EvoLlama: Enhancing LLMs' Understanding of Proteins via Multimodal Structure and Sequence Representations

Anonymous ACL submission

Abstract

Current Large Language Models (LLMs) for understanding proteins primarily treats amino acid sequences as a text modality. Meanwhile, Protein Language Models (PLMs), such as ESM-2, have learned massive sequential evolutionary knowledge from the universe of natural protein sequences. Furthermore, structure-based encoders like ProteinMPNN learn the structural information of proteins through Graph Neural Networks. However, whether the incorporation of protein encoders can enhance the protein understanding of LLMs has not been explored. To bridge this gap, we propose EvoLlama, a multimodal framework that connects a structure-based encoder, a sequence-based protein encoder and an LLM for protein understanding. EvoLlama consists of a ProteinMPNN structure encoder, an ESM-2 protein sequence encoder, a multimodal projector to align protein and text representations and a Llama-3 text decoder. To train EvoLlama, we fine-tune it on protein-oriented instructions and protein property prediction datasets verbalized via natural language instruction templates. Our experiments show that EvoLlama's protein understanding capabilities have been significantly enhanced, outperforming other fine-tuned protein-oriented LLMs in zero-shot settings by an average of 1%-8% and surpassing the state-of-the-art baseline with supervised fine-tuning by an average of 6%. On protein property prediction datasets, our approach achieves promising results that are competitive with state-of-the-art task-specific baselines. We will release our code in a future version.

1 Introduction

The rapid advancements in Natural Language Processing (NLP) have led to the development of Large Language Models (LLMs) that are capable of understanding and generating human language. These models such as GPT-3.5 (OpenAI, 2022), GPT-4 (Achiam et al., 2023) and Llama (Touvron et al., 2023a,b; Dubey et al., 2024), inherently possess a certain level of world knowledge and have demonstrated remarkable proficiency across a wide range of tasks. Recently, the field of Bioinformatics has seen the emergence of Transformer-based (Vaswani et al., 2017) Protein Language Models (PLMs) like ProtBert (Elnaggar et al., 2021) and ESM (Rives et al., 2021; Lin et al., 2022). These sequence-based encoders are pre-trained on a large number of amino acid sequences to capture the functional information embedded within proteins. Moreover, structure-based encoders like ProteinMPNN (Dauparas et al., 2022) and GearNet (Zhang et al., 2022b) utilize Graph Neural Networks to learn the structural information of proteins.

Despite the success of protein encoders and LLMs in their respective domains, a significant gap remains in integrating the knowledge from protein encoders into LLMs to address biological problems by leveraging the parametric knowledge of LLMs. Current LLMs treat amino acid sequences as a text modality (Pei et al., 2023; Fang et al., 2023), potentially failing to leverage the rich structural and sequential information of proteins that protein encoders are designed to capture. Moreover, protein encoders face challenges in multi-task learning, making them unable to follow human instructions. Besides, the gap between protein encoders and LLMs leads to significant challenges in aligning different modalities, even between the primary and tertiary structures of proteins (Zhang et al., 2023).

To address the aforementioned challenges, we introduce EvoLlama, a multimodal framework designed to integrate the capabilities of protein encoders with an LLM. EvoLlama combines the ESM-2 (Lin et al., 2022) protein sequence encoder, which captures sequential evolutionary knowledge from amino acid sequences, the ProteinMPNN (Dauparas et al., 2022) structure encoder that learns geometric features from 3D protein structures, a

multimodal projector that aligns protein and text representations, and a Llama-3 (Dubey et al., 2024) text decoder for generating natural language outputs.

We propose a two-stage training approach, and the experimental results demonstrate that EvoLLAMA with zero-shot outperforms the baselines fine-tuned on the Mol-Instructions (Fang et al., 2023) dataset by an average of 1%-8% and surpasses the current state-of-the-art model with supervised fine-tuning by an average of 6%. Additionally, on protein property prediction tasks based on the PEER benchmark (Xu et al., 2022), EvoLLAMA shows promising results that are competitive with task-specific baselines.

Our contributions are listed as follows:

- **Leverage multimodal representations of protein structures and sequences.** We align protein structure and sequence representations with LLM text modalities, bridging the gap in limitations of protein encoders that are unable to directly exploit the advanced capabilities of LLMs. Our approach enhances LLMs' understanding of proteins, leveraging their parametric knowledge to address biological problems and laying a foundation for future research on incorporating a broader range of biomolecular modalities.

- **Multi-task learning and instruction following capability.** We implement a two-stage training approach. After projection tuning, EvoLLAMA can follow various human instructions and solve downstream tasks in zero-shot settings. During the supervised fine-tuning stage, experiments demonstrate that tasks in the PEER benchmark have few interrelations and do not negatively affect each other when multi-task fine-tuning is employed.

- **Plug-and-play architecture and efficient fine-tuning approach.** Different protein encoders and LLMs can be used in our plug-and-play architecture. Extensive experiments demonstrate that the projection tuning stage can be optional, with the frozen LLM parameters during supervised fine-tuning significantly reducing trainable parameters. Additionally, we introduce a simple yet effective fusion method to align structure and sequence representations, improving efficiency during both training and inference.

2 Related Work

Protein-oriented LLMs BioT5 (Pei et al., 2023) and BioT5+ (Pei et al., 2024) captures the underlying relations and properties of bio-entities such as molecules and proteins. ProLLaMA (Lv et al., 2024) introduces a training framework to transform a LLM into a multi-task protein LLM, focusing on protein sequence generation and superfamily prediction tasks. InstructProtein (Wang et al., 2024b) utilizes a knowledge graph based-generation framework to construct instructions. These methods utilize text-format protein sequences while EvoLLAMA focuses on leveraging multimodal representations of proteins. Prot2Text (Abdine et al., 2024) directly fuses the structure and sequence representations as inputs into the multi-head cross-attention module within the Transformer decoder. Compared to Prot2Text, EvoLLAMA maps the structure and sequence features into language embedding tokens, enabling it to handle PPI prediction tasks, which typically requires two proteins as inputs. Additionally, Prot2Text is designed to generate protein descriptions rather than handle various protein-oriented tasks, whereas EvoLLAMA can follow human instructions, even in zero-shot settings. Further related work is discussed in Appendix A.

Protein Representations BERT-based PLMs such as ProtBert (Elnaggar et al., 2021), ProteinBERT (Brandes et al., 2022) and ESM (Rives et al., 2021; Lin et al., 2022) learn protein sequence representations through Masked Language Modeling objective. Gligorijević et al. (2021) propose a Graph Convolutional Network to encode protein structures. Zhang et al. (2022b) present a protein graph encoder to learn protein geometric features. Apart from these sequence-based protein encoders, some work employ Graph Neural Networks to learn the geometric features of proteins. Dauparas et al. (2022) introduce a protein sequence design method based on Message Passing Neural Network with 3 encoder and 3 decoder layers. We adopt the encoder layers of ProteinMPNN as a structure-based encoder in our approach. GearNet (Zhang et al., 2022b) performs relational message passing on protein residue graphs for protein representation learning. While these methods effectively learn the protein representations through sequences or structures, they do not utilize natural language with knowledge of protein properties. Therefore, EvoLLAMA aligns protein and text representations to enhance LLM's understanding of proteins.

3 Approach

EvoLlama aims to align protein information from both pre-trained structure-based and sequence-based protein encoders with an advanced LLM. Both language and protein models are open-sourced. We target to bridge the gap between the protein encoders and LLM using MLP projection layers (Sec. 3.1), with an overview of our model displayed in Fig. 1. To achieve an effective EvoLlama, we propose a two-stage training approach (Sec. 3.2). The initial stage involves pre-training the model on a large collection of aligned protein-text pairs to acquire protein language knowledge. In the second stage, we fine-tune the model with the high-quality protein-text dataset to enhance generation reliability and usability.

3.1 Architecture

In this section, we will introduce the overall EvoLlama in three parts: the protein encoders, the projection layer and the language decoder.

Protein Encoders Given the input amino acid sequence $\mathbf{X}_{\text{seq}}$, we consider the pre-trained protein encoder ESM-2 (Lin et al., 2022), which provides the protein feature $\mathbf{Z}_{\text{seq}} = \texttt{SeqEncoder}(\mathbf{X}_{\text{seq}})$. The 3D structure of the given amino acid sequence is predicted using AlphaFold-2 (Jumper et al., 2021) or ESMFold (Lin et al., 2022). A protein structure encoder, such as ProteinMPNN and GearNet, is used to extract the feature $\mathbf{Z}_{\text{struct}} = \texttt{StructEncoder}(\mathbf{X}_{\text{struct}})$.

