai.ac.cn, zhengliu1026@gmail.com

Abstract

Large language models (LLMs) pretrained on extensive general corpora, such as GPT-4 and Llama series, have shown exceptional performance across a wide range of natural language processing (NLP) tasks. These models provide a user-friendly and efficient interface that aligns well with user preferences through natural language instructions. Despite these advances, the application of LLMs in biomolecular sciences, particularly in protein-related research, remains constrained, with the boundaries of their capabilities yet to be fully explored. To bridge this gap, we present **Open Protein Instructions (OPI)**, a comprehensive dataset containing over 1.64M instruction-tuning samples (98.38% training, 1.62% testing) dedicated to protein research. OPI enables LLMs to perform a broad array of protein-related tasks efficiently and cost-effectively. Experimental evaluations across three task categories—sequence understanding (SU), annotation prediction (AP), and knowledge mining (KM)—demonstrate OPI's effectiveness in adapting LLMs to protein-specific applications. Our findings support the feasibility of leveraging LLMs for biomolecular research through instruction tuning. Data, codes, and instruction-tuned models are publicly available at `https://github.com/baaihealth/opi` to advance research in this field.

1 Introduction

Large Language Models (LLMs) such as GPT-4[OpenAI, 2023] and the Llama series[Dubey et al., 2024] have shown exceptional performance across a wide range of natural language processing (NLP) tasks[Tamkin et al., 2021, Zhao et al., 2023]. These models can serve as general assistants to follow instructions[Wu et al., 2023], adept at addressing various tasks. Furthermore, this capability suggests tremendous potential for applying LLMs to tackle complex scientific challenges[Jablonka et al., 2023]. Preliminary research has indicated that LLMs, like GPT-4, possess extensive domain knowledge and strong predictive abilities in scientific fields like drug discovery and materials design[Sharma and Thakur, 2023, Blanco-Gonzalez et al., 2023, Pradhan et al., 2024]. Their proficiency in problem-solving and knowledge integration is impressive[Hu et al., 2023]. Recent years have witnessed great promise of LLMs in the field of biomedical domain, such as analysing vast amounts of literature and patents for drug discovery[Rane, 2023], or improving the design and implementation of clinical trials[Zhang et al., 2024].

[*]Equal contribution.
[†]Project lead.
[‡]Corresponding author: qwye@baai.ac.cn

Foundation Models for Science Workshop,38th Conference on Neural Information Processing Systems (NeurIPS 2024).

However, LLMs still struggle to provide highly accurate and reliable results[Meyer et al., 2023, Yang et al., 2024], particularly in complex scientific scenarios like biology and computational chemistry[AI4Science and Quantum, 2023]. These challenges mainly arise from the inherent complexity and scarcity of domain-specific scientific data, leading to limited ability of pre-trained LLMs in these specialized fields. Therefore, this underscores the significance of developing well-designed datasets and methodologies to incorporate specialized domain knowledge into LLMs and to enhance their adaptability and ability to diverse scientific challenges.

Some previous studies have demonstrated that instruction tuning can further enhance LLMs' performance in biomolecular tasks by integrating and understanding diverse data types, including biomolecular sequences, structures, and functional texts[Pei et al., 2024, Fang et al., 2024]. This allows LLMs for a comprehensive analysis of biomolecular mechanisms[Feng et al., 2024], such as enzyme catalysis and gene regulation, etc. By leveraging these capabilities, LLMs can aid in understanding complex biological processes, elucidate mechanisms underlying disease states. Despite these advancements, the capabilities of LLMs in handling diverse protein-related tasks remain limited. A major challenge is the lack of comprehensive instruction datasets specifically designed for tasks related to protein biology. Existing work fall short in capturing the full diversity and complexity of protein data, which hinders the effective application and optimization of LLMs in this domain. This gap underscores a significant limitation in the current LLM landscape, highlighting the importance for the development of specialized and exhaustive protein instruction datasets. Addressing this challenge is crucial for enabling LLMs to be fully adapted and effectively utilized in the study of protein biology.

In this study, we created an instruction dataset for protein biology, Open Protein Instructions (OPI), which is suitable for a range of protein-related tasks, including **sequence understanding(SU)**, **annotation prediction(AP)**, and **knowledge mining(KM)**. OPI is able to adapt open-source LLMs to these protein-related tasks via instruction tuning. Comprehensive experiments demonstrate that the OPI-tuned LLMs exhibit decent performance on diverse evaluation tasks. Main contributions of the work go as follows:

I. To the best of our knowledge, OPI is the largest protein instruction dataset to date with over 1.64M samples. This diverse dataset is designed to evaluate, adapt and enhance LLMs in a wide range of protein-related tasks.

II. Comprehensive experiments based on OPI, uncover valuable insights that OPI could effectively enhance LLMs' understanding of diverse protein-related data, enabling reasoning about protein domain knowledge and further solving protein-related tasks.

III. All materials pertinent to this study, including datasets, source codes, and fine-tuned models utilizing OPI, have been made publicly accessible.

We anticipate that the findings of this study will significantly contribute to advancing the field of computational biology driven by LLMs, fostering further innovation and collaboration within the scientific community.

2 Related work

Recent advancements in natural language processing (NLP) have been driven by the remarkable performance of large language models (LLMs) like GPT-4[OpenAI, 2023], Llama series[Dubey et al., 2024] and Galactica[Taylor et al., 2022] which excel across a wide range of tasks. The former two models are primarily trained on large, general corpora, such as CommonCrawl*; whereas Galactica is predominantly pre-trained on scientific literature, including academic papers and textbooks, etc. Despite their advancements, pre-trained LLMs still encounter challenges when handling diverse tasks across various domains if they do not undergo domain adaptation. Consequently, it is necessary to further enhance the capabilities of pre-trained LLMs for downstream tasks. There are three primary strategies for this: fine-tuning[Howard and Ruder, 2018], prompt tuning[Brown et al., 2020, Li and Liang, 2021, Lester et al., 2021] and instruction tuning[Wei et al., 2022]. Among these, instruction tuning has increasingly become a standard practice, recognized for its remarkable effectiveness. This method refines language models through the use of datasets annotated with natural language instructions, thereby substantially enhancing the models' generalization across a wide range of

*https://commoncrawl.org/

tasks. To facilitate these advancements, several large-scale instruction tuning datasets have been developed. Notable examples include Alpaca[Taori et al., 2023] employing self-instruct techniques and COIG[Zhang et al., 2023] which focuses on instruction tuning in Chinese.

Recent studies have integrated LLMs into molecular biology to address critical challenges such as drug discovery. And for further enhancement of LLM performance in molecular biology, particularly in understanding biomolecular data, several instruction-tuning datasets have been developedFang et al. [2024], Cao et al. [2023], Li et al. [2024], Shi et al. [2023], Wang et al. [2023], Jin et al. [2024]. However, most existing efforts focus on small molecules, covering tasks like molecular property prediction[Li et al., 2024, Cao et al., 2023] and chemical reaction prediction[Shi et al., 2023]. Only a few studies, such as Mol-Instructions[Fang et al., 2024], partially involve protein-related data with only around 0.5M protein instructions, which is insufficient in scale to address the complexity of protein-related scenarios.

3 Dataset construction and analysis

3.1 Construction process of OPI

The process of constructing the OPI dataset is depicted in Fig. 1. The first step involves extracting data from various protein databases, which are subsequently divided into training and testing sets based on a predefined ratio. For each evaluation task, we initially craft a set of task-specific instructions through manual efforts. These initial instructions are then expanded into a larger set using the GPT-3.5 model to generate additional task-specific examples (Step 1). Following this, each sample in the training and testing sets is formatted into an instruction-based style, comprising an instruction, input, and output (Step 2). The instruction for each sample is randomly selected from the pool generated in Step 1. Ultimately, a total of 1.64M samples are constructed to form the complete OPI dataset, as summarized in Fig. 2(a). Examples of training and testing data can be found in Appendix A.1.

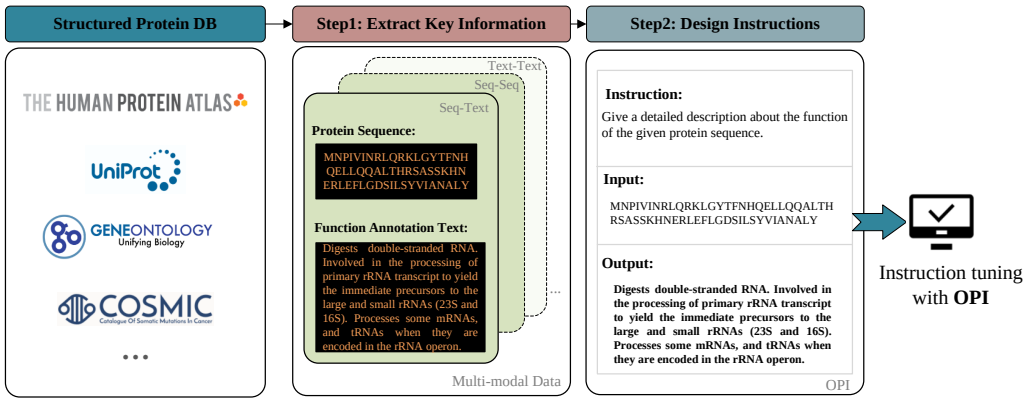
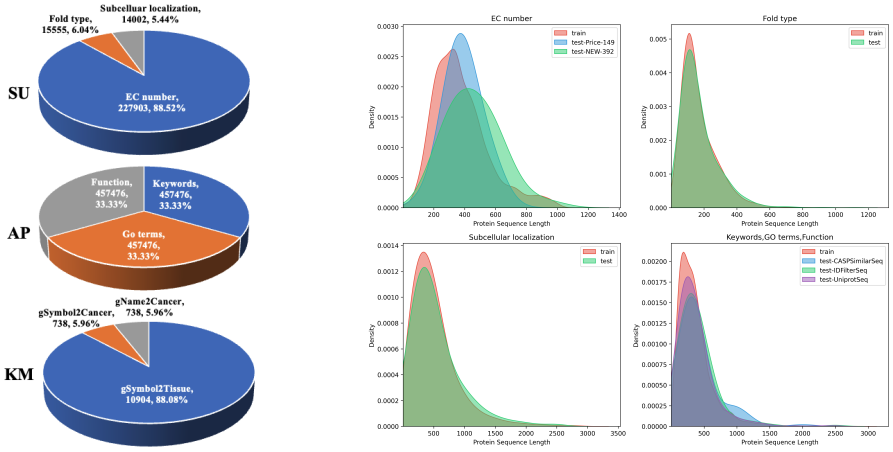


Figure 1: **Construction of the OPI dataset involves several key steps.** First, essential protein information from databases such as UniProt was extracted. This information is then transformed into an "Instruction-Input-Output" format to create the OPI dataset. The dataset is subsequently used to fine-tune LLMs like Galactica and Llama.

3.2 Distribution analysis of OPI

Fig. 2(b) highlights distinct differences in protein sequence length distributions across tasks. These variations in sequence length distributions suggest that OPI holds a good quality of sequence length range. This characteristic may affect model performance, as models trained on shorter sequences may not generalize well to longer ones. Ensuring that the model is robust across the full spectrum of protein lengths is critical for achieving reliable performance during evaluation. Further analysis of the label count distribution and function description length variation can be found in Fig. 5 in Appendix A.6.



(a) (b)

Figure 2: **a. Distribution of the full OPI dataset**, comprising over 1.64M samples (training: 1,615,661, testing: 26,607). **b. Comparison of protein sequence length distributions across nine tasks**, showing variations between training and testing sets.

4 Experimental design

This study is critical for advancing our understanding of how LLMs can be leveraged in protein biology. By systematically evaluating LLMs across three key categories of tasks—sequence understanding, annotation prediction, and knowledge mining—we aim to uncover their strengths and limitations in protein modeling. The two core research questions (**Q1** and **Q2**) we investigate are central to this endeavor: the capacity of base LLMs to generalize to protein-related tasks and the efficacy of instruction tuning for them. On the one hand, it is vital to determine whether base LLMs can successfully generalize to critical protein-related tasks. Addressing those tasks precisely can substantially advance our understanding of biological processes and disease mechanisms. On the other hand, assessing the effectiveness of instruction tuning offers insightful guidance to how specialized training can improve model performance. These findings hold significant implications for optimizing LLMs to address specific challenges in protein biology and related fields. Overall, this study not only bridges a critical gap between LLMs and protein biology, but also contributes to the development of user-friendly and efficient computational tools for biologists. By providing a comprehensive evaluation of LLMs' capabilities and the effect of instruction tuning with OPI, this study offers a pathway toward more precise and impactful applications of LLMs in protein biology and beyond. The two questions are as follows:

Q1: *Could base LLMs be effectively generalized to protein-related tasks, such as predicting enzyme commission (EC) numbers, gene ontology (GO) terms, and cancer types based on gene names?*

Q2: *Could instruction tuning enhance LLMs' performance in protein-related tasks? Additionally, how effectively do different base LLMs respond to instruction tuning?*

4.1 Methods

LLMs have greatly advanced research, especially in the open-source community. The Llama series, including the recently released Llama 3.1 models [Dubey et al., 2024], is noted for strong performance across various tasks and is often fine-tuned through instruction tuning to adapt to new scenarios. Galactica, the model trained for scientific domains, uses about 83% scientific data, including research papers, databases, and biomolecular sequences. Unlike Galactica, Llama 3.1's pre-training corpus is different, with about half of it consisting of general knowledge (see Fig. 4 in Appendix A.2 for the pre-training data summary). This study, therefore, concentrates on a comprehensive evaluation of Galactica and Llama-3.1 models for protein modeling tasks, representing the non-instruct version in the scientific domain and the instruct version in the natural language domain, respectively, to assess their effectiveness and performance in this specialized area. This comprehensive and comparative analysis aims to highlight the strengths and limitations of each model in the context of protein-related

tasks, providing a thorough assessment of their relative performance and suitability. The experimental design is outlined in Fig. 3. Candidate models for evaluation, including baseline models and OPI-tuned models, are listed in Table 1. During the inference phase, only a single output is generated for each model. Training and inference hyperparameters are detailed in Appendix A.3, and the corresponding evaluation metrics for each task are introduced in Section 4.2.

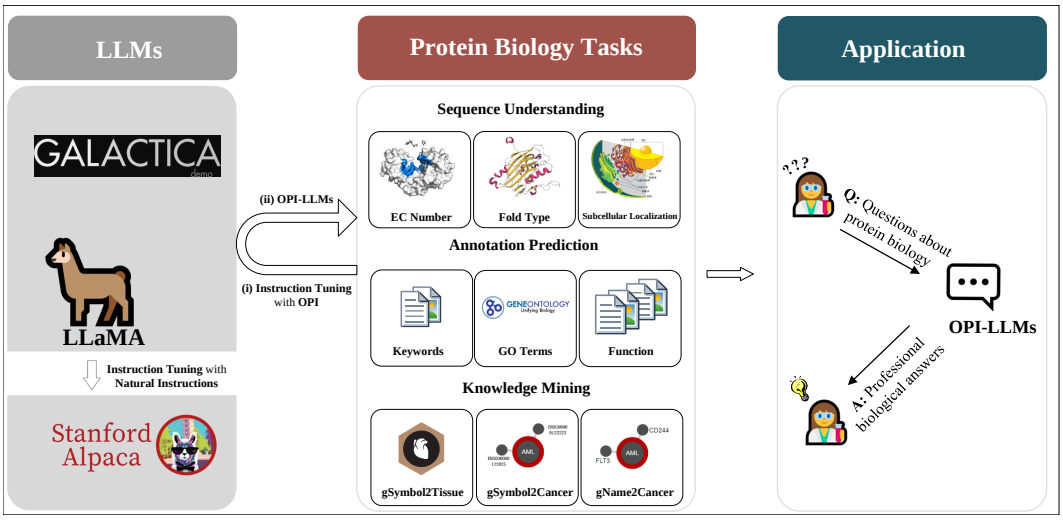


Figure 3: **Experimental design outline.** We begin by constructing the OPI dataset, which encompasses nine protein-related tasks, categorized into SU, AP and KM. Next, we use the training sets of OPI to perform instruction tuning on the original models, resulting in OPI-tuned models. All models are evaluated using the same testing sets. Following a thorough evaluation, these OPI-tuned models are anticipated to be effective for protein question-answering tasks in a conversational format.

Table 1: **Candidate baseline and OPI-tuned models.** The table lists various models categorized by their training approach. Baseline models include Galactica-6.7B and Llama-3.1-8B-Instruct. OPI-tuned models include OPI-Galactica-6.7B and OPI-Llama-3.1-8B-Instruct, which have been further finetuned on OPI training data.

Category	Model	Model Description
Baseline Models	Galactica-6.7B	Original model without instruction tuning
	Llama-3.1-8B-Instruct	Original instruction-tuned version of Llama-3.1-8B
OPI-tuned Models (ours)	OPI-Galactica-6.7B	Instruction-tuned Galactica-6.7B on OPI training data
	OPI-Llama-3.1-8B-Instruct	Continual instruction-tuned Llama-3.1-8B-Instruct on OPI training data

4.2 Evaluation task definition

This study aims to thoroughly explore and evaluate the capabilities of LLMs in protein modeling by evaluating them on three categories of protein-related tasks: **sequence understanding(SU)**, **annotation prediction(AP)**, and **knowledge mining(KM)**. Details of the evaluation tasks are listed in Table 2. These tasks are designed to systematically assess and reveal the potential of LLMs in the field of protein. Specifically, we have developed nine evaluation tasks, with three tasks for each category. Our evaluation addresses two critical questions (**Q1** and **Q2** introduced at the beginning of Section 4) regarding the application of LLMs to protein modeling. For insights into the significance of designing each task, see Appendix A.4.

4.2.1 Sequence understanding

[Rationale for task design] The **SU** evaluation aims to predict protein properties related to structure or function based on protein sequences. Unlike traditional regression or classification approaches, LLMs can generate textual results directly from relevant instructions. The goal of SU evaluation is

Table 2: **Evaluation tasks, training/testing sets and metrics.** The evaluation tasks are categorized into three types: SU, AP and KM. For each task, the specific training set size, testing set size, and corresponding evaluation metric used is indicated. Metrics include *precision*, *recall*, *F1* (see Appendix A.5 for the calculation formulas), *accuracy*, and *ROUGE-L*, as appropriate for the corresponding tasks. The input and label for each task is exampled in Appendix A.6.

Task category	Task name	Training set size	Testing set size	Metrics
SU 4.2.1	EC number prediction	227,362	392(N), 149(P)	*Precision*, *Recall* and *F1*
	Fold type prediction	12,311	718(Fo), 1254(S), 1272(Fa)	*Accuracy*
	Subcellular localization prediction	11,230	2,772	*Accuracy*
AP 4.2.2	Keywords prediction	451,618	184(C), 1,112(I), 4,562(U)	*Precision*, *Recall* and *F1*
	GO terms prediction	451,618	184(C), 1,112(I), 4,562(U)	*Precision*, *Recall* and *F1*
	Function prediction	451,618	184(C), 1,112(I), 4,562(U)	*ROUGE-L*[Lin, 2004]
KM 4.2.3	Tissue location prediction from gene symbol	8,723	2,181	*Precision*, *Recall* and *F1*
	Cancer prediction from gene symbol	590	148	*Precision*, *Recall* and *F1*
	Cancer prediction from gene name	590	148	*Precision*, *Recall* and *F1*

** N: NEW-392, P: Price-149, Fo: Fold, S: Superfamily, Fa: Family, C: CASPSimilarSeq, I: IDFilterSeq, U: UniprotSeq.*

to assess the sequence modeling capabilities of LLMs. This evaluation comprises three tasks: **EC number prediction**, **fold type prediction**, and **subcellular localization prediction**.

I. EC number prediction

Enzymes are crucial proteins involved in various biological processes. The Enzyme Commission (EC) number system classifies enzymes into thousands of categories based on their catalytic functions, using a four-digit format (e.g., 3.4.11.4). The training and testing datasets for this task are sourced from CLEAN's dataset [Yu et al., 2023], which is derived from the Swiss-Prot database [Boutet et al., 2007].

II. Fold type prediction

Protein folding refers to the classification of protein tertiary structures. Proteins with similar fold types may have significant variations in sequence identity. Typically, fold type prediction relies on models that map sequences to structures, assigning fold types from a set of 1,195 categories. The training and testing sets are taken from Hou's dataset [Hou et al., 2018], which is based on the SCOPE 1.75 database [Fox et al., 2014]. This study includes three testing set types: fold level, superfamily level, and family level (see Appendix A.7 for detailed explanations). For instance, the fold level testing set is constructed by excluding entire superfamilies from the original dataset, making this task also relevant for remote homology detection [Rao et al., 2019], which evaluates the model's ability to learn evolutionary relationships from distant homologous sequences.

III. Subcellular localization prediction

This task involves predicting the location of natural proteins within eukaryotic cells, with categories such as nucleus, cytoplasm, and membrane. The training and testing datasets are obtained from DeepLoc [Almagro Armenteros et al., 2017], which is extracted from the UniProt database. DeepLoc clusters sequences based on 30% sequence identity and divides the dataset into five parts with balanced numbers of sequences. Four parts are used for training and one part for testing.

4.2.2 Annotation prediction

[Rationale for Task design] Protein sequence and biological text are two critical information sources in protein biology. While significant research has focused on modeling these sources separately, there has been less emphasis on their joint modeling. The main challenge in integrating diverse protein information lies in establishing a unified multi-modal data representation, as different protein databases store data in varied formats, such as knowledge graphs [Zhang et al., 2022a] and tabular structures [Uhlen et al., 2010]. To address this challenge, we developed the AP evaluation to assess the ability of LLMs to jointly model protein sequence and text. The **AP** evaluation comprises three tasks: **keywords prediction**, **GO terms prediction**, and **function prediction**. Datasets for these tasks are created using a consistent pipeline. In order to evaluate the ability of LLMs comprehensively, we constructed three hold-out testing sets, similiar to those in [Taylor et al., 2022], with varying

criteria: CASPSimilarSeq, IDFilterSeq, and UniProtSeq (see Appendix A.8 for details on dataset construction).

I. Keywords prediction

Function keywords are used to summarize protein annotations in UniProt, categorized into ten types (e.g., biological process and molecular function) [†]. These keywords encapsulate protein properties through a controlled vocabulary, facilitating efficient retrieval of relevant protein sequences. This task involves multi-label prediction.

II. GO terms prediction

The Gene Ontology (GO) project provides a structured vocabulary to describe biological knowledge across three domains: biological processes, cellular components, and molecular functions (see Appendix A.9 for a comparison of GO terms and function keywords). We use GO terms from the Swiss-Prot database, where each term is manually mapped from the Gene Ontology knowledge base [‡].

III. Function prediction

This task evaluates the model's ability to generate relevant biological descriptions based on protein sequences. The dataset is constructed by pairing protein sequences with their descriptions, both sourced from the Swiss-Prot database. These descriptions detail the properties and functions of proteins in human-readable text [§].

4.2.3 Knowledge mining

[Rationale for task design] Recent advancements in NLP have demonstrated the impressive capabilities of LLMs in tasks such as reading comprehension, natural language inference, and question-answering. These successes can be attributed to LLMs' ability to store, integrate, and reason about vast amounts of knowledge [Taylor et al., 2022]. Unlike traditional search engines or databases, LLMs offer a novel interface for accessing and interpreting information. This capability motivates us to explore the application of LLMs to protein biology, aiming to extract valuable insights from the extensive protein databases and scientific literature accumulated over years of bioinformatics research. To investigate this potential, we designed the KM evaluation, which focuses on assessing LLMs' ability to uncover useful knowledge from large volumes of protein information. The **KM** evaluation consists of three tasks: tissue location prediction from gene symbols(**gSymbol2Tissue**), cancer prediction from gene symbols(**gSymbol2Cancer**), and cancer prediction from gene names(**gName2Tissue**).

I. gSymbol2Tissue

This task involves predicting the expression locations of protein-coding genes (PCGs) in various tissues and organs based on gene symbols. The dataset is sourced from the Human Protein Atlas (HPA) [¶], a comprehensive human proteomics database. It includes 10,904 PCGs across 45 tissues, with each gene associated with one or more expression locations. The dataset is divided into training and testing sets in an 80:20 ratio.

II. gSymbol2Cancer

This task evaluates the model's ability to identify cancer types associated with given gene symbols [‖]. The dataset is derived from the Cancer Gene Census, which classifies genes based on their roles in different cancers. It includes 738 records covering 420 cancer types, with 80% used for training and 20% for testing. The input is a gene symbol (e.g., ATRX), and the target is the associated cancer names (e.g., lung cancer, T-ALL).

III. gName2Tissue

[†] `https://www.uniprot.org/help/keywords`

[‡] `http://geneontology.org/`

[§] `https://www.uniprot.org/help/general_annotation`

[¶] `https://www.proteinatlas.org/humanproteome/tissue`

[‖] The terms *Gene Symbol* and *Gene Name* are adopted from the Human Gene Database (`https://www.genecards.org/`). For example, CPEB3 is the gene symbol and its gene name is Cytoplasmic Polyadenylation Element Binding Protein 3.

Similar to the previous task, this task assesses the model's ability to predict cancer types based on gene names rather than symbols. The dataset is the same as in the previous task, but the input consists of full gene names (e.g., alpha thalassemia/mental retardation syndrome X-linked) instead of symbols. The goal is to identify cancer names associated with these gene names.

5 Experimental results

This section provides comprehensive results of the experiments we have conducted. By analysing the results, we conclude the following answers to **Q1** and **Q2** in Section 4:

A1: *Overall, baseline Llama-3.1-8B-Instruct and Galactica-6.7B models demonstrate limited effectiveness in protein-related tasks, particularly in EC number prediction task (see Table 3), GO terms prediction task (see Table 5) and three knowledge mining tasks (see Table 6). Additionally, Galactica-6.7B exhibits slightly better performance compared to Llama-3.1-8B-Instruct in tasks such as function prediction and keywords prediction (see Table 5). This enhancement is probably attributed to Galactica's pre-training on relevant corpus, which has significantly improved its capabilities in these specific tasks. For more comparison of experimental results, please refer to Appendix A.10.*

A2: *Instruction tuning with OPI could notably enhance the performance of base LLMs in protein-related tasks. Furthermore, different OPI-tuned LLMs exhibit varying performance, with OPI-Galactica-6.7B outperforming OPI-full-1.61M-LaMA-3.1-8B-Instruct. Specifically, fine-tuning with OPI equips baseline LLMs with competitive capabilities on the task of EC_number_NEW-392 (see Table 3). For broader tasks like generating function descriptions or keywords based on sequences, OPI significantly boosts baseline LLMs' performance, enhancing the accuracy and depth of function annotations, which is expected in turn to support protein biologists in their research and discovery efforts (see Table 5). Additionally, OPI-tuned models, particularly OPI-Galactica-6.7B, have excelled in all three knowledge mining tasks (see Table 6). This highlights the significant potential of specialized LLMs to assist biologists as effective research tools in advancing protein studies. Appendix A.12 showcases the accuracy and reliability of several state-of-the-art LLMs, such as GPT-4o and Claude 3.5 Sonnet, alongside the OPI-tuned model, highlighting the effectiveness of OPI in adapting LLMs for protein-related tasks.*

5.1 Evaluation results of sequence understanding tasks

Our experiments in SU tasks demonstrate that models tuned with OPI are effective at modeling protein sequences. Based on the experimental results of EC number prediction shown in Table 3, we found that the original Galactica-6.7B, and Llama-3.1-8B-Instruct models, which were not fine-tuned using OPI, performed poorly on the EC number prediction task. This indicates that using the original models directly for specific protein-related tasks does not yield effective prediction results. In contrast, models fine-tuned with OPI showed some improvement on both testing datasets (Price-149 and NEW-392), but the overall scores fall behind existing models, such as CLEAN[Yu et al., 2023] and ProteInfer[Sanderson et al., 2023]. This suggests that although instruction-tuning with OPI can partially enhance the predictive ability of the models, further fine-tuning and optimization are required to achieve better performance on this task. These results underscore the importance of task-specific fine-tuning in improving model performance when applying LLMs in the life sciences domain.

To further investigate the capabilities and limitations of LLMs on protein-related tasks, we conducted experiments on folding type prediction, a more complex structure-related SU task, and subcellular localization prediction. As shown in Table 4, we design three different levels for fold type prediction according to the sequence identity between the testing and training sets, which progressively decreases from the Family level to the Superfamily level, and further to the Fold level. The performance of OPI-tuned models degrades significantly on "out of distribution" sequences, particularly on the Fold and Superfamily-level testing sets. Similarly, while the model shows some promise in subcellular localization prediction, it still falls short compared to the current state-of-the-art. This suggests that while instruction-tuned LLMs exhibit strong potential in protein-related tasks, they still encounter challenges in more advanced predictive tasks, especially those involving complex structural variations.

5.2 Evaluation results of annotation prediction tasks

As shown in Table 5, in the tasks of keywords prediction, GO term prediction, and function prediction, notable performance variations were observed across different testing sets: CASPSimilarSeq

Table 3: **Comparative evaluation results of EC number prediction task on testing sets Proce-149 and NEW-392.** The dataset size for fine-tuning indicates the number of sequence-label pairs, whose label is an EC number like 3.4.11.4. The original Galactica-6.7B and Llama-3.1-8B-Instruct models, which were not fine-tuned by OPI, did not demonstrate predictive capability on the EC number prediction task. This suggests that these models need further specific fine-tuning to improve their performance on this task.

Model	EC_number samples for fine-tuning	Model Type	Price-149			NEW-392		
			Precision	Recall	F1	Precision	Recall	F1
CLEAN[Yu et al., 2023]	227,362	pLM	0.5844	0.4671	0.4947	0.5965	0.4811	0.4988
ProteInfer[Sanderson et al., 2023]	348,105	CNN	0.2434	0.1382	0.1662	0.4088	0.2843	0.3086
DeepEC[Ryu et al., 2019]	1,388,606	CNN	0.1184	0.0724	0.0846	0.2976	0.2167	0.2297
DEEPre[Li et al., 2017]	22,168	CNN, LSTM	0.0415	0.0403	0.0386	-	-	-
Galactica-6.7B	-	LLM	-	-	-	-	-	-
Llama-3.1-8B-Instruct	-	LLM	-	-	-	-	-	-
OPI-Galactica-6.7B	227,362	LLM	0.0268	0.0268	0.0268	0.2700	0.2663	0.2596
OPI-Llama-3.1-8B-Instruct	227,362	LLM	0.0738	0.0738	0.0738	0.3724	0.3374	0.3468

* *pLM - Protein Language Model which is pre-trained with a large scale of protein sequence data.*

Table 4: **Comparative evaluation results (*Accuracy*) of fold type prediction and subcellular localization prediction tasks.** For information on the model architecture and the training procedure of the vanilla Transformer and ESM-1b [Rives et al., 2021] fine-tuning details, please refer to Appendix A.11. The results indicate that while instruction-tuning with OPI improves performance in some instances, there is still a noticeable gap between the OPI-tuned models and the current state-of-the-art.

Task name	Testing set	w/o pretrain	w/ pretrain	Literature SOTA	OPI-Galactica-6.7B	OPI-Llama-3.1-8B-Instruct
		Transformer	ESM-1b			
Fold type prediction	Family level	0.55	0.94	0.92[Rao et al., 2019]	0.49	**0.61**
	Superfamily level	0.11	0.49	0.43[Rao et al., 2019]	0.13	**0.15**
	Fold level	0.09	0.28	0.26[Rao et al., 2019]	0.08	**0.10**
Subcellular localization prediction	Hold-out	0.56	0.78	0.86[Xu et al., 2022]	**0.78**	0.42

(CSeq), IDFilterSeq (ISeq), and UniProtSeq (USeq). The Galactica-6.7B model demonstrated limited effectiveness on the three tasks with the CSeq, ISeq, and USeq testing sets, particularly in the GO term prediction task, where precision and F1 scores were not provided. Conversely, the Llama-3.1-8B-Instruct model did not provide effective prediction for all the tree tasks. This limitation may suggest potential issues related to data collection or model training for these tasks. Most notably, the instruction-tuned models exhibited exceptional performance across all testing sets. The OPI-Galactica-6.7B model achieved high precision, recall and F1 in function keywords prediction and GO terms prediction tasks. Furthermore, in the function prediction task, the model excelled, with Rouge-L scores consistently exceeding 0.7000 on all testing sets, highlighting its significant advantage in long text generation tasks.