Projection Layer To map the outputs of the protein encoders into the same space as the text features from word embedding, we apply an MLP to convert $\mathbf{Z}_{\text{seq}}$ and $\mathbf{Z}_{\text{struct}}$ into language embedding tokens $\mathbf{H}_{\text{seq}}$ and $\mathbf{H}_{\text{struct}}$ separately, which have the same dimensionality of the word embedding space in the language model:

$$\begin{cases} \mathbf{H}_{\text{seq}} = \texttt{MLP}_{\text{seq}}(\mathbf{Z}_{\text{seq}}), \\ \mathbf{Z}_{\text{seq}} = \texttt{SeqEncoder}(\mathbf{X}_{\text{seq}}), \\ \mathbf{H}_{\text{struct}} = \texttt{MLP}_{\text{struct}}(\mathbf{Z}_{\text{struct}}), \\ \mathbf{Z}_{\text{struct}} = \texttt{StructEncoder}(\mathbf{X}_{\text{struct}}) \end{cases}$$

Furthermore, since both structure-based and sequence-based protein encoders extract features based on residue positions, the lengths of their feature representations are dependent solely on the length of the amino acid sequence. Therefore, we fuse the structure and sequence features by employing an element-wise addition of the corresponding residue features. The fused protein representations $\mathbf{H}_p = \mathbf{H}_{\text{seq}} \oplus \mathbf{H}_{\text{struct}}$ reduces the protein embedding tokens by half, significantly decreasing the training and inference latency. Note that our simple projection scheme is lightweight and cost-effective, which allows us to iterate data centric experiments quickly. We leave exploring possibly more effective and sophisticated architecture designs for EvoLlama as future work.

Language Decoder Given the protein structure $\mathbf{X}_{\text{struct}}$, amino acid sequence $\mathbf{X}_{\text{seq}}$ and the fused projected embeddings $\mathbf{H}_p$, we have conversation data $(\mathbf{X}_q, \mathbf{X}_a)$, where $\mathbf{X}_q$ and $\mathbf{X}_a$ represent the protein-related question and its corresponding answer, respectively. We organize them as a sequence and perform instruction-tuning of the LLM on the prediction tokens, using its original auto-regressive training objective. Specifically, we compute the probability of generating target answers $\mathbf{X}_a$ by:

$$\begin{aligned} &p(\mathbf{X}_a|\mathbf{X}_{\text{struct}}, \mathbf{X}_{\text{seq}}, \mathbf{X}_q) \\ &= \prod_{i=1}^{|\mathbf{X}_a|} p_\theta(\mathbf{X}_{a,i}|\mathbf{X}_{\text{struct}}, \mathbf{X}_{\text{seq}}, \mathbf{X}_q, \mathbf{X}_{a,<i}) \quad (1) \end{aligned}$$

where θ is the trainable parameters. EvoLlama model design is compatible with any off-the-shelf GPT-style pre-trained LLM. EvoLlama adopts Llama-3 8B (Dubey et al., 2024) for further training. A causal mask is applied to all the attention operations, including the attention between protein features $\mathbf{H}_p$.

3.2 Training

As illustrated in Fig. 1, the training process of the EvoLlama model consists of two stages: projection tuning and supervised fine-tuning, with the first stage being optional.

Stage 1: Projection Tuning We keep both the protein encoders and LLM weights frozen, and maximize the likelihood of Eq. 1 with the parameters of projection MLP only (Fig. 1(a)). In this way, the protein features $\mathbf{H}_p$ can be aligned with the pre-trained LLM word embedding. This stage can be understood as training a compatible protein projector for the frozen LLM.

Stage 2: Supervised Fine-Tuning To efficiently fine-tune EvoLlama and preserve the internal knowledge of the LLM, its parameters are frozen

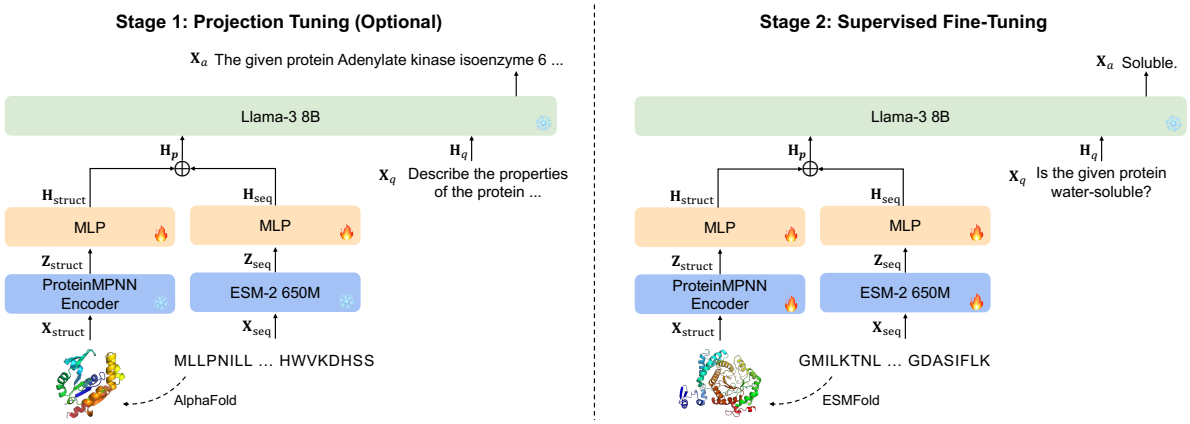


Figure 1: Overall architecture and the training pipeline of the EVOLLAMA.

during this stage. Continuing to update the pre-trained weights of the projection layers and protein encoders helps EVOLLAMA learn more protein knowledge and enhance its instruction following capability.

4 Protein Instruction-Following Data

In this section, we introduce the construction of protein instruction-following data. It consists of two sets, projection tuning and supervised fine-tuning, which are used at different training stages, described in Sec. 3. An example of the projection tuning data and supervised fine-tuning data is illustrated in Fig. 2. Formally, for each example, we define $\mathbf{X}_p, \mathbf{X}_q, \mathbf{X}_a$ as the protein (fused protein representations, consisting of the 3D structure $\mathbf{X}_{\text{struct}}$ and the amino acid sequence $\mathbf{X}_{\text{seq}}$), the protein-related natural language question, and the corresponding answer, respectively.

Projection Tuning Data It consists of protein-text pairs originated from the Swiss-Prot (Consortium, 2023) database. Due to limited computational resources, we directly utilize the 3D structures predicted by AlphaFold-2 (Jumper et al., 2021) from Swiss-Prot. The database contains 571K manually-annotated records, each containing information including protein name, subcellular location, function and families. For $\mathbf{X}_q$, we construct 10 templates that ask the model to briefly describe the input protein $\mathbf{X}_p$ from various aspects. For $\mathbf{X}_a$, information is extracted from the filtered Swiss-Prot annotation and constructed using a pre-defined template to ensure the consistency and clarity of protein descriptions. The question and answer templates are listed in Fig. 4.

Projection Tuning Data

Case #1:

Input $\mathbf{X}_{\text{struct}}$:  **Input $\mathbf{X}_{\text{seq}}$:** MLLPNILLTGTPGVGKTTLGKELASRSGLKYINVGDLAREGQLYD ...

Prompt $\mathbf{X}_q$: `<protein>` Examine the protein determined by the provided amino acid sequence, including an in-depth look at its function, where it resides within the cell, and the broader family of proteins it is a member of.

Response: The given protein Adenylate kinase isoenzyme 6 suggests its potential function as broad-specificity nucleoside monophosphate (NMP) kinase that catalyzes the reversible transfer of the terminal phosphate group between nucleoside triphosphates and monophosphates. May have a role in nuclear energy homeostasis. Has also ATPase activity. May be involved in regulation of Cajal body (CB) formation. It belongs to the adenylate kinase family. AK6 subfamily.

Supervised Fine-Tuning Data

Case #1 (PMol):

Input $\mathbf{X}_{\text{struct}}$:  **Input $\mathbf{X}_{\text{seq}}$:** MPAAVAIILAAGLGTRMRSSAPKALHPIAGRPMLGHLIAACEAVFD ...

Prompt $\mathbf{X}_q$: `<protein>` Examine the provided protein sequence and determine the catalytic activity of the enzyme it represents, focusing on the chemical reaction it promotes.

Response: Evaluation of the protein sequence indicates that the associated enzyme exhibits catalytic activity in the form of this chemical reaction: acetyl-CoA + alpha-D-glucosamine 1-phosphate = CoA + H(+) + N- acetyl-alpha-D-glucosamine 1-phosphate.

Case #2 (PEER):

Input $\mathbf{X}_{\text{struct}}$:  **Input $\mathbf{X}_{\text{seq}}$:** MPSGFQQIGSEDGEPPQQRVTGTLVLAVFSAVLGSLQFGYNIGVI ...

Input $\mathbf{X}_{\text{struct}}$:  **Input $\mathbf{X}_{\text{seq}}$:** MDELAGGGGGGPGMAAPPRQQQGPGGNLGLSPGGNGAAGGG ...

Prompt $\mathbf{X}_q$: `<protein>` `<sep>` `<protein>` Infer the interaction potential between the two human protein sequences. Answer the question using a single word or phrase.

Response: Yes.

Figure 2: An example of the projection tuning data and supervised fine-tuning data. Note that the special token `<protein>` denotes the fused protein representations of structural and sequential features.

Supervised Fine-tuning Data To align the model to follow a variety of instructions, we present and curate diverse instruction-following data about the provided proteins, by verbalizing protein-related tasks. It consists of 10 tasks including Mol-Instructions (Fang et al., 2023) and PEER benchmark (Xu et al., 2022). We use ESMFold (Lin et al., 2022) to accelerate protein structure prediction for sequences in these two datasets.

- **Mol-Instructions** is a comprehensive instruction dataset designed for the biomolecular realm. It includes three compo-

nents: molecule-oriented instructions, protein-oriented instructions and biomolecular text instructions. We adopt protein-oriented instructions in Mol-Instructions (named PMol) for the supervised fine-tuning stage. PMol details will be discussed in Sec. 5.1. For each task in Mol-Instructions, we make simple modifications to the original prompts to fit our use cases and ensure coherence. Details are discussed in Appendix B.2 and some modification cases are listed in Fig. 7.

- **PEER** is a comprehensive benchmark for general protein sequence understanding tasks including protein localization prediction, protein structure prediction and protein-protein interaction prediction. PEER benchmark details will be discussed in Sec. 5.2. For each task in PEER benchmark, there are 10 question templates and 1 answer template, some of which are listed in Fig. 6. In response templates for other tasks, categories are represented by natural language. However, for fold classification, we use integers from 0 to 1,194 due to the large number of categories.

5 Experiments

We evaluate EvoLlama[1] on downstream tasks including protein understanding tasks based on Mol-Instructions (Sec. 5.1) and protein property prediction tasks based on PEER benchmark (Sec. 5.2). Additionally, the structure encoder in our approach is replaced with GearNet to construct EvoLlama (GearNet+ESM-2) for the experiments, and further experiments on the substitution of protein sequence encoders are provided in Appendix E. We evaluate our approach in both zero-shot settings, where only the projection layers are aligned during the projection tuning stage, and in supervised fine-tuning without the projection tuning stage. Details of the experimental setups are discussed in Appendix C.

5.1 Protein Understanding Tasks

Task Descriptions Protein understanding tasks use PMol for fine-tuning and evaluation, which consist of four distinct tasks with datasets constructed based on UniProtKB (Consortium, 2021). Protein function prediction outputs the function of the given protein. Catalytic activity prediction outputs the catalytic activity of the input protein and the chemical reactions it promotes. Domain/motif prediction outputs the domains or motifs that the given protein may contain. Functional description generation outputs the description of the input protein's function, subcellular localization, and any biological processes it may be a part of.

Baselines We compare our approach with the protein-oriented LLMs in Mol-Instructions including LLaMA (Touvron et al., 2023a), Alpaca (Tloen, 2023), Baize (Xu et al., 2023a), ChatGLM (Zeng et al., 2022), Galactica (Taylor et al., 2022) and Vicuna. Apart from these LLMs, we use Prot2Text (Abdine et al., 2024) and ProLLaMA (Lv et al., 2024) as our baseline models in zero-shot settings. These models lack support for arbitrary prompts. Prot2Text is designed to generate protein descriptions, while ProLLaMA predicts protein superfamilies. Therefore, we evaluate protein function prediction for Prot2Text and domain/motif prediction for ProLLaMA. For protein understanding tasks, we follow Mol-Instructions, taking ROUGE-L (Lin, 2004) as the evaluation metric. Details of ROUGE-L implementation are discussed in Appendix D.

Results As shown in Tab. 1, EvoLlama and EvoLlama (GearNet+ESM-2) with zero-shot not only handle all protein understanding tasks but also outperform Prot2Text and ProLLaMA. Furthermore, they surpass or approach ChatGLM, Llama-2-7B-Chat and Vicuna fine-tuned on protein-oriented instructions by 1%-8%, demonstrating that during the projection tuning stage, EvoLlama and EvoLlama (GearNet+ESM-2) learn protein knowledge and can follow human instructions to generalize their knowledge for various downstream tasks. Additionally, EvoLlama outperforms all baseline models, including Llama-2-7B-Chat fine-tuned on the complete Mol-Instructions dataset (Mol) after supervised fine-tuning, highlighting the effectiveness of our approach. Notably, our approach uses a relatively small amount of data and has significantly fewer trainable parameters than the baseline models using full-parameter fine-tuning. The experimental results highlight the importance of leveraging the multimodal structure and sequence representations during training LLMs.

5.2 Protein Property Prediction Tasks

Task Descriptions Protein property prediction tasks use PEER benchmark for fine-tuning and evaluation, which consist of 6 tasks. Solubility prediction (Khurana et al., 2018), defined as a bi-

[1]Unless specified otherwise, EvoLlama refers to EvoLlama (ProteinMPNN+ESM-2).

Models	Data	Trainable/Total #Params.	ROUGE-L(↑)				
			PF	GF	CA	DP	Avg.
Models in zero-shot settings							
Galactica	-	-	0.12	0.12	0.13	0.09	0.1150
Prot2Text	-	-	0.14	-	-	-	0.1400
ProLLaMA	-	-	-	-	-	0.02	0.0200
EvoLlama (GearNet+ESM-2)	-	-	0.16	0.16	0.21	0.15	0.1700
EvoLlama (ProteinMPNN+ESM-2)	-	-	0.16	0.14	0.15	0.11	0.1400
Fine-tuned models							
ChatGLM	PMol	6B/6B	0.15	0.14	0.13	0.10	0.1300
Llama-2-7B-Chat	PMol	7B/7B	0.15	0.14	0.16	0.12	0.1425
Llama-2-7B-Chat	Mol	7B/7B	0.42	0.44	0.52	0.46	0.4600
Vicuna	PMol	7B/7B	0.07	0.08	0.08	0.06	0.0725
Alpaca	PMol	7B/7B	0.2	0.1	0.23	0.12	0.1625
Baize	PMol	7B/7B	0.2	0.15	0.22	0.13	0.1750
EvoLlama (GearNet+ESM-2)	PMol, PEER	720M/8.8B (8.2%)	*0.25*	*0.32*	*0.34*	*0.31*	*0.3050*
EvoLlama (ProteinMPNN+ESM-2)	PMol, PEER	690M/8.8B (7.9%)	**0.48**	**0.50**	**0.60**	**0.50**	**0.5200**

Table 1: Results of protein understanding tasks (**Best**, Second Best, *Third Best*). PF refers to protein function prediction. GF refers to functional description generation. CA refers to catalytic activity prediction. DP refers to domain/motif prediction. Note that Mol refers to the Mol-Instructions with 3 components: molecule-oriented instructions, protein-oriented instructions (named PMol), and biomolecular text instructions. - indicates the data is not applicable to the task.

nary classification task, aims to predict whether a given protein is soluble or not. Subcellular localization prediction (Almagro Armenteros et al., 2017), defined as a ten-class classification task, aims to predict where a given protein locates in the cell. Binary localization prediction, a simplified version of subcellular localization prediction, is defined as a binary classification task that aims to determine whether a given protein is soluble or membrane-bound. Fold classification (Fox et al., 2014; Hou et al., 2018) requires the model to classify the global structural topology of a given protein into one of 1195 classes at the fold level. Yeast PPI prediction (Guo et al., 2008) and human PPI prediction (Peri et al., 2003; Pan et al., 2010) are defined as binary localization tasks, which aim to predict whether two given yeast or human proteins interact or not respectively. It is worth noting that for all tasks, protein sequences in the training set with high similarity to those in the test set are excluded based on the sequence identity. For example, sequences with $\geq 30\%$ identity are excluded in the solubility prediction task. Therefore, a key challenge in protein property prediction tasks lies in evaluating a model's ability to generalize across dissimilar protein sequences.

Baselines We compare our approach with the following baselines in PEER benchmark. Feature engineers include Dipeptide Deviation from Expected Mean (DDE) (Saravanan and Gautham, 2015) and Moran correlation (Moran) (Feng and Zhang, 2000). Protein sequence encoders include LSTM (Hochreiter and Schmidhuber, 1997), Transformer (Vaswani et al., 2017), shallow CNN (Shanehsazzadeh et al., 2020) and ResNet (He et al., 2016). Pre-trained PLMs include ProtBert (Elnaggar et al., 2021) and ESM-1b (Rives et al., 2021). Protein-oriented LLMs include Llama-3-8B-Instruct (Dubey et al., 2024) and InstructProtein (Wang et al., 2024b). For protein property prediction tasks, we take accuracy (Acc) as the evaluation metric.

Results As displayed in Tab. 2, EvoLlama with zero-shot achieves a comparable performance with both task-specific and single models across several tasks including solubility prediction, binary localization prediction and PPI prediction tasks. EvoLlama (GearNet+ESM-2) with zero-shot performs relatively worse than EvoLlama due to its lower capability to follow instructions. Besides, the supervised fine-tuning stage significantly enhances the performance of our approach, enabling it to outperform or approach the previous state-of-the-art models including ProtBert, ESM-1b and InstructProtein across multiple tasks. Additionally, we add a linear classification head to ESM-2 and fine-tune

Models	Sol	Sub	Bin	Fold	Yst	Hum	Avg.
Task-specific models							
DDE	59.77±1.21	49.17±0.40	77.43±0.42	9.57±0.46	55.83±3.13	62.77±2.30	52.42
Moran	57.73±1.33	31.13±0.47	55.63±0.85	7.10±0.56	53.00±0.50	54.67±4.43	43.21
CNN	64.43±0.25	58.73±1.05	82.67±0.32	10.93±0.35	55.07±0.02	62.60±1.67	55.74
ResNet	67.33±1.46	52.30±3.51	78.99±4.41	8.89±1.45	48.91±1.78	68.61±3.78	54.17
LSTM	*70.18±0.63*	62.98±0.37	88.11±0.14	8.24±1.61	53.62±2.72	63.75±5.12	57.81
Transformer	70.12±0.31	56.02±0.82	75.74±0.74	8.52±0.63	54.12±1.27	59.58±2.09	54.02
ProtBert	68.15±0.92	76.53±0.93	91.32±0.89	16.94±0.42	**63.72±2.80**	77.32±1.10	65.66
ESM-1b	70.23±0.75	**78.13±0.49**	92.40±0.35	**28.17±2.05**	*57.00±6.38*	**78.17±2.91**	**67.35**
Single models in zero-shot settings							
EvoLlama (GearNet+ESM-2)	34.65±0.48	-	50.89±0.88	-	2.52±0.11	3.70±0.10	22.94
EvoLlama (ProteinMPNN+ESM-2)	50.76±0.47	8.16±0.27	*92.85±0.21*	0.65±0.06	53.59±0.36	50.21±0.79	42.70
Single fine-tuned models							
InstructProtein	69.08±0.00	70.79±0.00	85.19±0.00	10.86±0.00	-	-	58.98
Llama-3-8B-Instruct	69.13±0.39	51.36±0.06	98.91±0.00	8.77±0.40	56.05±0.53	62.87±0.34	57.85
EvoLlama (GearNet+ESM-2)	61.13±0.15	42.27±0.23	85.21±0.34	3.11±0.36	50.42±0.57	67.72±1.08	51.64
EvoLlama (ProteinMPNN+ESM-2)	**72.37±0.35**	*73.25±0.27*	**99.73±0.05**	*10.96±0.36*	57.45±0.78	*72.71±1.15*	*64.41*