Table 5: **Comparative evaluation of annotation prediction tasks.** It presents a comprehensive evaluation of the performance of various LLMs across three tasks on three distinct testing sets.

Model	Testing data	Keywords			GO terms			Function
		Precision	Recall	F1	Precision	Recall	F1	Rouge-L
Galactica-6.7B	CSeq	0.1050	0.1640	0.1160	-	-	-	0.1490
	ISeq	0.1270	0.2380	0.1570	-	-	-	0.1350
	USeq	0.1250	0.2220	0.1500	-	-	-	0.1390
Llama-3.1-8B-Instruct	CSeq	-	-	-	-	-	-	0.0555
	ISeq	-	-	-	-	-	-	0.0561
	USeq	-	-	-	-	-	-	0.0610
OPI-Galactica-6.7B	CSeq	**0.8120**	**0.7360**	**0.7643**	**0.7613**	**0.7492**	**0.7476**	0.7430
	ISeq	**0.8377**	**0.8019**	**0.8070**	**0.7404**	**0.7274**	**0.7207**	**0.7014**
	USeq	**0.8596**	**0.8196**	**0.8276**	**0.7638**	**0.7373**	**0.7358**	**0.7133**
OPI-Llama-3.1-8B-Instruct	CSeq	0.4202	0.5057	0.4385	0.1113	0.0936	0.0990	**0.7524**
	ISeq	0.6762	0.6905	0.6650	0.6686	0.6287	0.6304	0.4786
	USeq	0.7606	0.7489	0.7374	0.7150	0.6897	0.6849	0.5144

5.3 Evaluation results of knowledge mining tasks

As illustrated in Table 6, when assessing the tasks of gSymbol2Tissue, gSymbol2Cancer and gName2Cancer, the original Galactica-6.7B model has no predictive ability for all testing sets, which may imply that the model has certain limitations in these tasks. The Llama-3.1-8B-Instruct model also fails providing predictive results for all testing sets, which may indicate that the model has insufficiencies in data collection or model training for these tasks. In contrast, the OPI-tuned models show significant performance improvements across all testing sets. Especially in the gSymbol2Tissue testing set, both models achieve a recall of 0.9077 and 0.9356 respectively, indicating a high level of recall capability in the task of tissue location prediction. However, in the gSymbol2Cancer and gName2Cancer testing sets, the precision, Recall and F1 scores remain at a low level, indicating that the models have limitations in cancer name prediction tasks.

Overall, the OPI-tuned models demonstrate good comprehensive performance in KM tasks, especially achieving recall value higher than 0.9000 in tissue location prediction. These results suggest that by instruction-tuning with OPI, LLMs can be well adapted to protein-related KM tasks. Future research can further explore how to use these models to enhance the efficiency and accuracy of biomedical information mining.

Table 6: **Comparative evaluation results of knowledge mining tasks, including gSymbol2Tissue, gSymbol2Cancer and gName2Cancer.**

Model	Evaluation task	Precision	Recall	F1
Galactica-6.7B	gSymbol2Tissue	-	-	-
	gSymbol2Cancer	-	-	-
	gName2Cancer	-	-	-
Llama-3.1-8B-Instruct	gSymbol2Tissue	-	-	-
	gSymbol2Cancer	-	-	-
	gName2Cancer	-	-	-
OPI-Galactica-6.7B	gSymbol2Tissue	0.3917	**0.9077**	0.5303
	gSymbol2Cancer	0.3555	0.3189	0.3229
	gName2Cancer	0.2728	0.2554	0.2533
OPI-Llama-3.1-8B-Instruct	gSymbol2Tissue	0.4002	**0.9356**	0.5466
	gSymbol2Cancer	0.2890	0.2701	0.2664
	gName2Cancer	0.2786	0.2707	0.2659

6 Conclusion and future work

We conducted a comprehensive evaluation of LLMs in the context of protein modeling, encompassing nine critical tasks, categorized into sequence understanding, annotation prediction, and knowledge mining. This study developed the OPI dataset with over 1.64M samples — the largest high-quality protein instruction dataset available to date to the best of our knowledge, and applied instruction tuning to the original Galactica-6.7B and Llama-3.1-8B-Instruct models. Systematic assessments revealed that the OPI-tuned models perform well in nine protein tasks. These findings not only validate the exceptional quality of the OPI dataset but also highlight its critical role in advancing research in instruction-based protein modeling using LLMs, which holds considerable significance for the fields of protein biology. Moreover, this study points to some key areas for future exploration. Further research should focus on expanding the scope of OPI to encompass more protein-related tasks including protein engineering and protein design, and exploring multi-modal modeling that integrates knowledge across multiple protein modalities (e.g, protein tertiary structure) and biomolecular domains (e.g., DNA, RNA, small molecule). By continuing to develop and optimize such methodologies, it is expected to offer a unified LLM-based tool for protein biology, enabling highly accurate function prediction, de novo protein design, and ultimately, transformative advancements in biomedical applications.

Acknowledgments and Disclosure of Funding

This work was supported by China Postdoctoral Science Foundation (No. 2022TQ0049).

References

OpenAI. GPT-4 technical report, 2023.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Alex Tamkin, Miles Brundage, Jack Clark, and Deep Ganguli. Understanding the capabilities, limitations, and societal impact of large language models. *arXiv preprint arXiv:2102.02503*, 2021.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. A survey of large language models, 2023.

Xuansheng Wu, Wenlin Yao, Jianshu Chen, Xiaoman Pan, Xiaoyang Wang, Ninghao Liu, and Dong Yu. From language modeling to instruction following: Understanding the behavior shift in llms after instruction tuning. *arXiv preprint arXiv:2310.00492*, 2023.

Kevin Maik Jablonka, Qianxiang Ai, Alexander Al-Feghali, Shruti Badhwar, Joshua D Bocarsly, Andres M Bran, Stefan Bringuier, L Catherine Brinson, Kamal Choudhary, Defne Circi, et al. 14 examples of how LLMs can transform materials science and chemistry: a reflection on a large language model hackathon. *Digital Discovery*, 2(5):1233–1250, 2023.

Gaurav Sharma and Abhishek Thakur. ChatGPT in drug discovery. *ChemRxiv. 2023*, 2023. doi: 10.26434/chemrxiv-2023-qgs3k.

Alexandre Blanco-Gonzalez, Alfonso Cabezon, Alejandro Seco-Gonzalez, Daniel Conde-Torres, Paula Antelo-Riveiro, Angel Pineiro, and Rebeca Garcia-Fandino. The role of AI in drug discovery: challenges, opportunities, and strategies. *Pharmaceuticals*, 16(6):891, 2023.

Tathagata Pradhan, Ojasvi Gupta, and Gita Chawla. The future of ChatGPT in medicinal chemistry: Harnessing AI for accelerated drug discovery. *ChemistrySelect*, 9(13):e202304359, 2024.

Xin Hu, Yu Tian, Keisuke Nagato, Masayuki Nakao, and Ang Liu. Opportunities and challenges of ChatGPT for design knowledge management. *Procedia CIRP*, 119:21–28, 2023.

Nitin Rane. Contribution and challenges of ChatGPT and similar generative artificial intelligence in biochemistry, genetics and molecular biology. *Genetics and Molecular Biology (October 16, 2023)*, 2023.

Qiang Zhang, Keyang Ding, Tianwen Lyv, Xinda Wang, Qingyu Yin, Yiwen Zhang, Jing Yu, Yuhao Wang, Xiaotong Li, Zhuoyi Xiang, et al. Scientific large language models: A survey on biological & chemical domains. *arXiv preprint arXiv:2401.14656*, 2024.

Jesse G Meyer, Ryan J Urbanowicz, Patrick CN Martin, Karen O'Connor, Ruowang Li, Pei-Chen Peng, Tiffani J Bright, Nicholas Tatonetti, Kyoung Jae Won, Graciela Gonzalez-Hernandez, et al. ChatGPT and large language models in academia: opportunities and challenges. *BioData Mining*, 16(1):20, 2023.

Jingfeng Yang, Hongye Jin, Ruixiang Tang, Xiaotian Han, Qizhang Feng, Haoming Jiang, Shaochen Zhong, Bing Yin, and Xia Hu. Harnessing the power of LLMs in practice: A survey on ChatGPT and beyond. *ACM Transactions on Knowledge Discovery from Data*, 18(6):1–32, 2024.

Microsoft Research AI4Science and Microsoft Azure Quantum. The impact of large language models on scientific discovery: a preliminary study using gpt-4. *arXiv preprint arXiv:2311.07361*, 2023.

Qizhi Pei, Lijun Wu, Kaiyuan Gao, Jinhua Zhu, Yue Wang, Zun Wang, Tao Qin, and Rui Yan. Leveraging biomolecule and natural language through multi-modal learning: A survey. *arXiv preprint arXiv:2403.01528*, 2024.

Yin Fang, Xiaozhuan Liang, Ningyu Zhang, Kangwei Liu, Rui Huang, Zhuo Chen, Xiaohui Fan, and Huajun Chen. Mol-Instructions: A large-scale biomolecular instruction dataset for large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=Tlsdsb6l9n.

Ruijun Feng, Chi Zhang, and Yang Zhang. Large language models for biomolecular analysis: From methods to applications. *TrAC Trends in Analytical Chemistry*, page 117540, 2024.

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

Jeremy Howard and Sebastian Ruder. Universal language model fine-tuning for text classification, 2018.

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, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020.

Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4582–4597, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.353. URL https://aclanthology.org/2021.acl-long.353.

Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning, 2021.

Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. Finetuned language models are zero-shot learners, 2022.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Stanford alpaca: An instruction-following llama model, 2023. URL https://github.com/tatsu-lab/stanford_alpaca.

Ge Zhang, Yemin Shi, Ruibo Liu, Ruibin Yuan, Yizhi Li, Siwei Dong, Yu Shu, Zhaoqun Li, Zekun Wang, Chenghua Lin, Wenhao Huang, and Jie Fu. Chinese open instruction generalist: A preliminary release, 2023. URL https://arxiv.org/abs/2304.07987.

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

Sihang Li, Zhiyuan Liu, Yanchen Luo, Xiang Wang, Xiangnan He, Kenji Kawaguchi, Tat-Seng Chua, and Qi Tian. Towards 3D molecule-text interpretation in language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=xI4yNlkaqh.

Yaorui Shi, An Zhang, Enzhi Zhang, Zhiyuan Liu, and Xiang Wang. ReLM: Leveraging language models for enhanced chemical reaction prediction. *arXiv preprint arXiv:2310.13590*, 2023.

Zeyuan Wang, Qiang Zhang, Keyan Ding, Ming Qin, Xiang Zhuang, Xiaotong Li, and Huajun Chen. InstructProtein: Aligning human and protein language via knowledge instruction. *arXiv preprint arXiv:2310.03269*, 2023.

Mingyu Jin, Haochen Xue, Zhenting Wang, Boming Kang, Ruosong Ye, Kaixiong Zhou, Mengnan Du, and Yongfeng Zhang. ProLLM: Protein chain-of-thoughts enhanced LLM for protein-protein interaction prediction. *bioRxiv*, pages 2024–04, 2024.

Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81, 2004.

Tianhao Yu, Haiyang Cui, Jianan Canal Li, Yunan Luo, Guangde Jiang, and Huimin Zhao. Enzyme function prediction using contrastive learning. *Science*, 379(6639):1358–1363, 2023. doi: 10.1126/science.adf2465. URL https://www.science.org/doi/abs/10.1126/science.adf2465.

Emmanuel Boutet, Damien Lieberherr, Michael Tognolli, Michel Schneider, and Amos Bairoch. UniProtKB/Swiss-Prot. *Methods Mol. Biol.*, 406:89–112, 2007.

Jie Hou, Badri Adhikari, and Jianlin Cheng. DeepSF: deep convolutional neural network for mapping protein sequences to folds. *Bioinformatics*, 34(8):1295–1303, 2018.

Naomi K Fox, Steven E Brenner, and John-Marc Chandonia. SCOPe: Structural classification of proteins—extended, integrating scop and astral data and classification of new structures. *Nucleic acids research*, 42(D1):D304–D309, 2014.

Roshan Rao, Nicholas Bhattacharya, Neil Thomas, Yan Duan, Peter Chen, John Canny, Pieter Abbeel, and Yun Song. Evaluating protein transfer learning with tape. *Advances in neural information processing systems*, 32, 2019.

José Juan Almagro Armenteros, Casper Kaae Sønderby, Søren Kaae Sønderby, Henrik Nielsen, and Ole Winther. DeepLoc: prediction of protein subcellular localization using deep learning. *Bioinformatics*, 33(21):3387–3395, 2017.

Ningyu Zhang, Zhen Bi, Xiaozhuan Liang, Siyuan Cheng, Haosen Hong, Shumin Deng, Jiazhang Lian, Qiang Zhang, and Huajun Chen. Ontoprotein: Protein pretraining with gene ontology embedding. *arXiv preprint arXiv:2201.11147*, 2022a.

Mathias Uhlen, Per Oksvold, Linn Fagerberg, Emma Lundberg, Kalle Jonasson, Mattias Forsberg, Martin Zwahlen, Caroline Kampf, Kenneth Wester, Sophia Hober, et al. Towards a knowledge-based human protein atlas. *Nature biotechnology*, 28(12):1248–1250, 2010.

Theo Sanderson, Maxwell L Bileschi, David Belanger, and Lucy J Colwell. ProteInfer, deep neural networks for protein functional inference. *eLife*, 12:e80942, feb 2023. ISSN 2050-084X. doi: 10.7554/eLife.80942. URL https://doi.org/10.7554/eLife.80942.

Jae Yong Ryu, Hyun Uk Kim, and Sang Yup Lee. Deep learning enables high-quality and high-throughput prediction of enzyme commission numbers. *Proceedings of the National Academy of Sciences*, 116(28):13996–14001, jun 2019. doi: 10.1073/pnas.1821905116. URL https://doi.org/10.1073%2Fpnas.1821905116.

Yu Li, Sheng Wang, Ramzan Umarov, Bingqing Xie, Ming Fan, Lihua Li, and Xin Gao. DEEPre: sequence-based enzyme EC number prediction by deep learning. *Bioinformatics*, 34(5):760–769, oct 2017. doi: 10.1093/bioinformatics/btx680. URL https://doi.org/10.1093%2Fbioinformatics%2Fbtx680.

Alexander Rives, Joshua Meier, Tom Sercu, Siddharth Goyal, Zeming Lin, Jason Liu, Demi Guo, Myle Ott, C Lawrence Zitnick, Jerry Ma, et al. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proceedings of the National Academy of Sciences*, 118(15):e2016239118, 2021.

Minghao Xu, Zuobai Zhang, Jiarui Lu, Zhaocheng Zhu, Yangtian Zhang, Chang Ma, Runcheng Liu, and Jian Tang. PEER: A comprehensive and multi-task benchmark for protein sequence understanding, 2022.

Stacey D Finley, Linda J Broadbelt, and Vassily Hatzimanikatis. Computational framework for predictive biodegradation. *Biotechnol. Bioeng.*, 104(6):1086–1097, December 2009.

Changdai Gu, Gi Bae Kim, Won Jun Kim, Hyun Uk Kim, and Sang Yup Lee. Current status and applications of genome-scale metabolic models. *Genome Biol.*, 20(1):121, June 2019.

Yuichi Kodama, Martin Shumway, Rasko Leinonen, and International Nucleotide Sequence Database Collaboration. The sequence read archive: explosive growth of sequencing data. *Nucleic Acids Res.*, 40(Database issue):D54–6, January 2012.

Jun-Jie Liu, Natalia Orlova, Benjamin L Oakes, Enbo Ma, Hannah B Spinner, Katherine LM Baney, Jonathan Chuck, Dan Tan, Gavin J Knott, Lucas B Harrington, et al. Casx enzymes comprise a distinct family of rna-guided genome editors. *Nature*, 566(7743):218–223, 2019.