Table 2: Results (in %) of protein property prediction tasks (**Best**, Second Best, *Third Best*). Sol represents solubility prediction. Sub represents subcellular localization prediction. Bin represents binary localization prediction. Fold represents fold classification. Yst represents yeast PPI prediction. Hum represents human PPI prediction. - indicates the data is not applicable to the task.

it with full parameters for 1K steps per task. ESM-2 achieves an average score of 67.10, demonstrating that ESM-2 is a competitive baseline compared to ESM-1b. Our approach surpasses ESM-2 on binary localization prediction and human PPI prediction tasks while approaching it on other tasks. We find that when jointly fine-tuning models on various tasks through multi-task learning, the performance improves compared to task-specific models, demonstrating no negative impact between tasks. However, more interrelated downstream tasks or data could be introduced in future work to further enhance the model's performance.

In particular, compared to task-specific models, EvoLlama outperforms or approaches them across several tasks, except for fold classification. A possible explanation for the promising results is that EvoLlama learns the properties of proteins through various tasks by leveraging multimodal structure and sequence representations. Notably, both EvoLlama and Llama-3-8B-Instruct achieve the best performance on binary localization task, demonstrating the significant potential of LLMs to understand proteins. Additionally, BioT5 and BioT5+ are trained to solve four tasks, excluding subcellular localization prediction and fold classification, achieving average accuracy scores of 79.36 and 79.01, respectively, across these tasks. In comparison, EvoLlama has an average score of 75.57, showing comparable performance with these two models. A possible explanation for the relatively higher accuracy of these two models is that they incorporate molecules as a new modality and leverage more molecule-related data, potentially yielding positive effects, while we focus only on protein structures and sequences. A similar result can be observed in Tab. 1, where Llama-2-7B-Chat fine-tuned on the complete Mol-Instructions dataset, including molecule-related data, performs significantly better than when fine-tuned only on protein-oriented instructions. We leave the exploration of incorporating more biomolecular modalities, such as molecules and DNA, for future work.

Additionally, compared to Llama-3-8B-Instruct, used as a text decoder in our approach, EvoLlama improves performance on all tasks by incorporating the multimodal structure and sequence representations of proteins.

5.3 Ablation Study

In this section, we conduct ablation studies to explore the effect of the projection tuning stage, structure and sequence representations, and the fusion method. For a fair comparison, the baselines are trained using the same experimental setups discussed in Appendix C with 10K steps. We conduct more evaluations on protein property prediction tasks in Appendix E, where we further discuss the effect of protein sequence encoder sizes.

Effect of the projection tuning stage We compare the performance of EVOLLAMA with and without the projection tuning stage in Tab. 3. EVOLLAMA, when continued to be fine-tuned after the projection tuning stage, achieves lower performance compared to direct supervised fine-tuning. A possible reason is that we use protein structures predicted by AlphaFold-2 during the projection tuning stage, while structures predicted by ESMFold are used during the supervised fine-tuning stage. The zero-shot ability discussed in Sec. 5.1 and Sec. 5.2 highlights the effectiveness of the projection tuning stage, demonstrating that the model learns to generalize its structural knowledge from structures predicted by AlphaFold-2 to those predicted by ESMFold during inference. However, it's challenging for models to learn the gap between protein structures predicted by AlphaFold-2 and ESMFold when the structure encoder and projection layer are updated simultaneously during the supervised fine-tuning stage.

Models	PF	GF	CA	DP	Avg.
EVOLLAMA	0.43	0.46	0.55	0.48	**0.4800**
EVOLLAMA w/ PT	0.28	0.34	0.40	0.34	0.3400

Table 3: Effect of the projection tuning stage. The experiments are conducted on protein understanding tasks. PT refers to the projection tuning stage.

Effect of structure and sequence representations As shown in Tab. 4, EVOLLAMA incorporating both structure and sequence representations outperforms those utilizing either alone, demonstrating the effectiveness of integrating both protein representations. EVOLLAMA (GearNet+ESM-2) significantly enhances the performance by leveraging structure and sequence representations. Furthermore, EVOLLAMA with only a sequence-based protein encoder surpasses the one with only a structure-based protein encoder, regardless of the protein structure encoder chosen. A possible explanation is that the features extracted by ESM-2 implicitly contain structural information, indicating that sequence representations are easier for LLMs to learn. Moreover, the parameters of the structure encoder are one to two orders of magnitude fewer than those of the sequence encoder, leading to a more limited extraction of structural features. A case study on the effect of structure and sequence representations is conducted in Appendix F.

Models	PF	GF	CA	DP	Avg.
EVOLLAMA	0.43	0.46	0.55	0.48	**0.4800**
EVOLLAMA w/o ProteinMPNN	0.39	0.46	0.54	0.48	0.4675
EVOLLAMA w/o ESM-2	0.35	0.42	0.50	0.41	0.4200
EVOLLAMA (GearNet+ESM-2)	0.30	0.32	0.36	0.30	**0.3200**
EVOLLAMA (GearNet+ESM-2) w/o ESM-2	0.20	0.28	0.31	0.29	0.2700

Table 4: Effect of structure and sequence representations. The experiments are conducted on protein understanding tasks. Note that EVOLLAMA (GearNet+ESM-2) (w/o GearNet) is equivalent to EVOLLAMA (w/o ProteinMPNN), as both only utilize the sequence representations extracted by the protein sequence encoder.

Effect of the fusion method To evaluate the effect of the fusion method, we directly use the structure and sequence features instead of fusing their representations. As shown in Tab. 5, EVOLLAMA with fused representations surpass the one without, demonstrating the effectiveness of our fusion method. However, EVOLLAMA (GearNet+ESM-2) without fused representations outperforms the one with them, indicating that for different protein structure encoders, different fusion methods may be chosen. Furthermore, the fused representations reduce the token cost of the LLM, resulting in approximately 20% lower inference latency.

Models	PF	GF	CA	DP	Avg.	Latency
EVOLLAMA	0.43	0.46	0.55	0.48	**0.4800**	×**1.0**
EVOLLAMA w/o fused representations	0.42	0.44	0.55	0.48	0.4725	×1.17
EVOLLAMA (GearNet+ESM-2)	0.30	0.32	0.36	0.30	0.3200	×**0.8**
EVOLLAMA (GearNet+ESM-2) w/o fused representations	0.31	0.34	0.39	0.36	**0.3500**	×1.0

Table 5: Effect of the fusion method. The experiments are conducted on protein understanding tasks.

6 Conclusion

In this paper, we propose EVOLLAMA, a multimodal framework that connects ProteinMPNN, ESM-2 650M and Llama-3 8B for protein understanding through a two-stage training process. Experiments demonstrate that after the projection tuning stage, EVOLLAMA in zero-shot settings outperforms the fine-tuned baselines with full parameters, surpassing the current state-of-the-art model with supervised fine-tuning on the Mol-Instructions. Additionally, our approach achieves promising results that are competitive with state-of-the-art task-specific baselines on the PEER benchmark.

Limitations

EvoLlama incorporates the structure and sequence representations of proteins to enhance LLM's understanding of proteins. Due to the lack of experimentally determined structures for many proteins in our experiments, we use 3D structures predicted by AlphaFold-2 and ESMFold to fully leverage the data. These computationally predicted structures generally have relatively lower accuracy compared to wet lab experiments. Besides, training a single model to predict various protein properties presents challenges, causing EvoLlama to only approach the state-of-the-art performance for some tasks in protein property prediction.

References

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

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

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

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

The UniProt Consortium. 2021. Uniprot: the universal protein knowledgebase in 2021. *Nucleic acids research*, 49(D1):D480–D489.

The UniProt Consortium. 2023. Uniprot: the universal protein knowledgebase in 2023. *Nucleic acids research*, 51(D1):D523–D531.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N Fung, and Steven Hoi. 2024. Instructblip: Towards general-purpose vision-language models with instruction tuning. *Advances in Neural Information Processing Systems*, 36.

Justas Dauparas, Ivan Anishchenko, Nathaniel Bennett, Hua Bai, Robert J Ragotte, Lukas F Milles, Basile IM Wicky, Alexis Courbet, Rob J de Haas, Neville Bethel, et al. 2022. Robust deep learning–based protein sequence design using proteinmpnn. *Science*, 378(6615):49–56.

Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang, and Jie Tang. 2021. Glm: General language model pretraining with autoregressive blank infilling. *arXiv preprint arXiv:2103.10360*.

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

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

Yin Fang, Xiaozhuan Liang, Ningyu Zhang, Kangwei Liu, Rui Huang, Zhuo Chen, Xiaohui Fan, and Huajun Chen. 2023. Mol-instructions: A large-scale biomolecular instruction dataset for large language models. *arXiv preprint arXiv:2306.08018*.

Zhi-Ping Feng and Chun-Ting Zhang. 2000. Prediction of membrane protein types based on the hydrophobic index of amino acids. *Journal of protein chemistry*, 19:269–275.

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

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

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2021. Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare (HEALTH)*, 3(1):1–23.

Han Guo, Mingjia Huo, Ruiyi Zhang, and Pengtao Xie. 2023. Proteinchat: Towards achieving chatgpt-like functionalities on protein 3d structures. *Authorea Preprints*.

Yanzhi Guo, Lezheng Yu, Zhining Wen, and Menglong Li. 2008. Using support vector machine combined with auto covariance to predict protein–protein interactions from protein sequences. *Nucleic acids research*, 36(9):3025–3030.

Tomas Hayes, Roshan Rao, Halil Akin, Nicholas J Sofroniew, Deniz Oktay, Zeming Lin, Robert Verkuil, Vincent Q Tran, Jonathan Deaton, Marius Wiggert, et al. 2024. Simulating 500 million years of evolution with a language model. *bioRxiv*, pages 2024–07.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

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

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

Sameer Khurana, Reda Rawi, Khalid Kunji, Gwo-Yu Chuang, Halima Bensmail, and Raghvendra Mall. 2018. Deepsol: a deep learning framework for sequence-based protein solubility prediction. *Bioinformatics*, 34(15):2605–2613.

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

Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Allan dos Santos Costa, Maryam Fazel-Zarandi, Tom Sercu, Sal Candido, et al. 2022. Language models of protein sequences at the scale of evolution enable accurate structure prediction. *BioRxiv*, 2022:500902.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2024a. Visual instruction tuning. *Advances in neural information processing systems*, 36.

Tianyu Liu, Yijia Xiao, Xiao Luo, Hua Xu, W Jim Zheng, and Hongyu Zhao. 2024b. Geneverse: A collection of open-source multimodal large language models for genomic and proteomic research. *arXiv preprint arXiv:2406.15534*.

Liuzhenghao Lv, Zongying Lin, Hao Li, Yuyang Liu, Jiaxi Cui, Calvin Yu-Chian Chen, Li Yuan, and Yonghong Tian. 2024. Prollama: A protein large language model for multi-task protein language processing. *arXiv preprint arXiv:2402.16445*.

OpenAI. 2022. Introducing chatgpt. https://openai.com/blog/chatgpt/. Accessed: 2024-03-10.

Xiao-Yong Pan, Ya-Nan Zhang, and Hong-Bin Shen. 2010. Large-scale prediction of human protein- protein interactions from amino acid sequence based on latent topic features. *Journal of proteome research*, 9(10):4992–5001.

Qizhi Pei, Lijun Wu, Kaiyuan Gao, Xiaozhuan Liang, Yin Fang, Jinhua Zhu, Shufang Xie, Tao Qin, and Rui Yan. 2024. Biot5+: Towards generalized biological understanding with iupac integration and multi-task tuning. *arXiv preprint arXiv:2402.17810*.

Qizhi Pei, Wei Zhang, Jinhua Zhu, Kehan Wu, Kaiyuan Gao, Lijun Wu, Yingce Xia, and Rui Yan. 2023. Biot5: Enriching cross-modal integration in biology with chemical knowledge and natural language associations. *arXiv preprint arXiv:2310.07276*.

Suraj Peri, J Daniel Navarro, Ramars Amanchy, Troels Z Kristiansen, Chandra Kiran Jonnalagadda, Vineeth Surendranath, Vidya Niranjan, Babylakshmi Muthusamy, TKB Gandhi, Mads Gronborg, et al. 2003. Development of human protein reference database as an initial platform for approaching systems biology in humans. *Genome research*, 13(10):2363–2371.

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

Vijayakumar Saravanan and Namasivayam Gautham. 2015. Harnessing computational biology for exact linear b-cell epitope prediction: a novel amino acid composition-based feature descriptor. *Omics: a journal of integrative biology*, 19(10):648–658.

Amir Shanehsazzadeh, David Belanger, and David Dohan. 2020. Is transfer learning necessary for protein landscape prediction? *arXiv preprint arXiv:2011.03443*.

Jin Su, Chenchen Han, Yuyang Zhou, Junjie Shan, Xibin Zhou, and Fajie Yuan. 2023. Saprot: Protein language modeling with structure-aware vocabulary. *bioRxiv*, pages 2023–10.

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

Tloen. 2023. Alpaca-lora. https://github.com/tloen/alpaca-lora.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023a. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

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

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

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

Zeyuan Wang, Qiang Zhang, Keyan Ding, Ming Qin, Xiang Zhuang, Xiaotong Li, and Huajun Chen. 2024b. Instructprotein: Aligning human and protein language via knowledge instruction. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1114–1136, Bangkok, Thailand. Association for Computational Linguistics.

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

Yijia Xiao, Edward Sun, Yiqiao Jin, Qifan Wang, and Wei Wang. 2024. Proteingpt: Multimodal llm for protein property prediction and structure understanding. *arXiv preprint arXiv:2408.11363*.

Canwen Xu, Daya Guo, Nan Duan, and Julian McAuley. 2023a. Baize: An open-source chat model with parameter-efficient tuning on self-chat data. *arXiv preprint arXiv:2304.01196*.

Minghao Xu, Xinyu Yuan, Santiago Miret, and Jian Tang. 2023b. Protst: Multi-modality learning of protein sequences and biomedical texts. In *International Conference on Machine Learning*, pages 38749–38767. PMLR.

Minghao Xu, Zuobai Zhang, Jiarui Lu, Zhaocheng Zhu, Yangtian Zhang, Ma Chang, Runcheng Liu, and Jian Tang. 2022. Peer: a comprehensive and multi-task benchmark for protein sequence understanding. *Advances in Neural Information Processing Systems*, 35:35156–35173.

Aohan Zeng, Xiao Liu, Zhengxiao Du, Zihan Wang, Hanyu Lai, Ming Ding, Zhuoyi Yang, Yifan Xu, Wendi Zheng, Xiao Xia, et al. 2022. Glm-130b: An open bilingual pre-trained model. *arXiv preprint arXiv:2210.02414*.

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

Zuobai Zhang, Chuanrui Wang, Minghao Xu, Vijil Chenthamarakshan, Aurélie Lozano, Payel Das, and Jian Tang. 2023. A systematic study of joint representation learning on protein sequences and structures. *arXiv preprint arXiv:2303.06275*.

Zuobai Zhang, Minghao Xu, Arian Jamasb, Vijil Chenthamarakshan, Aurelie Lozano, Payel Das, and Jian Tang. 2022b. Protein representation learning by geometric structure pretraining. *arXiv preprint arXiv:2203.06125*.

Le Zhuo, Zewen Chi, Minghao Xu, Heyan Huang, Heqi Zheng, Conghui He, Xian-Ling Mao, and Wentao Zhang. 2024. Protllm: An interleaved protein-language llm with protein-as-word pre-training. *arXiv preprint arXiv:2403.07920*.

A Further Related Work

Large Language Models Llama-3 (Dubey et al., 2024) is a family of pre-trained and fine-tuned open-source LLMs with sizes ranging from 8B to 70B parameters. Baize (Xu et al., 2023a) is an open source model aligned by introducing self-distillation with feedback. ChatGLM (Zeng et al., 2022) is an open bilingual language model based on General Language Model (GLM) framework (Du et al., 2021), with 6B parameters. Galactica (Taylor et al., 2022) is a large language model that trained on an extensive scientific corpus of papers and knowledge bases, capable of storing, combining and reasoning about scientific knowledge.

Vision Language Models Vision Language Models (VLMs) such as LLaVA (Liu et al., 2024a) and InstructBLIP (Dai et al., 2024) have demonstrated their capability to understand visual content. LLaVA introduces an end-to-end LMM that connects a visual encoder and a LLM for visual and language understanding. InstructBLIP proposes a instruction tuning framework towards generalized vision-language models. Geneverse (Liu et al., 2024b) is a collection of fine-tuned LLMs and VLMs for three novel tasks in genomic and proteomic research. For the protein task, Geneverse uses protein structure images with the fixed capture angle. In contrast, our approach treats amino acid sequences and 3D structures as distinct modalities. Unlike the visual information provided by protein structure images, EvoLlama captures sequential and structural features from the primary and tertiary structures of proteins, offering a complementary perspective on multimodal protein representations.