Letícia S Tavares, Carolina SF Silva, Vinicius C de Souza, Vânia L da Silva, Cláudio G Diniz, and Marcelo O Santos. Strategies and molecular tools to fight antimicrobial resistance: resistome, transcriptome, and antimicrobial peptides. *Frontiers in microbiology*, 4:412, 2013.

Guoqing Zhang, Hui Wang, Zhiguo Zhang, Lu Zhang, Guibing Guo, Jian Yang, Fajie Yuan, and Feng Ju. Ultra-accurate classification and discovery of functional protein-coding genes from microbiomes using fungenetyper: An expandable deep learning-based framework. *bioRxiv*, pages 2022–12, 2022b.

Mien-Chie Hung and Wolfgang Link. Protein localization in disease and therapy. *Journal of cell science*, 124(20):3381–3392, 2011.

Peter J Thul, Lovisa Åkesson, Mikaela Wiking, Diana Mahdessian, Aikaterini Geladaki, Hammou Ait Blal, Tove Alm, Anna Asplund, Lars Björk, Lisa M Breckels, et al. A subcellular map of the human proteome. *Science*, 356(6340):eaal3321, 2017.

Andreas Schmidt, Ignasi Forne, and Axel Imhof. Bioinformatic analysis of proteomics data. *BMC systems biology*, 8(2):1–7, 2014.

Angelique Stalmach, Ines Boehm, Marco Fernandes, Alison Rutter, Richard JE Skipworth, and Holger Husi. Gene Ontology (GO)-driven inference of candidate proteomic markers associated with muscle atrophy conditions. *Molecules*, 27(17):5514, 2022.

Scott McGinnis and Thomas L. Madden. BLAST: at the core of a powerful and diverse set of sequence analysis tools. *Nucleic Acids Research*, 32(Web Server issue)(W20–W25), 2004. doi: 10.1093/nar/gkh435. URL `https://doi.org/10.1093/nar/gkh435`.

A Appendix

A.1 Examples of training and testing data

A data sample follows a format of "Instruction-Input-Output". The “Input” and “Output” sections consist of key information extracted directly from the Swiss-Prot protein database. The “Instruction” is generated using ChatGPT based on a set of manually created seed instructions. For each protein task, we develop a set of manual instructions, such as “What is the function of the protein sequence?” To generate variations of these instructions, we prompt ChatGPT with requests like, “Could you provide 30 alternative phrasings for ‘What is the function of the protein sequence?’ Please save each variation in a .txt file, one per line without numbering.” We generate approximately 50 high-quality variations for each instruction and manually verify their accuracy. Instructions are then randomly assigned to each Input-Output pair.

An example of training data

"instruction": "What is the EC classification of the input protein sequence based on its biological function?"
"input": "MGLVSSKKPDKEKPIKEKDKGQWSPLKVSAQDKDAPPLPPLVVFNHLTPPPPDE
HLDEDKHFVVALYDYTAMNDRDLQMLKGEKLQVLKGTGDWWLARSLVTGREGYVPSNF
VARVESLEMERWFFRSQGRKEAERQLLAPINKAGSFLIRESETNKGAFSLSVKDVTTQGELIK
HYKIRCLDEGGYYISPRITFPSLQALVQHYSKKGDGLCQRLTLPCVRPAPQNPWAQDEWEIP
RQSLRLVRKLGSGQFGEVWMGYYKNNMKVAIKTLKEGTMSPEAFLGEANVMKALQHERL
VRLYAVVTKEPIYIVTEYMARGCLLDFLKTDEGSRLSLPRLIDMSAQIAEGMAYIERMNSIHR
DLRAANILVSEALCCKIADFGLARIIDSEYTAQEGAKFPIKWTAPEAIHFGVFTIKADVWSFG
VLLMEVVTYGRVPYPGMSNPEVIRNLERGYRMPRPDTCPPELYRGVIAECWRSRPEERPTFE
FLQSVLEDFYTATERQYELQP"
"output": "2.7.10.2"

An example of testing data

{"**id**": "seed_task_0", "**name**": "EC number of price dataset from CLEAN", "**instruction**": "Return the EC number of the protein sequence.", "**instances**": ["**input**": "MAIPPYPDFRSAA
FLRQHLRATMAFYDPVATDASGGQFHFFLDDGTVYNTHTRHLVSATRFVVTHAMLYRTTG
EARYQVGMRHALEFLRTAFLDPATGGYAWLIDWQDGRATVQDTTRHCYGMAFVMLAYA
RAYEAGVPEARVWLAEAFDTAEQHFWQPAAGLYADEASPDWQLTSYRGQNANMHACEA
MISAFRATGERRYIERAEQLAQGICQRQAALSDRTHAPAAEGWVWEHFHADWSVDWDYN
RHDRSNIFRPWGYQVGHQTEWAKLLLQLDALLPADWHLPCAQRLFDTAVERGWDAEHGG
LYYGMAPDGSICDDGKYHWVQAESMAAAAVLAVRTGDARYWQWYDRIWAYCWAHFVD
HEHGAWFRILHRDNRNTTREKSNAGKVDYHNMGACYDVLLWALDAPGFSKESRSAALGR
P", "**output**": "5.3.1.7"], "is_classification": false}

A.2 Pre-training data summary of Llama-3.1 and Galactica

As depicted in[**?**], the pre-training data of Llama-3.1 is composed with general knowledge tokens (50%), mathematical and reasoning tokens (25%), code tokens (17%), and multilingual tokens (8%). According to [Zhao et al., 2023], the pre-training dataset of Galactica is made up of scientific data tokens (83%), Webpages tokens (10%) and Code tokens(7%). Their comparison is shown in Fig. 4.

A.3 Hyperparameters of experiments

Hyperparameters for the instruction-tuning and testing phases are summarized in Table 7.

A.4 Significance of designing each evaluation task

EC number prediction (SU): The EC number describes the catalytic function of enzymes, which is a major class of protein that accelerates chemical reaction to maintain steady biological activities. Through the study of enzyme catalytic function, we can further understand metabolic pathways and accelerate designing new metabolic pathways[Finley et al., 2009], building genome-scale metabolic

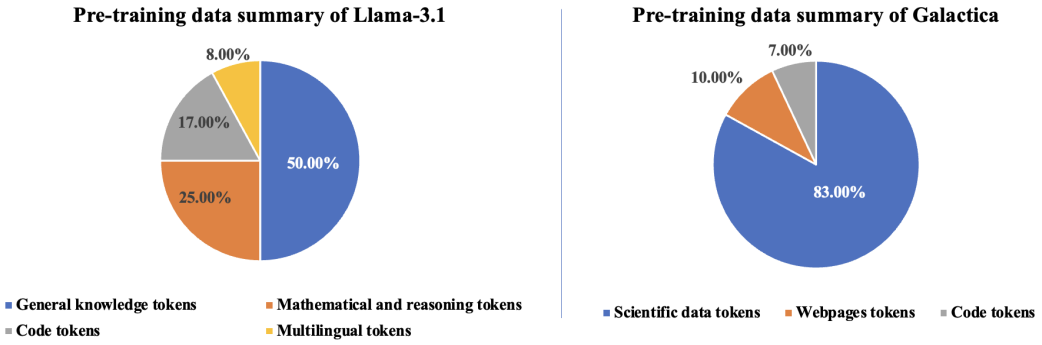


Figure 4: **Comparison of the pre-training data summary of Llama-3.1 and Galactica.**

Table 7: **Hyperparameters for the instruction-tuning and testing phases.**

Hyperparameters	Instruction-tuning	Testing
DeepSpeed ZeRO Stage	3	-
optimizer	AdamW	-
optimizer hyperparameters	(β_1,β_2)=(0.9, 0.999), eps=1e-8	-
per_device_train_batch_size	4	-
gradient_accumulation_steps	4	-
gradient_checkpointing	True	-
learning rate	2e-5	-
weight decay	0.0	-
warmup ratio	0.03	-
lr scheduler type	cosine	-
training epochs	1	-
GPU	8*A100	1*A100
temperature	-	0.2
top_k	-	50
top_p	-	0.75
num_beams	-	1
max_new_tokens	-	400
use_cache	-	True
do_sample	-	True
eos_token_id	-	tokenizer.eos_token_id
pad_token_id	-	tokenizer.pad_token_id

models of bacteria, archaea and eukarya[Gu et al., 2019] and annotating the fast-growing next generation sequencing data[Kodama et al., 2012].

Fold type prediction (SU): Through fold type prediction task, we can inspect the capability of models to identify structural similarities from distantly related sequences. It is of much significance in numerous biological problems, such as finding novel sequences with similar structures but different sequences (e.g., enzyme design[Liu et al., 2019]), and detecting distant homologous based functional genes (e.g., antibiotic resistance genes[Tavares et al., 2013, Zhang et al., 2022b]).

Subcellular localization prediction (SU): Abnormal subcellular localization of proteins affects their functions and are pathogenesis of many human diseases, like cardiovascular diseases and cancers[Hung and Link, 2011]. Therefore, identifying the subcellular localization of proteins can provide crucial clues to understand the mechanism of cellular actions and biomolecular interactions, as well as identification for drug discovery[Thul et al., 2017].

Keywords prediction (AP): Basically, function keywords embody the summary of structures and functions of a protein sequence, and have great potential to be utilized to mine potential functional proteins from unannotated sequences.

GO terms prediction (AP): As a widely-accepted sequence-functional classification scheme, GO terms play a crucial role in many aspects of protein biology research, such as assigning functions to protein domains and integrating proteomic information from different organisms[Schmidt et al., 2014, Stalmach et al., 2022]. Currently, GO terms have been associated with many other biomedical ontologies and have become fundamental to the application of computer science to biomedical research.

Function prediction (AP): Function description provides the richest human-readable textual information of a protein, describing in details the whole discovery process of this protein in the course of bioinformatics development. Undoubtedly, the detailed function descriptions about proteins could greatly assist practitioners in related research fields and accelerate the development of protein biology.

Tissue location prediction from gene symbol (KM): The location of proteins in human tissues can further assist in revealing their functional properties, so they are reported by many proteomics databases and biomedical literature, which can well show the information mining capabilities of LLMs.

Cancer prediction from gene symbol and gene name (KM): Cancers are major threats to human health, thus understanding the relationship between gene encoded proteins and the development of cancers is quite important. For decades, researchers contribute lots of scientific insights about cancers and proteins in numerous literature. Biomedical text mining is a long-term research topic, and this task can evaluate the performance of LLMs in extracting knowledge from pre-training scientific literatures.

A.5 Metrics for multi-label tasks

The multi-label task means that each sample has multiple corresponding labels. For the data set with m samples, the Precision, Recall and F1 value are defined as follows:

$$\text{Precision} = \frac{1}{m}\sum_{i=1}^{m}\frac{\left|y^{(i)} \cap \hat{y}^{(i)}\right|}{\left|\hat{y}^{(i)}\right|}$$

$$\text{Recall} = \frac{1}{m}\sum_{i=1}^{m}\frac{\left|y^{(i)} \cap \hat{y}^{(i)}\right|}{\left|y^{(i)}\right|}$$

$$\text{F1} = \frac{1}{m}\sum_{i=1}^{m}\frac{2\left|y^{(i)} \cap \hat{y}^{(i)}\right|}{\left|y^{(i)}\right| + \left|\hat{y}^{(i)}\right|}$$

where $y^{(i)}$ is true labels of the i-th sample, and $\hat{y}^{(i)}$ is predicted labels.

A.6 Example of input and label for each task

As shown in Table 8, the tasks exhibit diverse input formats, with SU and AP tasks containing protein sequences, while KM tasks take gene symbols or names as input. There are also variations in label formats; for example, SU tasks involve numerical labels for EC number and fold type prediction, while other tasks use character labels. This diversity in both input types and label formats highlights the heterogeneous nature of the tasks, presenting distinct challenges for model design and performance optimization across different prediction objectives.

Furthermore, Fig. 5 presents the distribution of label counts for tasks with multiple items per label, as well as the variation in description length for the function prediction task. Tasks with single-label samples or predominantly single-label distributions are excluded from this figure. Notably, the tasks of keyword prediction, GO term prediction, and gSymbol2Tissue contain a significant proportion of samples with more than five items per label, underscoring the complexity of these tasks. Furthermore, the majority of samples in the function description task contain over 25 words, indicating the presence of rich and detailed information in the descriptions.

Table 8: **Example of input and label for each task.** In SU and AP, the input data contain protein sequences, while in KM, they contain gene symbols or names. Each task is associated with specific labels format, such as numerical labels for EC number and fold type prediction, and character labels for subcellular localization prediction in SU, as well as all tasks in AP and KM.

Task category	Task name	Input	Label
SU 4.2.1	EC number prediction	Protein sequence	2.7.10.2
	Fold type prediction	Protein sequence	10
	Subcellular localization prediction	Protein sequence	membrane
AP 4.2.2	Keywords prediction	Protein sequence	Chloroplast;DNA-directedRNApolymerase;Metal-binding;Nucleotidyltransferase;Plastid;Transcription;Transferase;Zinc
	GO terms prediction	Protein sequence	plasma membrane; alpha,alpha-trehalase activity; trehalase activity; trehalose catabolic process
	Function prediction	Protein sequence	SpecificallycatalyzesthedecarboxylationofL-arginine toagmatine.HasnoS-adenosylmethioninedecarboxylase(AdoMetDC)activity.
KM 4.2.3	gSymbol2Tissue	Gene symbol	bone marrow; lymph node; oral mucosa; spleen
	gSymbol2Cancer	Gene symbol	peripheral T-cell lymphoma
	gName2Cancer	Gene name	acute megakaryocytic leukaemia; ETP ALL

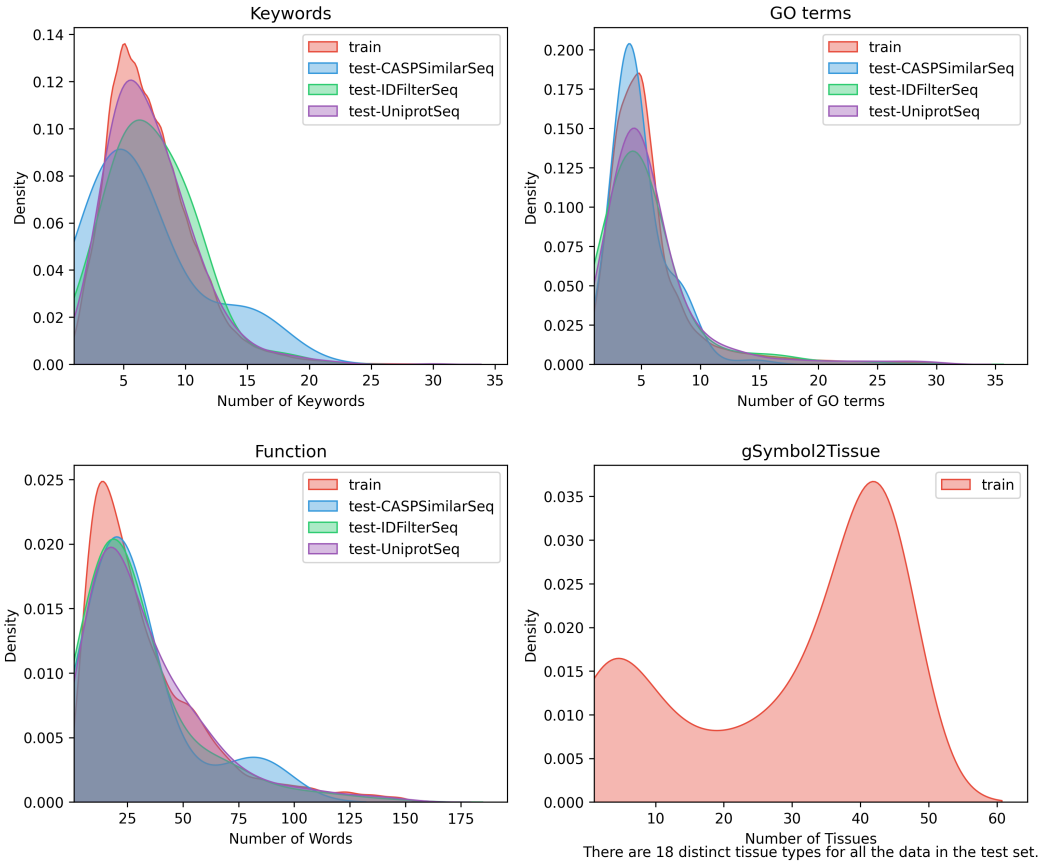


Figure 5: **Label count distribution and function description length variation in multi-label tasks.** Tasks with single-label samples or predominantly single-label distributions are excluded from this figure.