Protein-oriented LLMs OntoProtein (Zhang et al., 2022a) integrate external factual knowledge from gene ontology into PLMs to enhance protein representations. ProteinChat (Guo et al., 2023) utilizes a Graph Neural Network encoder block combined with a Transformer encoder block to extract features from protein structures. Instead of introducing complex Transformer blocks, EvoLLAMA uses MLP as a lightweight and cost-effective approach to align different modalities. ProtST (Xu et al., 2023b) is a framework to enhance protein sequence understanding through biomedical texts by utilizing a Protein Language Model (PLM) and a Biomedical Language Model (BLM). The weights of BLM are initialized from PubMedBERT-abs (Gu et al., 2021), which is pre-trained on PubMed abstracts. ProtChatGPT (Wang et al., 2024a) is trained with sequence-text pairs using a Protein-Language Pretraining Transformer initialized with the pre-trained weights of PubMedBERT (Gu et al., 2021) to incorporate external knowledge. Compared to ProtST and ProtChatGPT, EvoLLAMA does not rely on LLMs specifically trained on external biomedical domain knowledge. Instead, we tune the projection layers from scratch, highlighting the generalizability of our plug-and-play architecture. ProteinGPT (Xiao et al., 2024) is a multimodal protein chat system trained through a two-stage process: modality alignment and instruction tuning. In contrast, our approach demonstrates that the initial alignment stage can be optional (Sec. 5.3), making our method more efficient and reducing training costs. Furthermore, EvoLLAMA employs a template-based strategy to construct the instruction-following dataset, avoiding the use of GPT-4o for generating the QA dataset as described in Xiao et al. (2024). This template-driven approach not only enables zero-shot capability in handling unseen instructions during the projection tuning stage (Sec. 5.1 and Sec. 5.2), showcasing the generalization and robustness of our method, but also eliminates the API call costs. ProteinCLIP (Wu et al., 2024) performs contrastive learning between protein sequences and texts by employing a frozen protein encoder and a frozen text embedding encoder. While ProteinCLIP is designed for protein function related tasks, our approach can be adapted to various downstream tasks including catalytic activity prediction and domain/motif prediction. ProtLLM (Zhuo et al., 2024) is a cross-modal LLM designed for protein-centric and protein-language tasks. Unlike ProtLLM, our approach integrates both structure and sequence representations of proteins, rather than relying solely on the sequence modality. This allows for a more comprehensive understanding of protein features.

Protein Representations DDE (Saravanan and Gautham, 2015), based on the dipeptide frequency within the protein sequence, and Moran (Feng and Zhang, 2000), which defines the distribution of amino acid properties along a protein sequence, are two typical protein sequence feature descriptors. Shallow CNN (Shanehsazzadeh et al., 2020) and ResNet (He et al., 2016) are protein sequence encoders designed to capture the short-range interactions within the protein sequence, while LSTM (Hochreiter and Schmidhuber, 1997) and Transformer (Vaswani et al., 2017) aim to capture the long-range interactions. The output layers of these protein sequence encoders aggregate the representations of different residues into a protein-level representation. Apart from these methods, some recent work has focused on simultaneously encoding protein sequences and structures. SaProt (Su et al., 2023) integrates residue tokens with structure tokens and is pre-trained on approximately 40M sequences and structures. In contrast, EvoLLAMA avoids the introduction of additional tokens and instead aligns the structure and sequence representations of proteins with embeddings from natural language prompts, achieving this with significantly fewer sequences and structures. ESM-3 (Hayes et al., 2024) is a generative language model that reasons over the sequence, structure, and function of proteins. Compared to ESM-3, our approach treats arbitrary protein functions as natural language rather than discrete function tokens or keywords, showing flexibility in both understanding and generating descriptive texts about protein functions.

B Dataset Construction Details

B.1 Projection Tuning Data

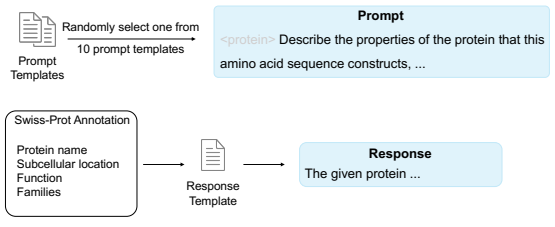


Figure 3: Overview of the projection tuning data construction.

Prompt Template

<protein> Elaborate on the characteristics of the protein specified by this amino acid sequence, detailing its biological role, cellular compartmentalization, and its classification within protein families.

<protein> Provide a comprehensive analysis of the protein encoded by the given amino acid sequence, focusing on its physiological function, location within the cell, and the protein family it is associated with.

<protein> Examine the protein determined by the provided amino acid sequence, including an in-depth look at its function, where it resides within the cell, and the broader family of proteins it is a member of.

<protein> Offer a detailed account of the protein described by this sequence of amino acids addressing its functional significance, subcellular niche, and its familial lineage within the protein taxonomy.

<protein> Describe the properties of the protein that this amino acid sequence constructs, including its operational role in the cell, its intracellular positioning, and the protein family to which it pertains.

<protein> Furnish a detailed exposition of the protein that correlates with the specified amino acid sequence, highlighting its role within cellular processes, its subcellular habitat, and the family of proteins it integrates with.

<protein> Detail the profile of the protein corresponding to the listed amino acid sequence, encompassing its function within the organism, its cellular localization, and the protein family it is derived from.

<protein> Discuss the protein that this amino acid sequence forms, including an analysis of its functional role, subcellular distribution, and its affiliation with particular protein families.

<protein> Present a thorough description of the protein represented by the amino acid sequence given, considering its function in biological systems, its localization within the cell, and the family of proteins it belongs to.

<protein> Illustrate the features of the protein specified by this sequence of amino acids, with emphasis on its functional activity, place within the cell structure, and its classification among protein families.

Response Template

```
{% if function is not none and similarity is not none %}
The given protein {{ protein_name }} suggests its potential function as {{ function }} It {{ similarity }}
{% elif function is none and similarity is not none %}
The given protein {{ protein_name }} {{ similarity }}
{% else %}
The given protein {{ protein_name }} suggests its potential function as {{ function }}
{% endif %}
```

Figure 4: The prompt and response template of the projection tuning data. In the response template, `similarity` refers to the families of the protein in Swiss-Prot.

As illustrated in Fig. 3, projection tuning data consists of sequence-description pairs originated from the Swiss-Prot (Consortium, 2023) database. The database contains 571K manually-annotated records, each containing information including protein name, subcellular location, function and families. To prevent from data leakage, we filter the Swiss-Prot annotation to 369K as our projection tuning data based on the downstream tasks.

For prompts, we construct 10 templates that ask the model to briefly describe the input protein from various aspects. For responses, information is extracted from the filtered Swiss-Prot annotation and constructed using a pre-defined template to ensure the consistency and clarity of protein descriptions. The prompt and response templates are listed in Fig. 4.

B.2 Supervised Fine-Tuning Data

As shown in Fig. 5, fine-tuning dataset consists of 10 tasks including PEER benchmark (Xu et al., 2022) and Mol-Instructions (Fang et al., 2023). For each task in PEER benchmark, there are 10 prompt templates and 1 response template, some of which are listed in Fig. 6. Except for fold classification, the categories in the response templates for other tasks are represented by natural language. For fold classification, we use integers ranging from 0 to 1194 to represent its categories due to the excessive number of categories.

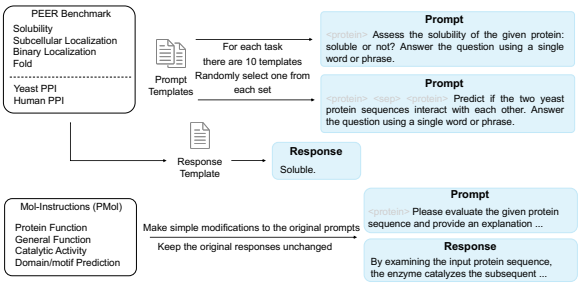


Figure 5: Overview of the supervised fine-tuning data construction.

For each task in Mol-Instructions, we make simple modifications to the original prompts to ensure that they are suitable for our use cases and maintain coherence. First, we remove the appended

Prompt Template	Response Template
<protein> Is the given protein water-soluble? Answer the question using a single word or phrase. ... (omit the other 9 prompt templates) ...	Not soluble./Soluble.
<protein> Identify the subcellular localization of the specified protein. Answer the question using a single word or phrase. ... (omit the other 9 prompt templates) ...	Cell membrane./Cytoplasm./ Endoplasmic reticulum./ ... (omit the other 7 responses) ...
<protein> Is the given protein membrane-bound or soluble? Answer the question using a single word or phrase. ... (omit the other 9 prompt templates) ...	Membrane-bound./Soluble.
<protein> Analyze the global fold topology of the specified protein. Answer the question using an integer between 0 and 1194 as the classification label. ... (omit the other 9 prompt templates) ...	0./1./.../1194.
<protein> <sep> <protein> Analyze the potential interaction between the two protein human sequences. Answer the question using a single word or phrase. ... (omit the other 9 prompt templates) ...	No./Yes.
<protein> <sep> <protein> Analyze the potential interaction between the two protein yeast sequences. Answer the question using a single word or phrase. ... (omit the other 9 prompt templates) ...	No./Yes.

Figure 6: The prompt and response template of PEER benchmark in the supervised fine-tuning data.

Original prompt used in Mol-Instructions:

Given the protein sequence **below**, please analyze and describe the catalytic activity of the corresponding enzyme, specifically the chemical reaction it catalyzes**:**

Modified prompt in our fine-tuning dataset:

<protein> Given the protein sequence **above**, please analyze and describe the catalytic activity of the corresponding enzyme, specifically the chemical reaction it catalyzes**.**

Original prompt used in Mol-Instructions:

Given the **following** protein sequence, can you perform an analysis and identify any potential motifs or domains**? Here is the sequence:**

Modified prompt in our fine-tuning dataset:

<protein> Given the protein sequence, can you perform an analysis and identify any potential motifs or domains?