A.7 Detailed explanation for the testing set of fold type prediction task

A protein fold type can be further subdivided into superfamilies, wherein all members share evolutionary relationships and a common ancestor. Further, each superfamily is classified into families

based on sequence identity. In this study, we employ three testing sets at different hierarchical levels: fold level, superfamily level, and family level. The fold-level testing set ensures that sequences from the testing set do not share any superfamily-level overlap with the training set by holding out entire superfamilies exclusively for testing. Similarly, the superfamily-level testing set guarantees that there is no intersection at the family level between the training and testing sets. For the family-level testing set, 20% of sequences within each family are allocated to the testing set, ensuring a more nuanced evaluation of model performance.

Among these, the fold-level testing set presents the greatest challenge due to its minimal similarity to the training set, followed by the superfamily level. The family-level testing set, on the other hand, maintains the highest degree of similarity to the training data, allowing for consistency in evaluation. This hierarchical approach enables a thorough assessment of model generalization across different levels of protein sequence similarity and evolutionary relationships.

A.8 Dataset construction details of the annotation prediction tasks

Our dataset is constructed employing Swiss-Port database including 565,861 samples (data as of January 2022, which is officially named 2022_01 relsease in UniProtKB/Swiss-Prot protein knowledgebase [**]). The function keywords, go terms and function description of some protein sequences are missing. Therefore, we delete the samples where one of them is empty, and finally filter our dataset to 457,476 samples, called Swiss-Port-all. Next, three testing sets are hold-out from it through three steps, and the last remaining samples are used as the training set.

Step 1: Compared to 51 target sequences released by CASP14, the sequences whose sequence identity $\geq$ 50% in Swiss-Port-all with one of CASP14 sequences are removed through BLAST[McGinnis and Madden, 2004], with a total of 184 samples, called **CASPSimilarSeq**.

Step 2: For the remaining 457,292 samples, we clustered these sequences by setting sequence identity $\geq$ 80% using CD-HIT. We randomly selected all samples of 500 clusters, a total of 1,112 sequences, as our second testing set, termed as **IDFilterSeq**.

Step 3: For the remaining 456,180 sequences from the second step, We randomly selected 1% of the sequences as the third testing set, with 4,562 sequences, named **UniProtSeq**. All remaining 451,618 samples are used for training.

Thence, for the three tasks of function keywords prediction, go terms prediction and protein function prediction, we respectively construct corresponding training sets and three types of testing sets. Moreover, for three testing sets, UniProtSeq is the most similar to the training set, followed by IDFilterSeq and CASPSimilarSeq is the least similar to the training set.

A.9 Comparison of GO terms and function keywords

As of February 2023, the Swiss-Prot database contains a total of 1,191 unique function keywords, classified into the following ten categories: (1) Biological process (495); (2) Cellular component (158); (3) Coding sequence diversity (13); (4) Developmental stage (9); (5) Disease (156); (6) Domain (35); (7) Ligand (69); (8) Molecular function (198); (9) PTM (47); (10) Technical term (11). In contrast, Gene Ontology (GO) organizes biological knowledge into three categories: molecular function, biological process, and cellular component. As of January 2023, GO includes 27,942 terms for molecular functions, 11,263 terms for biological processes, and 4,043 terms for cellular components. While Swiss-Prot function keywords offer a broad overview of functional descriptions, GO provides a more detailed classification within these key categories, which is crucial for supporting detailed biomedical research.

A.10 Evaluation results

A.10.1 Evaluation of OPI-Llama-3.1-8B-Instruct model on nine tasks

Each testing result shown in Table 9 is derived from the Llama-3.1-8B-Instruct model that has been fine-tuned using the full OPI dataset and subsequently evaluated on the respective testing set for each specific task.

[**] `https://ftp.uniprot.org/pub/databases/uniprot/previous_releases/release-2022_01/knowledgebase/UniProtKB_SwissProt-relstat.html`

Table 9: **Evaluation results of OPI-Llama-3.1-8B-Instruct on nine tasks.**

Task Type	Task Name	Testing set	Accuracy	Precision	Recall	F1	Rouge-L
SU	EC number prediction	CLEAN_EC_number_new_test	-	0.3724	0.3374	0.3468	-
		CLEAN_EC_number_price_test	-	0.0738	0.0738	0.0738	-
	Fold type prediction	fold_type_test_Fold_Holdout	0.1045	-	-	-	-
		fold_type_test_Superfamily_Holdout	0.1507	-	-	-	-
		fold_type_test_Family_Holdout	0.6145	-	-	-	-
	Subcellular localization prediction	subcell_loc_test	0.4214	-	-	-	-
AP	Keywords prediction	CASPSimilarSeq_keywords_test	-	0.4202	0.5057	0.4385	-
		IDFilterSeq_keywords_test	-	0.6762	0.6905	0.6650	-
		UniProtSeq_keywords_test	-	0.7606	0.7489	0.7374	-
	GO terms prediction	CASPSimilarSeq_go_terms_test	-	0.1113	0.0936	0.0990	-
		IDFilterSeq_go_terms_test	-	0.6686	0.6287	0.6304	-
		UniProtSeq_go_terms_test	-	0.7150	0.6897	0.6849	-
	Function prediction	CASPSimilarSeq_function_test	-	-	-	-	0.7524
		IDFilterSeq_function_test	-	-	-	-	0.4786
		UniProtSeq_function_test	-	-	-	-	0.5144
KM	gSymbol2Tissue	gene_symbol_to_tissue_test	-	0.4002	0.9356	0.5466	-
	gSymbol2Cancer	gene_symbol_to_cancer_test	-	0.2890	0.2701	0.2664	-
	gName2Cancer	gene_name_to_cancer_test	-	0.2786	0.2707	0.2659	-

A.10.2 Evaluation of OPI-Galactica-6.7B model on nine tasks

Each testing result shown in Table 10 is derived from the Galactica-6.7B model that has been fine-tuned using the full OPI dataset and subsequently evaluated on the respective testing set for each specific task.

Table 10: **Evaluation results of OPI-Galactica-6.7B model on various tasks.**

Task Type	Task Name	Testing file	Accuracy	Precision	Recall	F1	Rouge-L
SU	EC number prediction	CLEAN_EC_number_new_test	-	0.2700	0.2663	0.2596	-
		CLEAN_EC_number_price_test	-	0.0268	0.0268	0.0268	-
	Fold type prediction	fold_type_test_Fold_Holdout	0.0808	-	-	-	-
		fold_type_test_Superfamily_Holdout	0.1348	-	-	-	-
		fold_type_test_Family_Holdout	0.4854	-	-	-	-
	Subcellular localization prediction	subcell_loc_test	0.7771	-	-	-	-
AP	Keywords prediction	CASPSimilarSeq_keywords_test	-	0.8120	0.7360	0.7643	-
		IDFilterSeq_keywords_test	-	0.8377	0.8019	0.8070	-
		UniProtSeq_keywords_test	-	0.8596	0.8196	0.8276	-
	GO terms prediction	CASPSimilarSeq_go_terms_test	-	0.7613	0.7492	0.7476	-
		IDFilterSeq_go_terms_test	-	0.7404	0.7274	0.7207	-
		UniProtSeq_go_terms_test	-	0.7638	0.7373	0.7358	-
	Function prediction	CASPSimilarSeq_function_test	-	-	-	-	0.7430
		IDFilterSeq_function_test	-	-	-	-	0.7014
		UniProtSeq_function_test	-	-	-	-	0.7133
KM	gSymbol2Tissue	gene_symbol_to_tissue_test	-	0.3917	0.9077	0.5303	-
	gSymbol2Cancer	gene_symbol_to_cancer_test	-	0.3555	0.3189	0.3229	-
	gName2Cancer	gene_name_to_cancer_test	-	0.2728	0.2554	0.2533	-

A.10.3 A comparative analysis of OPI-Llama-3.1-8B-Instruct and OPI-Galactica-6.7B

As depicted in Fig. 6, the Llama-3.1 model consistently outperforms Galactica across different testing sets of the EC number prediction task, as well as on the fold type prediction task. The similarity between these two tasks is that both of their prediction targets are numeric type, such as *3.4.11.4* and *10*. This is probably attributed to a large volume of mathematical and reasoning tokens for Llama-3.1 pre-training. For the tasks whose prediction targets are character type, Galactica consistently surpasses Llama-3.1 model, particularly on the three AP tasks and cancer prediction from gene symbols. On the other tasks, the performance is relatively balanced between the two models.

A.11 Baseline model details for fold type prediction and subcellular localization prediction

The **Transformer baseline model** is a simple Transformer architecture with 2 blocks without pre-training. Conversely, the **ESM-1b baseline model** is pre-trained on UniRef50 dataset with 33 blocks. The corresponding input and feed-forward network (FFN) layers vector dimensions of these two models are 256 and 512, 1280 and 5120, respectively. Unlike LLMs, which treat folding type prediction and subcellular localization prediction as generative tasks, the two baseline models based

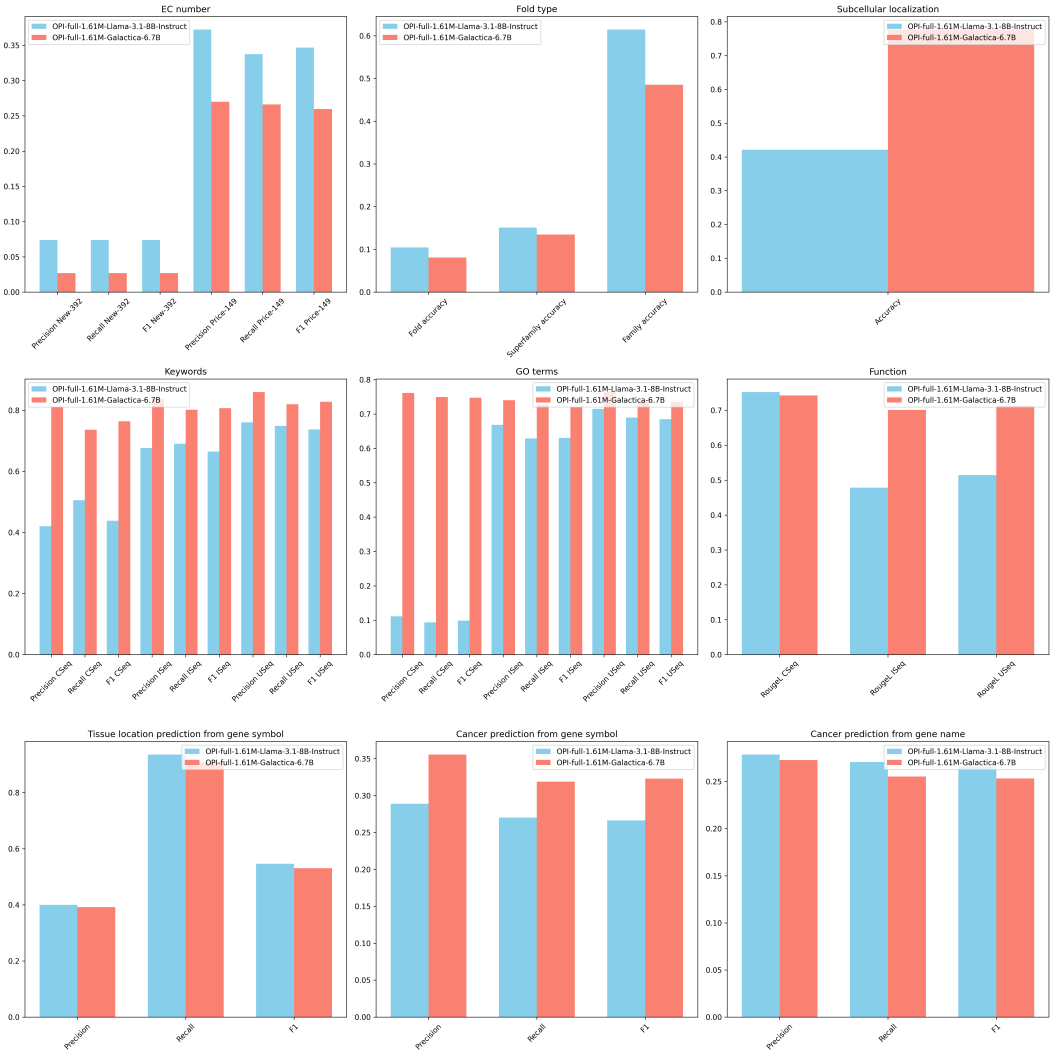
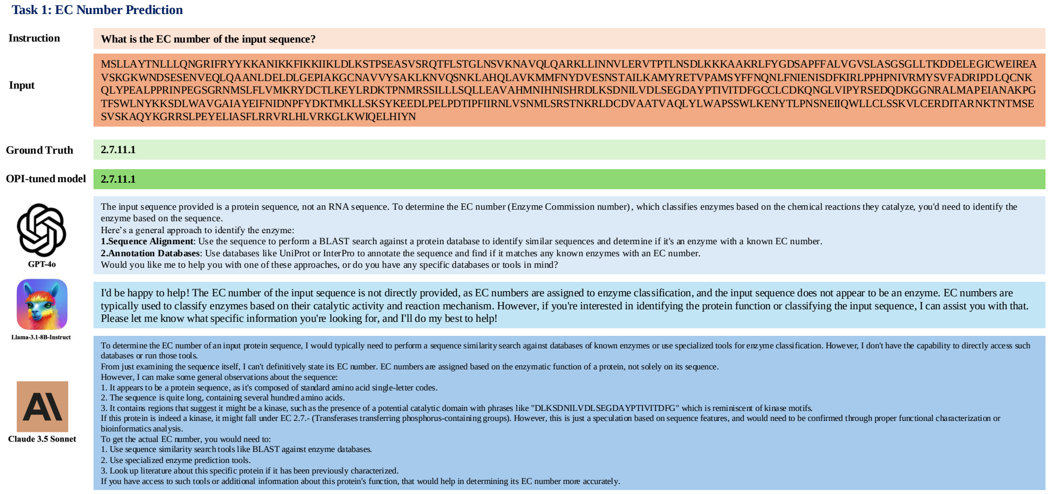


Figure 6: **Comparison of performance between OPI-Llama-3.1-8B-Instruct and OPI-Galactica-6.7B across nine tasks**. It highlights task-specific strengths of each model, with Llama-3.1 excelling in EC number prediction and fold type prediction, while Galactica leads in all three AP tasks, as well as cancer prediction from gene symbols.

on representation learning regard them as classification tasks corresponding to 1195 and 10 categories, respectively. For both tasks, the models extract the representation of the last layer of <CLS> token and add a non-linear layer on top of it to predict the corresponding class of the sequence.