Figure 7: Examples of modifications to the original prompts in Mol-Instructions. Red expressions are highlighted for modifications.

text-format FASTA sequences. Additionally, we modify some expressions in the original prompts. Some modification cases are listed in Fig. 7.

C Experimental Setups

For protein understanding tasks, we follow Mol-Instructions to split the dataset into an 8:1:1 ratio for training/validation/test, where the training and validation sets are used for the supervised fine-tuning stage, and the test set is used for assessing model performance. For protein property prediction tasks, we follow the PEER benchmark to split the dataset for each task. The details of dataset splits are listed in Tab. 6. The results are averaged over three runs with different random seeds. Specifically, we follow the PEER benchmark to report the mean and standard deviation of three runs' results.

We conduct both the projection tuning stage and supervised fine-tuning stage on 80GB H800 GPUs. The experiments to evaluate the inference latency, reported in Tab. 5, are conducted on 24GB RTX 3090 GPUs. The hyperparameters are listed in Tab. 7.

D Evaluation Implementation Details

For a fair comparison, we follow Mol-Instructions (Fang et al., 2023) to compute the ROUGE-L (Lin, 2004) score. Specifically, we take the complete references and predicted answers as inputs. However, both the references and predictions contain some non-protein-related parts, which are non-critical. In Fig. 8, we show that the critical parts are re-

Tasks	Sub-tasks	Data Source	#Training	#Validation	#Test
Protein Understanding Tasks	Solubility	PEER Benchmark	62,478	6,942	1,999
	Subcellular Localization		8,420	2,811	2,773
	Binary Localization		5,184	1,749	1,749
	Fold Classification		12,312	736	718
	Yeast PPI		9,890	190	788
	Human PPI		71,338	630	474
Protein Property Prediction Tasks	Protein Function	Mol-Instructions (PMol)	110,689		3,494
	Catalytic Activity		51,573		1,601
	Domain/Motif		43,700		1,400
	Functional Description		83,939		2,633

Table 6: Details of dataset splits for supervised fine-tuning data.

Stages	lr	Scheduler	Optimizer	#Batch Size	#Epochs/#Steps
Projection Tuning	2×10^{-4}	cosine	AdamW	64	2 epochs
Supervised Fine-tuning	2×10^{-5}	cosine	AdamW	32	25,000 steps

Table 7: Hyperparameters for the projection tuning stage and supervised fine-tuning stage.

Critical Non-critical

Domain/motif Prediction

Ground Truth: The computational analysis of the sequence suggests the presence of the following protein domains or motifs: Glutamine amidotransferase, CTP synthase N-terminal domains.

Prediction: Our predictive analysis of the given protein sequence reveals possible domains or motifs. These include: Glutamine amidotransferase, CTP synthase N-terminal domains.

Catalytic Activity Prediction

Ground Truth: Based on the provided protein sequence, the enzyme appears to facilitate the chemical reaction: H2O + L-glutamine = L-glutamate + NH4(+).

Prediction: Evaluation of the protein sequence indicates that the associated enzyme exhibits catalytic activity in the form of this chemical reaction: H2O + L-glutamine = L-glutamate + NH4(+).

Functional Description Generation

Ground Truth: A summary of the protein's main attributes with the input amino acid sequence reveals: Mono-ADP-ribosyltransferase that mediates mono-ADP- ribosylation of target proteins. Acts as a negative regulator of transcription.

Prediction: A short report on the protein with the given amino acid sequence highlights: Involved in the regulation of DNA repair.

Protein Function Prediction

Ground Truth: Based on the given amino acid sequence, the protein appears to have a primary function of ATP binding, DNA replication origin binding. It is likely involved in the DNA replication initiation, regulation of DNA replication, and its subcellular localization is within the cytoplasm.

Prediction: The protein with the amino acid sequence is expected to exhibit ATP binding, DNA replication origin binding, DNA replication origin recognition activity, contributing to the DNA replication, DNA replication initiation, regulation of DNA replication. It can be typically found in the cytoplasm of the cell.

Figure 8: Examples of computing ROUGE-L score on protein understanding tasks.

lated with the domains, catalytic activities, and functions. To illustrate how the critical parts affect the ROUGE-L score, we exclude the non-critical parts, retaining only the domains/motifs, chemical reactions, and functional descriptions in both the reference and prediction. Note that the mod-

ifications are applied to three sub-tasks, with the exception of the protein function prediction task, where every part of the generation is critical to the protein functions.

The re-evaluation results are displayed in Tab. 8, demonstrating that by excluding the non-critical parts from both the references and predictions, EvoLlama still outperforms the previous state-of-the-art model by 5%. Compared to the 6% improvement discussed in Sec. 5.1, the impact of non-critical parts is relatively minor.

Models	PF	GF	CA	DP	Avg.
	ROUGE-L				
Llama-2-7B-Chat	0.42	0.44	0.52	0.46	0.4600
EvoLlama	0.48	0.50	0.60	0.50	**0.5200**
	ROUGE-L (w/o non-critical parts)				
Llama-2-7B-Chat	0.42	0.46	0.56	0.57	0.5025
EvoLlama	0.48	0.42	0.60	0.71	**0.5525**

Table 8: Re-evaluation on protein understanding tasks. The ROUGE-L score is computed excluding the non-critical parts from both the references and predictions.

E More Evaluations

We conduct more evaluations on protein property prediction tasks for each ablation study introduced in Sec. 5.3.

Effect of the projection tuning stage As shown in Tab. 9, EvoLlama with supervised fine-tuning fails to maintain performance after the projection tuning stage, while EvoLlama in zero-shot settings remains competitive with the baselines fine-tuned on PMol (see Tab. 1). This indicates that bridging the gap between protein structures predicted by AlphaFold-2 and ESMFold during the supervised fine-tuning stage is more challenging than inference.

Effect of structure and sequence representations As shown in Tab. 10, EvoLlama and EvoLlama (GearNet+ESM-2) enhances the performance on protein understanding tasks by incorporating both structure and sequence representations, specifically on protein function prediction and catalytic activity prediction.

Effect of the fusion method As shown in Tab. 11, EvoLlama with fused representations outperforms the one without, particularly on subcellular localization prediction task, while EvoLlama (GearNet+ESM-2) achieves better performance when the structure and sequence representations are not fused. This demonstrates that the fusion method is more effective when ProteinMPNN is used as the structure encoder.

Effect of protein sequence encoder sizes The ESM-2 650M protein sequence encoder in EvoLlama is substituted with encoders of different sizes to demonstrate that the scaling law observed by Lin et al. (2022) extends to our multimodal framework. The experimental results in Fig. 9(a) indicate that performance on protein understanding tasks improves as the size of the protein sequence encoder increases. Furthermore, EvoLlama employing ESM-2 with only 8M parameters outperforms Llama-2-7B-Chat fine-tuned on PMol by an average of 27%. This result highlights the effectiveness of our approach, as the protein sequence encoder effectively captures evolutionary knowledge from amino acid sequences, substantially enhancing the LLM's understanding of proteins. We also conduct experiments on protein property prediction tasks, as illustrated in Fig. 9(b). The results show positive accuracy scaling across most tasks, with the exception of fold classification. A possible explanation is that the limited amount of training data makes it challenging for our multimodal architecture to effectively learn the distinctions among the 1,195 fold levels.

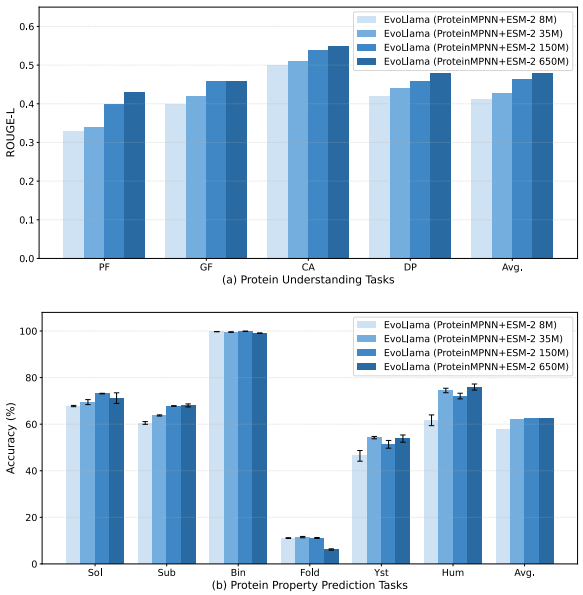


Figure 9: Effect of protein sequence encoder sizes. The experiments are conducted on (a) protein understanding tasks and (b) protein property prediction tasks.

Models	Sol	Sub	Bin	Fold	Yst	Hum	Avg.
EvoLlama	71.19±2.27	68.05±0.65	99.10±0.05	6.18±0.29	53.81±1.55	75.95±1.30	**62.38**
EvoLlama w/ PT	62.28±0.25	50.73±0.11	92.37±0.35	2.14±0.92	50.30±0.22	65.54±0.99	53.89

Table 9: More evaluations on the effect of the projection tuning stage. The experiments are conducted on protein property prediction tasks.