A.12 Prediction comparison of OPI-tuned and state-of-the-art general purpose models

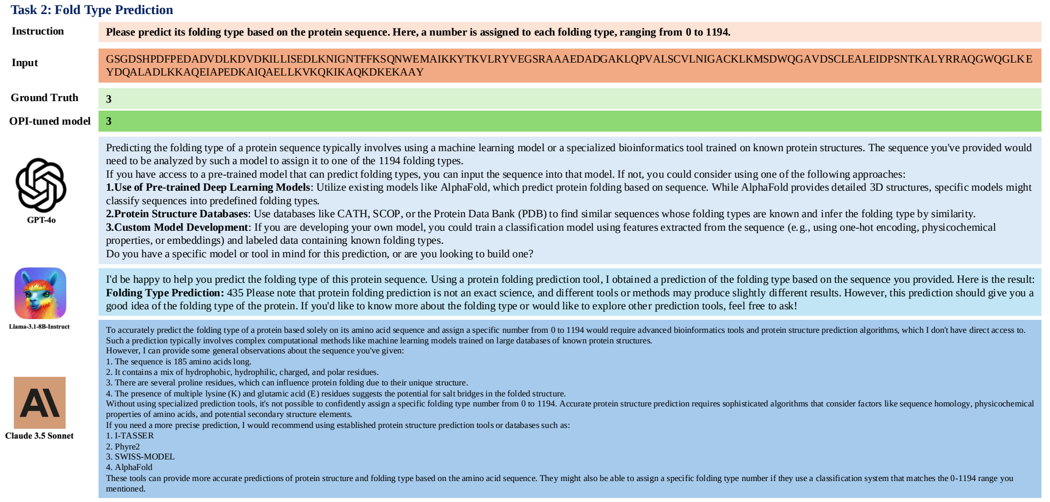
In this section, we compare prediction performance of the OPI-tuned model and state-of-the-art general purpose models, such as GPT-4o, Llama-3.1-8B-Instruct, and Claude 3.5 Sonnet. Fig. 7 illustrates the accuracy and reliability of each model across nine protein-related tasks. By examining the discrepancies and congruences between the models' predictions and the true answers, we can identify the strengths and weaknesses inherent to each model. These insights are instrumental in guiding further fine-tuning efforts and optimizing model selection for particular applications. Ultimately, this analysis is essential for enhancing model robustness and ensuring that the OPI-tuned model is well-suited to the intended use case for protein biology.



Task 1: EC Number Prediction

Instruction	**What is the EC number of the input sequence?**
Input	MSLLAYTNLLLQNGRIFRYYKKANIKKFIKKIIKLDLKSTPSEASVSRQTFLSTGLNSVKNAVQLQARKLLINNVLERVTPTLNSDLKKKAAKRLFYGDSAPFFALVGVSLASGSGLLTKDDELEGICWEIREAVSKGKWNDSESENVEQLQAANLDELDLGEPIAKGCNAVVYSAKLKNVQSNKLAHQLAVKMMFNYDVESNSTAILKAMYRETVPAMSYFFNQNLFNIENISDFKIRLPPHPNIVRMYSVFADRIPDLQCNKQLYPEALPPRINPEGSGRNMSLFLVMKRYDCTLKEYLRDKTPNMRSSILLLSQLLEAVAHMNIHNISHRDLKSDNILVDLSEGDAYPTIVITDFGCCLCDKQNGLVIPYRSEDQDKGGNRALMAPEIANAKPGTFSWLNYKKSDLWAVGAIAYEIFNIDNPFYDKTMKLLSKSYKEEDLPELPDTIPFIIRNLVSNMLSRSTNKRLDCDVAATVAQLYLWAPSSWLKENYTLPNSNEIIQWLLCLSSKVLCERDITARNKTNTMSESVSKAQYKGRRSLPEYELIASFLRRVRLHLVRKGLKWIQELHIYN
Ground Truth	**2.7.11.1**
OPI-tuned model	**2.7.11.1**
GPT-4o	The input sequence provided is a protein sequence, not an RNA sequence. To determine the EC number (Enzyme Commission number), which classifies enzymes based on the chemical reactions they catalyze, you'd need to identify the enzyme based on the sequence. Here's a general approach to identify the enzyme: **1.Sequence Alignment**: Use the sequence to perform a BLAST search against a protein database to identify similar sequences and determine if it's an enzyme with a known EC number. **2.Annotation Databases**: Use databases like UniProt or InterPro to annotate the sequence and find if it matches any known enzymes with an EC number. Would you like me to help you with one of these approaches, or do you have any specific databases or tools in mind?
Llama-3.1-8B-Instruct	I'd be happy to help! The EC number of the input sequence is not directly provided, as EC numbers are assigned to enzyme classification, and the input sequence does not appear to be an enzyme. EC numbers are typically used to classify enzymes based on their catalytic activity and reaction mechanism. However, if you're interested in identifying the protein function or classifying the input sequence, I can assist you with that. Please let me know what specific information you're looking for, and I'll do my best to help!
Claude 3.5 Sonnet	To determine the EC number of an input protein sequence, I would typically need to perform a sequence similarity search against databases of known enzymes or use specialized tools for enzyme classification. However, I don't have the capability to directly access such databases or run those tools. From just examining the sequence itself, I can't definitively state its EC number. EC numbers are assigned based on the enzymatic function of a protein, not solely on its sequence. However, I can make some general observations about the sequence: 1. It appears to be a protein sequence, as it's composed of standard amino acid single-letter codes. 2. The sequence is quite long, containing several hundred amino acids. 3. It contains regions that suggest it might be a kinase, such as the presence of a potential catalytic domain with phrases like "DLKSDNILVDLSEGDAYPTIVITDFG" which is reminiscent of kinase motifs. If this protein is indeed a kinase, it might fall under EC 2.7.- (Transferases transferring phosphorus-containing groups). However, this is just a speculation based on sequence features, and would need to be confirmed through proper functional characterization or bioinformatics analysis. To get the actual EC number, you would need to: 1. Use sequence similarity search tools like BLAST against enzyme databases. 2. Use specialized enzyme prediction tools. 3. Look up literature about this specific protein if it has been previously characterized. If you have access to such tools or additional information about this protein's function, that would help in determining its EC number more accurately.

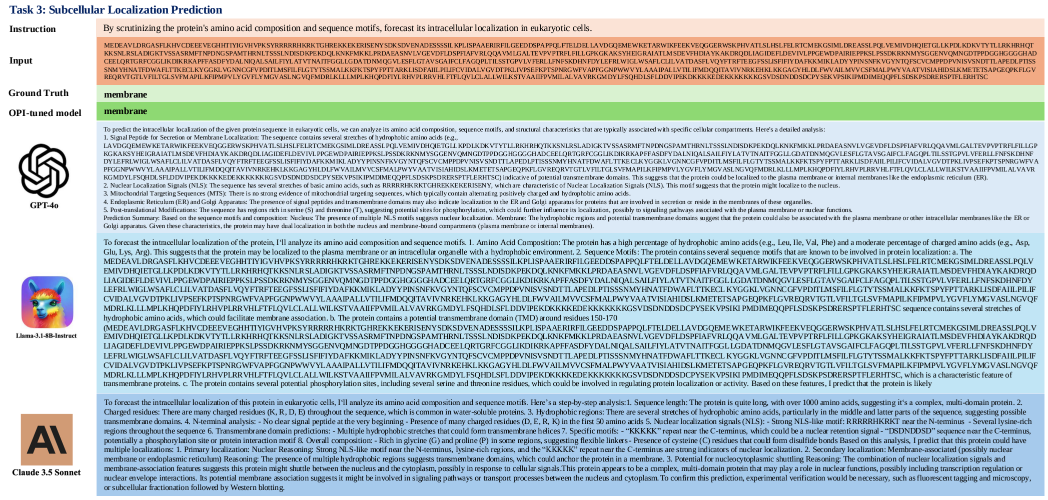
(a) Task 1: EC number prediction



Task 2: Fold Type Prediction

Instruction	**Please predict its folding type based on the protein sequence. Here, a number is assigned to each folding type, ranging from 0 to 1194.**
Input	GSGDSHPDFPEDADVDLKDVDKILLISEDLKNIGNTFFKSQNWEMAIKKYTKVLRYVEGSRAAAEDADGAKLQPVALSCVLNIGACKLKMSDWQGAVDSCLEALEIDPSNTKALYRRAQGWQGLKEYDQALADLKKAQEIAPEDKAIQAELLKVKQKIKAQKDKEKAAY
Ground Truth	**3**
OPI-tuned model	**3**
GPT-4o	Predicting the folding type of a protein sequence typically involves using a machine learning model or a specialized bioinformatics tool trained on known protein structures. The sequence you've provided would need to be analyzed by such a model to assign it to one of the 1194 folding types. If you have access to a pre-trained model that can predict folding types, you can input the sequence into that model. If not, you could consider using one of the following approaches: **1.Use of Pre-trained Deep Learning Models**: Utilize existing models like AlphaFold, which predict protein folding based on sequence. While AlphaFold provides detailed 3D structures, specific models might classify sequences into predefined folding types. **2.Protein Structure Databases**: Use databases like CATH, SCOP, or the Protein Data Bank (PDB) to find similar sequences whose folding types are known and infer the folding type by similarity. **3.Custom Model Development**: If you are developing your own model, you could train a classification model using features extracted from the sequence (e.g., using one-hot encoding, physicochemical properties, or embeddings) and labeled data containing known folding types. Do you have a specific model or tool in mind for this prediction, or are you looking to build one?
Llama-3.1-8B-Instruct	I'd be happy to help you predict the folding type of this protein sequence. Using a protein folding prediction tool, I obtained a prediction of the folding type based on the sequence you provided. Here is the result: **Folding Type Prediction:** 435 Please note that protein folding prediction is not an exact science, and different tools or methods may produce slightly different results. However, this prediction should give you a good idea of the folding type of the protein. If you'd like to know more about the folding type or would like to explore other prediction tools, feel free to ask!
Claude 3.5 Sonnet	To accurately predict the folding type of a protein based solely on its amino acid sequence and assign a specific number from 0 to 1194 would require advanced bioinformatics tools and protein structure prediction algorithms, which I don't have direct access to. Such a prediction typically involves complex computational methods like machine learning models trained on large databases of known protein structures. However, I can provide some general observations about the sequence you've given: 1. The sequence is 185 amino acids long. 2. It contains a mix of hydrophobic, hydrophilic, charged, and polar residues. 3. There are several proline residues, which can influence protein folding due to their unique structure. 4. The presence of multiple lysine (K) and glutamic acid (E) residues suggests the potential for salt bridges in the folded structure. Without using specialized prediction tools, it's not possible to confidently assign a specific folding type number from 0 to 1194. Accurate protein structure prediction requires sophisticated algorithms that consider factors like sequence homology, physicochemical properties of amino acids, and potential secondary structure elements. If you need a more precise prediction, I would recommend using established protein structure prediction tools or databases such as: 1. I-TASSER 2. Phyre2 3. SWISS-MODEL 4. AlphaFold These tools can provide more accurate predictions of protein structure and folding type based on the amino acid sequence. They might also be able to assign a specific folding type number if they use a classification system that matches the 0-1194 range you mentioned.