Models	Sol	Sub	Bin	Fold	Yst	Hum	Avg.
EvoLlama	71.19±2.27	68.05±0.65	99.10±0.05	6.18±0.29	53.81±1.55	75.95±1.30	**62.38**
EvoLlama w/o ProteinMPNN	70.91±0.10	68.63±0.21	99.73±0.03	7.94±0.11	54.48±0.60	65.12±2.76	61.14
EvoLlama w/o ESM-2	63.06±0.24	40.56±0.56	99.41±0.03	6.78±0.36	53.34±0.61	52.74±0.60	52.65
EvoLlama (GearNet+ESM-2)	67.20±0.40	37.96±0.33	91.96±0.06	3.67±0.29	52.16±0.63	60.41±0.10	**52.23**
EvoLlama (GearNet+ESM-2) w/o ESM-2	57.63±0.07	12.15±0.18	91.96±0.10	3.90±0.20	49.58±0.74	48.59±0.44	43.97

Table 10: More evaluations on the effect of structure and sequence representations. The experiments are conducted on protein property prediction tasks.

Models	Sol	Sub	Bin	Fold	Yst	Hum	Avg.
EvoLlama	71.19±2.27	68.05±0.65	99.10±0.05	6.18±0.29	53.81±1.55	75.95±1.30	**62.38**
EvoLlama w/o fused representations	69.53±0.87	59.29±2.35	99.16±0.50	10.91±0.37	56.26±1.09	74.40±0.70	61.59
EvoLlama (GearNet+ESM-2)	67.20±0.40	37.96±0.33	91.96±0.06	3.67±0.29	52.16±0.63	60.41±0.10	52.23
EvoLlama (GearNet+ESM-2) w/o fused representations	64.33±0.00	47.66±0.87	91.90±0.27	6.59±0.23	56.56±1.72	70.04±0.86	**56.18**

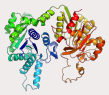
Table 11: More evaluations on the effect of the fusion method. The experiments are conducted on protein property prediction tasks.

F Case Study

As shown in Fig. 10, we compare the outputs of EvoLlama, EvoLlama (GearNet+ESM-2) and Mol-Instructions on domain/motif prediction task. Only EvoLlama correctly predicts all possible domains or motifs of the given protein, regardless of whether structure or sequence representations are incorporated. Compared to EvoLlama, Mol-Instructions fails to predict all the domains or motifs while EvoLlama (GearNet+ESM-2) generates incorrect ones, indicating that the structure representations extracted by GearNet fail to capture domain- or motif-related information.

As shown in Fig. 11, we compare the outputs of these models on catalytic activity prediction task. Both EvoLlama, with fused structure and sequence representations, and Mol-Instructions generate the accurate and complete chemical reaction. Furthermore, models with only structure or sequence representations fail to produce the correct chemical reaction, demonstrating the significance of fused representations.

Correct Incorrect **Domain/motif Prediction**

Input $\mathbf{X}_{\text{struct}}$: 

Input $\mathbf{X}_{\text{seq}}$:

MANTKYIFITGGVVSSLGKGIAAASIGALLESRGLSVSLIKVDPYINVDPGTMSPFQHGEVFVTEDGTETDLDLGHYERFVRFKASK
KNNFTAGKVYETVIRNERKGNYLGGTVQVIPHITNEIKKRIKKGGQNKDIAIVEVGGTVGDIESQPFVEALRQMALELPNSSWAFVH
LTLVPFINASGELKTKPTQHSVKELRSLGISPDVLVCRSEQELPKDEKNKIALFCSVPAKSVISMHDVDTVYSIPILLNKQKVDDTILK
KLNLKIKKPNLNDWKRVVKAKLLPEKEVNVSFVGKYTELKDSYKSINEALEHAGIQNKAKVNINFVEAEQITSQNVRKVLKKSDAILV
PGGFGERGIEGMILACKYARENNVPYLGICLGMQIAIIEYARNVLKLKSANSTEFDSSTKFPVIGLITEWSDISGKKEKRTKNSDLGG
TMRLGGQVCKLKKKSNSYKMYKKSEIIERHRHRYEVNPNYKDKMIEQGLDVVGTSIDGKLVEMIELPSHKWFLACQFHPEFTSNP
RDGHPIFNSYIKSTITK

Instruction: Given this protein sequence, can you identify any common protein motifs or domains that it may contain?

Output (Ground Truth): Our predictive analysis of the given protein sequence reveals possible domains or motifs. These include: Glutamine amidotransferase, CTP synthase N-terminal domains.

Output (Mol-Instructions): Based on computational analysis, the provided sequence potentially contains the following protein domains or motifs: Glutamine amidotransferase type-1 domains.

Output (EvoLlama (ProteinMPNN+ESM-2)): The computational analysis of the sequence suggests the presence of the following protein domains or motifs: Glutamine amidotransferase, CTP synthase N-terminal domains.

Output (EvoLlama (ProteinMPNN+ESM-2), w/o ProteinMPNN): Upon evaluating your submitted sequence, our predictive algorithms suggest the presence of: Glutamine amidotransferase, CTP synthase N-terminal domains.

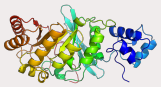
Output (EvoLlama (ProteinMPNN+ESM-2), w/o ESM-2): After analyzing the given sequence, the following protein domains or motifs are predicted: Glutamine amidotransferase, CTP synthase N-terminal domains.

Output (EvoLlama (GearNet+ESM-2)): The sequence you provided suggests the presence of a protein with a significant chance of containing a helicase, Rosy domain, SDH C-terminal domain, 'Y' domain, 'Z' domain type of protein.

Output (EvoLlama (GearNet+ESM-2), w/o ESM-2): Our bioinformatics algorithm indicates that the protein with sequence has potentially significant domains or motifs. These include: Helical transmembrane domains (e.g., 4-alpha-helical, 7-transmembrane G-protein coupled receptors).

Figure 10: Case study of performance on protein understanding tasks (Domain/motif prediction).

Correct Incorrect **Catalytic Activity Prediction**

Input $\mathbf{X}_{\text{struct}}$: 

Input $\mathbf{X}_{\text{seq}}$:
MDKVAVAGFLPEELCASLSLSPSFRGNQIFQWIGKGVDSFDAMTNLSAELRASLAEKAILRSTRVSDVLKADDGTVKLQIQTEDDL
AVETVLLTDKAARKTACVSCQAGCAMGCAFCKTGTLGLARNLSAAEIVEQFLYLEKHAGALDNIVFMGMGEPLLNLDALRKAIAVL
TDKRGRNLSSRRITVSTVGIVSGIYDLANNGPDVRLAVSLTTADETLRRELMPASLTNPLSDLRQAISYYIEKTGKRVTLEAVLLSGK
NTSEKNADSLIAFAKGLDVHVNLIPWNPVEGLSFVTPDPEETAQFVSRLEKGGLNVTLRMHRGKSISGACGQLGKTNPYA
Instruction: Please evaluate the following protein sequence and provide an explanation of the enzyme's catalytic activity, including the chemical reaction it facilitates:

Output (Ground Truth): Based on the provided protein sequence, the enzyme appears to facilitate the chemical reaction: adenosine(37) in tRNA + 2 reduced [2Fe-2S]-[ferredoxin] + 2 S-adenosyl-L-methionine = 2-methyladenosine(37) in tRNA + 5'-deoxyadenosine + L-methionine + 2 oxidized [2Fe-2S]-[ferredoxin] + S-adenosyl-L-homocysteine.

Output (Mol-Instructions): An analysis of the protein sequence reveals that the enzyme's catalytic function corresponds to the chemical reaction: adenosine(37) in tRNA + 2 reduced [2Fe-2S]-[ferredoxin] + 2 S-adenosyl-L-methionine = 2-methyladenosine(37) in tRNA + 5'-deoxyadenosine + L-methionine + 2 oxidized [2Fe-2S]-[ferredoxin] + S-adenosyl-L-homocysteine.

Output (EvoLlama (ProteinMPNN+ESM-2)): By examining the input protein sequence, the enzyme catalyzes the subsequent chemical reaction: adenosine(37) in tRNA + 2 reduced [2Fe-2S]-[ferredoxin] + 2 S-adenosyl-L-methionine = 2-methyladenosine (37) in tRNA + 5'-deoxyadenosine + L-methionine + 2 oxidized [2Fe-2S]-[ferredoxin] + S-adenosyl-L-homocysteine.

Output (EvoLlama (ProteinMPNN+ESM-2), w/o ProteinMPNN): By examining the input protein sequence, the enzyme catalyzes the hydrolytic removal of adenine (A) from tRNAs, resulting in 7,8-dihydro-8-oxo-A and 5,6-methyl-5,6-dihydro formyl-C.

Output (EvoLlama (ProteinMPNN+ESM-2), w/o ESM-2): Based on the provided protein sequence, the enzyme appears to facilitate the chemical reaction: adenosine 5'-phosphosulfamidase + adenosine 5'-phosphosulfate = (R)-(+)- \u00a0 (S)-methylmethionine + adenosine 5'-phosphate + sulfuric acid.

Output (EvoLlama (GearNet+ESM-2)): A chemical reaction and its participating enzymes have been identified: S-adenosyl-L-methionine tRNA (cytidine(34) 2'-O-)-methyltransferase [acetyl-CoA pathway].

Output (EvoLlama (GearNet+ESM-2), w/o ESM-2): The enzyme catalyzes the chemical reaction: 2 2-oxo-4-methyl-1,7-diazabicyclo[3.3.0]octane + 2 H(+) + 2 H(2)O = 2 2-oxo-4-methyl-1,7-diazabicyclo[3.3.0]octan-7-ol + 2 H(+) + N(2),N(5)-dimethyl-1,5-diaminopentane.

Figure 11: Case study of performance on protein understanding tasks (Catalytic activity prediction).