(b) Task 2: Fold type prediction



Task 3: Subcellular Localization Prediction

Instruction	By scrutinizing the protein's amino acid composition and sequence motifs, forecast its intracellular localization in eukaryotic cells.
Input	MEDEAVLDRGASFLKHVCDEEEVEGHHTIYIGVHVPKSYRRRRRHKRKTGHREKKEKERISENYSDKSDVENADESSSILKPLISPAAERIRFILGEEDDSPAPPQLFTELDELLAVDGQEMEWKETARWIKFEEKVEQGGERWSKPHVATLSLHSLFELRTCMEKGSIMLDREASSLPQLVEMIVDHQIETGLLKPDLKDKVTYTLLRKHRHQTKKSNLRSLADIGKTVSSASRMFTNPDNGSPAMTHRNLTSSSLNDISDKPEKDQLKNKFMKKLPRDAEASNVLVGEVDFLDSPFIAFVRLQQAVMLGALTEVPVPTRFLFILLGPKGKAKSYHEIGRAIATLMSDEVFHDIAYKAKDRQDLIAGIDEFLDEVIVLPPGEWDPAIRIEPPKSLPSSDKRKNMYSGGENVQMNGDTPPDGGHGGGHADCEELQRTGRFCGGLIKDIKRKAPFFASDFYDALNIQALSAILFIYLATVTNAITFGGLLGDATDNMQGVLESFLGTAVSGAIFCLFAGQPLTILSSTGPVLVFERLLFNFSKDHNFDYLEFRLWIGLWSAFLCLILVATDASFLVQYFTRFTEEGFSSLISFIFIYDAFKKMIKLADYYPINSNFKVGYNTQFSCVCMPPDPVNISVSNDTTLAPEDLPTISSSNMYHNATFDWAFLTTKECLKYGGKLVGNNCGFVPDITLMSFILFLGTYTSSMALKKFKTSPYFPTTARKLISDFAIILPILIFCVIDALVGVDTPKLIVPSEFKPTSPNRGWFVAPFGGNPWWVYLAAAIPALLVTILIFMDQQITAVIVNRKEHKLKKGAGYHLDLFWVAILMVVCSFMALPWYVAATVISIAHIDSLKMETETSAPGEQPKFLGVREQRVTGTLVFILTGLSVFMAPILKFIPMPVLYGVFLYMGVASLNGVQFMDRLKLLLMPLKHQPDFIYLRHVPLRRVHLFTFLQVLCLALLWILKSTVAAIIFPVMILALVAVRKGMDYLFSQHDLSFLDDVIPEKDKKKEDEKKKKKKGSVDSDNDDSDCPYSEKVPSIKIPMDIMEQQPFLSDSKPSDRERSPTFLERHTSC
Ground Truth	**membrane**
OPI-tuned model	**membrane**
GPT-4o	To predict the intracellular localization of the given protein sequence in eukaryotic cells, we can analyze its amino acid composition, sequence motifs, and structural characteristics that are typically associated with specific cellular compartments. Here's a detailed analysis: 1. Signal Peptide for Secretion or Membrane Localization: The sequence contains several stretches of hydrophobic amino acids (e.g., LAVDGQEMEWKETARWIKFEEKVEQGGERWSKPHVATLSLHSLFELRTCMEKGSIMLDREASSLPQLVEMIVDHQIETGLLKPDLKDKVTYTLLRKHRHQTKKSNLRSLADIGKTVSSASRMFTNPDNGSPAMTHRNLTSSSLNDISDKPEKDQLKNKFMKKLPRDAEASNVLVGEVDFLDSPFIAFVRLQQAVMLGALTEVPVPTRFLFILLGPKGKAKSYHEIGRAIATLMSDEVFHDIAYKAKDRQDLIAGIDEFLDEVIVLPPGEWDPAIRIEPPKSLPSSDKRKNMYSGGENVQMNGDTPPDGGHGGGHADCEELQRTGRFCGGLIKDIKRKAPFFASDFYDALNIQALSAILFIYLATVTNAITFGGLLGDATDNMQGVLESFLGTAVSGAIFCLFAGQPLTILSSTGPVLVFERLLFNFSKDHNFDYLEFRLWIGLWSAFLCLILVATDASFLVQYFTRFTEEGFSSLISFIFIYDAFKKMIKLADYYPINSNFKVGYNTQFSCVCMPPDPVNISVSNDTTLAPEDLPTISSSNMYHNATFDWAFLTTKECLKYGGKLVGNNCGFVPDITLMSFILFLGTYTSSMALKKFKTSPYFPTTARKLISDFAIILPILIFCVIDALVGVDTPKLIVPSEFKPTSPNRGWFVAPFGGNPWWVYLAAAIPALLVTILIFMDQQITAVIVNRKEHKLKKGAGYHLDLFWVAILMVVCSFMALPWYVAATVISIAHIDSLKMETETSAPGEQPKFLGVREQRVTGTLVFILTGLSVFMAPILKFIPMPVLYGVFLYMGVASLNGVQFMDRLKLLLMPLKHQPDFIYLRHVPLRRVHLFTFLQVLCLALLWILKSTVAAIIFPVMILALVAVRKGMDYLFSQHDLSFLDDVIPEKDKKKEDEKKKKKKGSVDSDNDDSDCPYSEKVPSIKIPMDIMEQQPFLSDSKPSDRERSPTFLERHTSC) indicative of potential transmembrane domains. This suggests that the protein could be localized to the plasma membrane or internal membranes like the endoplasmic reticulum (ER). 2. Nuclear Localization Signals (NLS): The sequence has several stretches of basic amino acids, such as RRRRRHKRKTGHREKKEKERISENY, which are characteristic of Nuclear Localization Signals (NLS). This motif suggests that the protein might localize to the nucleus. 3. Mitochondrial Targeting Sequences (MTS): There is no strong evidence of mitochondrial targeting sequences, which typically contain alternating positively charged and hydrophobic amino acids. 4. Endoplasmic Reticulum (ER) and Golgi Apparatus: The presence of signal peptides and transmembrane domains may also indicate localization to the ER and Golgi apparatus for proteins that are involved in secretion or reside in the membranes of these organelles. 5. Post-translational Modifications: The sequence has regions rich in serine (S) and threonine (T), suggesting potential sites for phosphorylation, which could further influence its localization, possibly to signaling pathways associated with the plasma membrane or nuclear functions. Prediction Summary: Based on the sequence motifs and composition: Nucleus: The presence of multiple NLS motifs suggests nuclear localization. Membrane: The hydrophobic regions and potential transmembrane domains suggest that the protein could also be associated with the plasma membrane or other intracellular membranes like the ER or Golgi apparatus. Given these characteristics, the protein may have dual localization in both the nucleus and membrane-bound compartments (plasma membrane or internal membranes).
Llama-3.1-8B-Instruct	To forecast the intracellular localization of the protein, I'll analyze its amino acid composition and sequence motifs. 1. Amino Acid Composition: The protein has a high percentage of hydrophobic amino acids (e.g., Leu, Ile, Val, Phe) and a moderate percentage of charged amino acids (e.g., Asp, Glu, Lys, Arg). This suggests that the protein may be localized to the plasma membrane or an intracellular organelle with a hydrophobic environment. 2. Sequence Motifs: The protein contains several sequence motifs that are known to be involved in protein localization: a. The MEDEAVLDRGASFLKHVCDEEEVEGHHTIYIGVHVPKSYRRRRRHKRKTGHREKKEKERISENYSDKSDVENADESSSILKPLISPAAERIRFILGEEDDSPAPPQLFTELDELLAVDGQEMEWKETARWIKFEEKVEQGGERWSKPHVATLSLHSLFELRTCMEKGSIMLDREASSLPQLVEMIVDHQIETGLLKPDLKDKVTYTLLRKHRHQTKKSNLRSLADIGKTVSSASRMFTNPDNGSPAMTHRNLTSSSLNDISDKPEKDQLKNKFMKKLPRDAEASNVLVGEVDFLDSPFIAFVRLQQAVMLGALTEVPVPTRFLFILLGPKGKAKSYHEIGRAIATLMSDEVFHDIAYKAKDRQDLIAGIDEFLDEVIVLPPGEWDPAIRIEPPKSLPSSDKRKNMYSGGENVQMNGDTPPDGGHGGGHADCEELQRTGRFCGGLIKDIKRKAPFFASDFYDALNIQALSAILFIYLATVTNAITFGGLLGDATDNMQGVLESFLGTAVSGAIFCLFAGQPLTILSSTGPVLVFERLLFNFSKDHNFDYLEFRLWIGLWSAFLCLILVATDASFLVQYFTRFTEEGFSSLISFIFIYDAFKKMIKLADYYPINSNFKVGYNTQFSCVCMPPDPVNISVSNDTTLAPEDLPTISSSNMYHNATFDWAFLTTKECLKYGGKLVGNNCGFVPDITLMSFILFLGTYTSSMALKKFKTSPYFPTTARKLISDFAIILPILIFCVIDALVGVDTPKLIVPSEFKPTSPNRGWFVAPFGGNPWWVYLAAAIPALLVTILIFMDQQITAVIVNRKEHKLKKGAGYHLDLFWVAILMVVCSFMALPWYVAATVISIAHIDSLKMETETSAPGEQPKFLGVREQRVTGTLVFILTGLSVFMAPILKFIPMPVLYGVFLYMGVASLNGVQFMDRLKLLLMPLKHQPDFIYLRHVPLRRVHLFTFLQVLCLALLWILKSTVAAIIFPVMILALVAVRKGMDYLFSQHDLSFLDDVIPEKDKKKEDEKKKKKKGSVDSDNDDSDCPYSEKVPSIKIPMDIMEQQPFLSDSKPSDRERSPTFLERHTSC sequence contains several stretches of hydrophobic amino acids, which could facilitate membrane association. b. The protein contains a potential transmembrane domain (TMD) around residues 150-170 (MEDEAVLDRGASFLKHVCDEEEVEGHHTIYIGVHVPKSYRRRRRHKRKTGHREKKEKERISENYSDKSDVENADESSSILKPLISPAAERIRFILGEEDDSPAPPQLFTELDELLAVDGQEMEWKETARWIKFEEKVEQGGERWSKPHVATLSLHSLFELRTCMEKGSIMLDREASSLPQLVEMIVDHQIETGLLKPDLKDKVTYTLLRKHRHQTKKSNLRSLADIGKTVSSASRMFTNPDNGSPAMTHRNLTSSSLNDISDKPEKDQLKNKFMKKLPRDAEASNVLVGEVDFLDSPFIAFVRLQQAVMLGALTEVPVPTRFLFILLGPKGKAKSYHEIGRAIATLMSDEVFHDIAYKAKDRQDLIAGIDEFLDEVIVLPPGEWDPAIRIEPPKSLPSSDKRKNMYSGGENVQMNGDTPPDGGHGGGHADCEELQRTGRFCGGLIKDIKRKAPFFASDFYDALNIQALSAILFIYLATVTNAITFGGLLGDATDNMQGVLESFLGTAVSGAIFCLFAGQPLTILSSTGPVLVFERLLFNFSKDHNFDYLEFRLWIGLWSAFLCLILVATDASFLVQYFTRFTEEGFSSLISFIFIYDAFKKMIKLADYYPINSNFKVGYNTQFSCVCMPPDPVNISVSNDTTLAPEDLPTISSSNMYHNATFDWAFLTTKECLKYGGKLVGNNCGFVPDITLMSFILFLGTYTSSMALKKFKTSPYFPTTARKLISDFAIILPILIFCVIDALVGVDTPKLIVPSEFKPTSPNRGWFVAPFGGNPWWVYLAAAIPALLVTILIFMDQQITAVIVNRKEHKLKKGAGYHLDLFWVAILMVVCSFMALPWYVAATVISIAHIDSLKMETETSAPGEQPKFLGVREQRVTGTLVFILTGLSVFMAPILKFIPMPVLYGVFLYMGVASLNGVQFMDRLKLLLMPLKHQPDFIYLRHVPLRRVHLFTFLQVLCLALLWILKSTVAAIIFPVMILALVAVRKGMDYLFSQHDLSFLDDVIPEKDKKKEDEKKKKKKGSVDSDNDDSDCPYSEKVPSIKIPMDIMEQQPFLSDSKPSDRERSPTFLERHTSC, which is a characteristic feature of transmembrane proteins. c. The protein contains several potential phosphorylation sites, including several serine and threonine residues, which could be involved in regulating protein localization or activity. Based on these features, I predict that the protein is likely
Claude 3.5 Sonnet	To forecast the intracellular localization of this protein in eukaryotic cells, I'll analyze its amino acid composition and sequence motifs. Here's a step-by-step analysis:1. Sequence length: The protein is quite long, with over 1000 amino acids, suggesting it's a complex, multi-domain protein. 2. Charged residues: There are many charged residues (K, R, D, E) throughout the sequence, which is common in water-soluble proteins. 3. Hydrophobic regions: There are several stretches of hydrophobic amino acids, particularly in the middle and latter parts of the sequence, suggesting possible transmembrane domains. 4. N-terminal analysis: - No clear signal peptide at the very beginning - Presence of many charged residues (D, E, R, K) in the first 50 amino acids 5. Nuclear localization signals (NLS): - Strong NLS-like motif: RRRRRHKRKT near the N-terminus - Several lysine-rich regions throughout the sequence 6. Transmembrane domain predictions: - Multiple hydrophobic stretches that could form transmembrane helices 7. Specific motifs: - “KKKKK” repeat near the C-terminus, which could be a nuclear retention signal - “DSDNDDSD” sequence near the C-terminus, potentially a phosphorylation site or protein interaction motif 8. Overall composition: - Rich in glycine (G) and proline (P) in some regions, suggesting flexible linkers - Presence of cysteine (C) residues that could form disulfide bonds Based on this analysis, I predict that this protein could have multiple localizations: 1. Primary localization: Nuclear Reasoning: Strong NLS-like motif near the N-terminus, lysine-rich regions, and the “KKKKK” repeat near the C-terminus are strong indicators of nuclear localization. 2. Secondary localization: Membrane-associated (possibly nuclear membrane or endoplasmic reticulum) Reasoning: The presence of multiple hydrophobic regions suggests transmembrane domains, which could anchor the protein in a membrane. 3. Potential for nucleocytoplasmic shuttling Reasoning: The combination of nuclear localization signals and membrane-association features suggests this protein might shuttle between the nucleus and the cytoplasm, possibly in response to cellular signals.This protein appears to be a complex, multi-domain protein that may play a role in nuclear functions, possibly including transcription regulation or nuclear envelope interactions. Its potential membrane association suggests it might be involved in signaling pathways or transport processes between the nucleus and cytoplasm. To confirm this prediction, experimental verification would be necessary, such as fluorescent tagging and microscopy, or subcellular fractionation followed by Western blotting.

(c) Task 3: Subcellular localization prediction

Task 4: Keywords prediction

Instruction

What are the UniProtKB keywords for this specific protein sequence?

Input

MRGSFFSRLPPQLSLLLLLLLLLSWRRVWTQEHIGTDPSKSPVAPVCPEACSCSPGGKANCSALALPAVPAGLSWQVRSLLLDRNRVSTLPPGAFADAGALLYLVLRENRLRSVHARAFWGLGVLQRLDLSSN QLETLSPGTFTPLRALSFLSLAGNRLALLEPSILGPLPLLRVLSLQDNSLSALEAGLLNSLPALDVLRLHGNPWACSCALRPLCTWLRKHPRPTSETETLLCVSPKLQTLNLLTDFPDNAFKQCTQSLAARDLAV VYALGPASFLASLAICLALGSVLTACGARRRRRRTTVRHLIRRQPDPEGPASLEDVGSPTTTAIQA

Ground Truth

Cell membrane; Cytoplasm; Cytoskeleton; Disulfide bond; Ion channel; Ion transport; Leucine-rich repeat; Membrane; Reference proteome; Repeat; Signal; Transmembrane; Transmembrane helix; Transport

OPI-tuned model

Cell membrane; Cytoplasm; Cytoskeleton; Disulfide bond; Ion channel; Ion transport; Leucine-rich repeat; Membrane; Reference proteome; Repeat; Signal; Transmembrane; Transmembrane helix; Transport

GPT-4o

To determine the UniProtKB keywords for this specific protein sequence, you would typically need to identify the protein based on its sequence. The UniProtKB keywords are standard terms used in the UniProt database to describe specific features, functions, and characteristics of a protein.
However, without direct access to the UniProt database or specific annotation tools, it isn't possible to definitively identify the UniProtKB keywords for this sequence directly from the sequence alone.
If you have access to a sequence alignment tool or a database search tool (like BLAST), you could input the sequence to find the closest matching protein in the UniProt database. Once you have identified the corresponding protein entry, you could then check the UniProtKB keywords associated with that entry.
Would you like guidance on how to use these tools to find this information?

Llama-3.1-8B-Instruct

I can help you with that! The UniProtKB keywords for this specific protein sequence are: * Cell membrane * Cytoplasm * Lipid anchor * Lipid-binding * Lipid metabolism * Membrane protein * Phospholipid * Protein binding * Protein modification * Protein processing * Signal peptide * Transmembrane region Please note that UniProtKB keywords are not always exhaustive and may not cover all possible functions or features of a protein.

Claude 3.5 Sonnet

To provide accurate UniProtKB keywords for this specific protein sequence, I would need to search a protein database, which I don't have direct access to. However, I can analyze the sequence and provide some general observations that might hint at potential keywords:
1. The sequence starts with a methionine (M), which is typical for the start of a protein.
2. There's a high concentration of leucine (L) near the N-terminus, which could suggest a signal peptide or transmembrane region.
3. The sequence contains several repeating patterns, particularly of leucine (L) and serine (S), which might indicate leucine-rich repeat (LRR) domains.
4. There are multiple cysteines (C) present, which could be involved in disulfide bond formation.
5. The C-terminal end has a stretch of hydrophobic residues, which might suggest a transmembrane region.
6. The sequence ends with AIQA, which doesn't appear to be a common motif but could be functionally significant.
Based on these observations, some potential UniProtKB keywords might include:
- Signal peptide
- Transmembrane
- Leucine-rich repeat
- Disulfide bond
- Membrane protein
However, to get the actual UniProtKB keywords, you would need to search this sequence in the UniProtKB database or use tools like BLAST to find similar proteins with known annotations. Would you like me to explain any of these observations in more detail?

(d) Task 4: Keywords prediction

Task 5: GO terms prediction

Instruction

The Gene Ontology project (GO) provides a controlled vocabulary to describe gene and gene product attributes in any organism. There are 3 disjoint categories: cellular component, molecular function and biological process. Predict the GO term for a given protein sequence.

Input

MEFVTNYTLEELKKRFTELGLEPYRAKQVFRWVYKKFVTDFEKMTDLGKKHRELLKEHFAFHPLEKLDRVEAPDAVKYLFKTKDGHILETVLIKERDHYTLCVSSQIGCAVGCTFCATALDGLKRNLSTAEIIDQYLQVQQDLGEEKIRN VVFMGMGEPLANYENVRKAVEIMVSPEGLDLSKRRITISTSGIVAQIKRMAQDPVMKEVNLAVSLNAVSQKKREELMPLTKTNTLEELMEVLKNYPLPKYRRITLEYVLIKGVNDSPNDAERLAK LIGRHKKKFKVNLIPFNPDPNLPYER PALTDIMKFQKVLWKYGISNFVRFSKGVEVFGACGQLRTQRLQLQRV

Ground Truth

cytoplasm ; 4 iron, 4 sulfur cluster binding ; metal ion binding ; rRNA (adenine-C2-)-methyltransferase activity ; rRNA binding ; tRNA (adenine-C2-)-methyltransferase activity ; tRNA binding ; rRNA base methylation ; tRNA methylation

OPI-tuned model

cytoplasm ; 4 iron, 4 sulfur cluster binding ; metal ion binding ; rRNA (adenine-C2-)-methyltransferase activity ; rRNA binding ; tRNA (adenine-C2-)-methyltransferase activity ; tRNA binding ; rRNA base methylation

GPT-4o

To predict the Gene Ontology (GO) term for a given protein sequence, one approach is to use machine learning models trained on a dataset of protein sequences with known GO annotations. The process generally involves feature extraction from the protein sequence, model training, and prediction.
Here's a high-level approach:
1. **Feature Extraction**: - Convert the protein sequence into a numerical representation that can be used as input for a machine learning model. Common methods include: - **Amino acid composition**: Count the frequency of each amino acid in the sequence. - **K-mer frequency**: Count the occurrence of subsequences of length k.- **Position-specific scoring matrix (PSSM)**: Represent evolutionary information from multiple sequence alignments. - **Physicochemical properties**: Encode properties like hydrophobicity, charge, and molecular weight.
2. **Model Selection**: - Use a classifier such as Random Forest, Support Vector Machine, or deep learning models like Convolutional Neural Networks (CNN) or Recurrent Neural Networks (RNN). - For deep learning models, techniques like one-hot encoding or embedding layers can be applied directly to the protein sequence.
3. **Training**: - Train the model on a labeled dataset where protein sequences are associated with known GO terms.- The dataset could be sourced from databases like UniProt, where sequences are annotated with GO terms.
4. **Prediction**: - Once the model is trained, input the new protein sequence (like the one provided) into the model to predict the most likely GO term(s).
5. **Post-Processing**: - Apply thresholds or confidence scoring to refine the predictions. - Consider hierarchical relationships in GO terms to make more accurate predictions.
For the specific sequence provided:
```MEFVTNYTLEELKKRFTELGLEPYRAKQVFRWVYKKFVTDFEKMTDLGKKHRELLKEHFAFHPLEKLDRVEAPDAVKYLFKTKDGHILETVLIKERDHYTLCVSSQIGCAVGCTFCATALDGLKRNLSTAEIIDQYLQVQQDLGEEKIRNVVFMGMGEPLANYENVRKAVEIMVSPEGLDLSKRRIT ISTSGIVAQIKRMAQDPVMKEVNLAVSLNAVSQKKREELMPLTKTNTLEELMEVLKNYPLPKYRRITLEYVLIKGVNDSPNDAERLAKLIGRHKKKFKVNLIPFNPDPNLPYERPALTDIMKFQKVLWKYGISNFVRFSKGVEVFGACGQLRTQRLQLQRV```
If a model is already in place, you would input this sequence and receive predictions for its GO terms across the three categories (cellular component, molecular function, biological process).
Would you like to see a code example for predicting GO terms, or any further specific details?

Llama-3.1-8B-Instruct

To predict the GO term for a given protein sequence, I'll use a combination of bioinformatics tools and algorithms. Since you didn't provide any specific information about the protein sequence, I'll assume it's a novel protein and try to identify its functional categories using various databases and tools. After analyzing the protein sequence using various tools, including BLAST, Pfam, and InterProScan, I've identified some potential functional domains and motifs. Here are my predictions for the GO terms: 1. Cellular Component: * Based on the presence of a transmembrane domain (TM) and a signal peptide, I predict that this protein is likely to be a membrane protein, possibly a receptor or a transporter. Therefore, I assign the GO term "cell membrane" (GO:0005886) as a potential cellular component. 2. Molecular Function: * The presence of a catalytic domain (Pfam: PF00067) suggests that this protein may have enzymatic activity. I predict that it may be involved in catalyzing a chemical reaction, possibly a hydrolysis reaction. Therefore, I assign the GO term "catalytic activity" (GO:0003824) as a potential molecular function. 3. Biological Process: * The presence of a signal peptide and a transmembrane domain suggests that this protein may be involved in cellular signaling or transport processes. Additionally, the presence of a catalytic domain suggests that it may be involved in metabolic processes. Therefore, I assign the GO term "cellular process" (GO:0009987) as a potential biological process. Please note that these predictions are based on computational analysis and may require experimental validation to confirm the protein's function and GO terms.

Claude 3.5 Sonnet

To predict the Gene Ontology (GO) term for this protein sequence, I would need to use specialized bioinformatics tools and databases, which I don't have direct access to. However, I can provide you with a general approach to predict GO terms for a given protein sequence and some insights based on the sequence you provided. General approach to predict GO terms: 1. Sequence similarity search: Use tools like BLAST to find similar proteins with known GO annotations. 2. Protein domain analysis: Identify conserved domains using tools like InterProScan or PFAM. 3. Machine learning methods: Use pre-trained models that can predict GO terms based on sequence features. 4. Literature mining: Search for publications mentioning this or similar proteins. 5. Integrative approaches: Combine multiple methods for more accurate predictions. Observations about the given sequence: 1. Length: The sequence is approximately 420 amino acids long, which is a moderate size for a protein. 2. Composition: It contains a mix of hydrophobic (e.g., L, V, I, F) and hydrophilic (e.g., K, R, E, D) amino acids, suggesting it might have both globular and membrane-interacting regions. 3. Potential functional motifs: - "GEPLANY" might be part of a binding site or active site. - "MSPEG" could be involved in protein-protein interactions. - "KKREELMP" might be a nuclear localization signal due to the basic residues. 4. Recurring patterns: There are several lysine (K) and arginine (R) rich regions, which might indicate DNA/RNA binding or nuclear localization. Based on these observations, some possible GO term predictions could be: 1. Cellular Component: - GO:0005634 (nucleus) - due to potential nuclear localization signals - GO:0005737 (cytoplasm) - as many proteins are present here 2. Molecular Function: - GO:0003677 (DNA binding) - due to the presence of basic amino acid clusters - GO:0005515 (protein binding) - common function for many proteins 3. Biological Process: - GO:0006355 (regulation of transcription, DNA-templated) - if it's a nuclear protein involved in DNA binding - GO:0016032 (viral process) - some viral proteins have similar characteristics Please note that these are speculative predictions based on limited information. For accurate GO term prediction, you would need to use specialized bioinformatics tools and databases, and ideally have experimental data to support the predictions.

(e) Task 5: GO terms prediction

Task 6: Function prediction

Instruction

Generate the functional description in free-text form based on the protein sequence.

Input

MAAPSGVHLLVRRGSHRIFSSPLNHIYLHKQSSSQQPRRNFFFRRQRDISHSIVLPAAVSSAHPVPKHIKKDPDYVTTGIVPDWGDSIEVKNEDQIQGLHQACQLARHVLLLAGKSLKVDMTTEEIDALVHREIISHNAYPSPLGYGGFPKSVCT SVNNVLCHGIPDSRPLQDGDIINIDVTVYYNGYHGDTSETFLVGNVDEGGKKLVEVARRCRDEAIAACRAGAPFSVIGNTISHITHQNGFQVCPHFVGHGIGSYFHGHPEIWHHANDSDLPMEEG MAFTIEPIITEGSPEFKVLEDAWTVVS LDNQRSAQFEHTVLITSRGAQILTKLPHEA

Ground Truth

Removes the N-terminal methionine from nascent proteins. The N-terminal methionine is often cleaved when the second residue in the primary sequence is small and uncharged (Met-Ala-, Cys, Gly, Pro, Ser, Thr, or Val). Requires deformylation of the N(alpha)-formylated initiator methionine before it can be hydrolyzed (By similarity). May play a role in colon tumorigenesis.

OPI-tuned model

Removes the N-terminal methionine from nascent proteins. The N-terminal methionine is often cleaved when the second residue in the primary sequence is small and uncharged (Met-Ala-, Cys, Gly, Pro, Ser, Thr, or Val). Requires deformylation of the N(alpha)-formylated initiator methionine before it can be hydrolyzed.

GPT-4o

Based on the provided protein sequence, the following functional description can be inferred:
Functional Description:
This protein appears to have several key functional domains that suggest a role in various cellular processes. The presence of multiple arginine-rich regions (RR) and lysine residues (K) indicates potential roles in nucleic acid binding, possibly involved in transcription regulation or RNA processing. The sequence also features glycine-rich regions (G) which are often associated with flexible linkers or regions that may facilitate interactions with other proteins or nucleic acids.
The presence of serine (S) and threonine (T) residues suggests potential sites for phosphorylation, implying regulation by kinases and a role in signal transduction pathways. The conserved regions with histidine (H) and cysteine (C) residues could indicate metal ion binding capabilities, possibly zinc fingers or other metal-binding motifs, which are crucial for DNA-binding and structural stability.
The sequence also contains stretches of hydrophobic amino acids (V, L, I), likely forming transmembrane domains or contributing to protein-protein interactions within the cellular membrane or other hydrophobic environments. This could suggest a role in membrane-associated processes or signal transduction across membranes.
Given the sequence, the protein might be involved in diverse functions such as gene regulation, signal transduction, and possibly interactions with membranes or other proteins, highlighting its multifaceted role in cellular biology.

Llama-3.1-8B-Instruct

Here is the functional description in free-text form based on the protein sequence: This protein is a serine protease that plays a crucial role in the blood coagulation cascade. It is a key enzyme in the formation of blood clots and is involved in the conversion of fibrinogen to fibrin. The protein has a serine active site and is activated by thrombin, which is itself an enzyme that is produced during the coagulation process. The protein is composed of multiple domains, including a serine protease domain, a thrombin-binding domain, and a fibrinogen-binding domain. The serine protease domain is responsible for the proteolytic activity of the protein, while the thrombin-binding domain allows the protein to interact with thrombin and be activated. The fibrinogen-binding domain allows the protein to bind to fibrinogen and convert it to fibrin. The protein is also involved in the regulation of platelet aggregation and the formation of blood clots. It is able to bind to platelets and activate them, leading to the formation of platelet aggregates and the release of pro-coagulant factors. Additionally, the protein is able to bind to von Willebrand factor, a protein that plays a key role in platelet adhesion and aggregation. Overall, this protein plays a critical role in the blood coagulation cascade and is essential for the formation of blood clots. Its dysregulation can lead to bleeding disorders or thrombotic events, such as deep vein thrombosis or pulmonary embolism.

Claude 3.5 Sonnet

Based on the provided protein sequence, I can generate a functional description in free-text form. Please note that this description is based on analysis of the sequence and potential structural motifs, without access to experimental data or specific literature about this exact protein. Here's a possible functional description:
This protein appears to be a multi-domain enzyme, likely involved in metabolic processes. The sequence suggests several key features:
1. N-terminal region: The protein starts with a potential signal peptide or targeting sequence (MAAPSGVHLLVRR...), which might direct it to a specific cellular compartment.
2. Regulatory domain: The presence of serine-rich regions (e.g., SSSQQ) suggests possible phosphorylation sites, indicating that the protein's activity might be regulated by kinases.
3. Catalytic core: The central portion of the protein contains sequences typical of enzymes, particularly oxidoreductases or dehydrogenases. The presence of motifs like GIVPDWGD and VLLLAG suggests potential NAD(P)H binding sites, which are common in these enzyme classes.
4. Substrate binding: Regions rich in hydrophobic amino acids (e.g., ILCHGIPDSRPL) may form binding pockets for organic substrates.
5. Metal binding: The sequence HVLLLAGK could potentially be part of a metal-binding site, suggesting that the protein might require a metal cofactor for its activity.
6. C-terminal domain: The C-terminus (ITSRGAQILTKLPHEA) might be involved in protein-protein interactions or provide additional regulatory functions.
Given these features, this protein is likely an oxidoreductase or dehydrogenase involved in cellular metabolism. It may catalyze the oxidation or reduction of specific substrates, possibly playing a role in pathways such as amino acid metabolism, fatty acid oxidation, or carbohydrate processing. The protein's activity is probably regulated through phosphorylation and potentially by cellular localization. It may interact with other proteins as part of a larger metabolic complex.
Further experimental characterization would be necessary to confirm its exact function, substrates, and physiological role within the cell.

(f) Task 6: Function prediction
```

Task 7: Tissue Location Prediction from Gene Symbol

Instruction	**According to the relevant data reported in the scientific literature, the Gene symbol of a given protein can be used to predict which tissues the protein is expressed in.**
Input	TMEM198
Ground Truth	**adrenal gland, appendix, breast, bronchus, cerebral cortex, colon, fallopian tube, gallbladder, kidney, nasopharynx, prostate, rectum, salivary gland, seminal vesicle, skeletal muscle, small intestine, stomach, urinary bladder**
OPI-tuned model	**adrenal gland, appendix, bone marrow, breast, bronchus, caudate, cerebellum, cerebral cortex, cervix, colon, duodenum, endometrium, epididymis, esophagus, fallopian tube, gallbladder, heart muscle, hippocampus, kidney, liver, lung, lymph node, nasopharynx, oral mucosa, ovary, pancreas, parathyroid gland, placenta, prostate, rectum, salivary gland, seminal vesicle, skeletal muscle, skin, small intestine, smooth muscle, soft tissue, spleen, stomach, testis, thyroid gland, tonsil, urinary bladder**
GPT-4o	The gene TMEM198 encodes a transmembrane protein that has been studied for its potential role in various tissues. According to scientific literature, TMEM198 is associated with specific tissue expression patterns, which can provide insights into its biological function. While the detailed tissue expression data for TMEM198 may vary depending on the study and the methodology used, general databases like GTEx (Genotype-Tissue Expression) can be used to predict and analyze the tissue-specific expression of TMEM198. Typically, proteins encoded by TMEM198 may show differential expression in tissues like the brain, liver, or other organs depending on the biological context. To obtain precise and up-to-date tissue expression profiles, it is recommended to consult resources like GTEx, BioGPS, or relevant scientific publications that provide experimental data on TMEM198's expression across various tissues.
Llama-3.1-8B-Instruct	According to the relevant data reported in the scientific literature, TMEM198 is a protein that is expressed in the following tissues: * Brain * Kidney * Liver * Lung * Spleen * Testis Please note that the expression patterns of proteins can vary across different species, and this information may not be applicable to all organisms. Additionally, the expression patterns of a protein can also vary depending on the specific conditions or disease states.
Claude 3.5 Sonnet	To accurately predict protein expression patterns in tissues based on gene symbols, we would need to consult scientific databases and recent literature. However, I can provide some general information about TMEM198 and how one might go about investigating its tissue expression: TMEM198 stands for "Transmembrane Protein 198". As the name suggests, it's a protein that spans cellular membranes. To predict which tissues express TMEM198, researchers would typically use resources such as: 1. Gene expression databases like GTEx (Genotype-Tissue Expression project) 2. Protein atlases like the Human Protein Atlas 3. Literature searches in scientific databases for studies that have examined TMEM198 expression Without access to these resources in real-time, I can't provide definitive information about TMEM198's tissue-specific expression. However, transmembrane proteins are often found in various tissues throughout the body, as they play crucial roles in cellular communication, transport, and signaling. To get accurate, up-to-date information on TMEM198's tissue expression, I'd recommend: 1. Checking the GTEx portal for RNA-seq data across different tissues 2. Examining the Human Protein Atlas for protein-level data 3. Conducting a literature search for any studies that have specifically investigated TMEM198 expression patterns Would you like me to explain how you might go about using these resources to investigate TMEM198's expression, or do you have any other questions about this topic?

(g) Task 7: Tissue location prediction from gene symbol (gSymbol2Tissue)

Task 8: Cancer Prediction from Gene Symbol

Instruction	Return the name of the cancer corresponding to the given gene symbol.
Input	FOXL2
Ground Truth	**granulosa-cell tumour of the ovary**
OPI-tuned model	**granulosa-cell tumour of the ovary**
GPT-4o	The gene symbol **FOXL2** is associated with **granulosa cell tumor** (GCT), which is a type of ovarian cancer. Mutations in the FOXL2 gene are commonly found in adult granulosa cell tumors of the ovary.
Llama-3.1-8B-Instruct	The cancer corresponding to the gene symbol FOXL2 is Ovarian Cancer.
Claude 3.5 Sonnet	The cancer most commonly associated with the FOXL2 gene is: Adult-type granulosa cell tumor This is a rare type of ovarian cancer that develops from the granulosa cells in the ovary. Mutations in the FOXL2 gene are found in over 95% of adult-type granulosa cell tumors, making it a key genetic marker for this cancer type. FOXL2 mutations are considered a defining molecular feature of adult-type granulosa cell tumors. However, it's important to note that while FOXL2 mutations are strongly associated with this cancer, genetics is complex and a single gene mutation does not always directly cause cancer. Would you like me to provide more information about the role of FOXL2 in this type of ovarian cancer or explain any other aspects of this gene-cancer association?

(h) Task 8: Cancer prediction from gene symbol (gSymbol2Cancer)

Task 9: Cancer Prediction from Gene Name

Instruction	Give back the cancer name that is associated with the provided gene name.
Input	immunoglobulin lambda locus
Ground Truth	**Burkitt lymphoma**
OPI-tuned model	**Burkitt lymphoma**
GPT-4o	The gene name "immunoglobulin lambda locus" (IGL) is associated with certain types of cancer, particularly: Multiple Myeloma: A type of blood cancer that affects plasma cells, which are responsible for producing antibodies (including immunoglobulins like the lambda chain). Lymphomas: Particularly non-Hodgkin lymphomas, where abnormal B cells may produce an excessive amount of immunoglobulin lambda light chains. These cancers involve the immune system and are characterized by abnormal proliferation of cells producing the immunoglobulin lambda chain.
Llama-3.1-8B-Instruct	The cancer associated with the immunoglobulin lambda locus is Waldenström macroglobulinemia.
Claude 3.5 Sonnet	The gene name "immunoglobulin lambda locus" is not specifically associated with one particular type of cancer. Instead, this gene locus is related to the production of antibodies and is important for the immune system. However, alterations or abnormalities in the immunoglobulin lambda locus can be associated with certain blood cancers, particularly: 1. Multiple Myeloma: A cancer of plasma cells, which are a type of white blood cell that produces antibodies. 2. Some types of Non-Hodgkin Lymphoma: Particularly those affecting B-cells, such as: - Lymphoplasmacytic lymphoma - Waldenstrom macroglobulinemia 3. Chronic Lymphocytic Leukemia (CLL): In some cases, CLL can involve abnormalities in the immunoglobulin genes. It's important to note that while alterations in this gene locus can be associated with these cancers, the presence of such alterations doesn't necessarily mean cancer is present, and conversely, these cancers can occur without specific alterations to this locus. If you need more specific information about the relationship between this gene and cancer, I'd recommend consulting recent medical literature or speaking with a genetics professional or oncologist.

(i) Task 9: Cancer prediction from gene name (gName2Cancer)

Figure 7: **Prediction comparison of OPI-tuned and state-of-the-art models.**