Multi-Modal Large Language Model Enables Protein Function Prediction

Han Guo [*1] **Mingjia Huo** [*1] **Xingyi Cheng** [2] **Digvijay Singh** [3] **Hamidreza Rahmani** [4] **Shen Li** [2] **Philipp Gerlof** [4] **Trey Ideker** [5] **Danielle A. Grotjahn** [4] **Elizabeth Villa** [3 6] **Le Song** [2 7] **Pengtao Xie** [1 8]

Abstract

Predicting the functions of proteins can greatly accelerate biological discovery and applications, where deep learning methods have recently shown great potential. However, these methods predominantly predict protein functions as discrete categories, which fails to capture the nuanced and complex nature of protein functions. Furthermore, existing methods require the development of separate models for each prediction task, a process that can be both resource-heavy and time-consuming. Here, we present ProteinChat, a versatile, multi-modal large language model that takes a protein's amino acid sequence as input and generates comprehensive narratives describing its function. ProteinChat is trained using over 1,500,000 (protein, prompt, answer) triplets curated from the Swiss-Prot dataset, covering diverse functions. This novel model can universally predict a wide range of protein functions, all within a single, unified framework. Furthermore, ProteinChat supports interactive dialogues with human users, allowing for iterative refinement of predictions and deeper exploration of protein functions. Our experimental results, evaluated through both human expert assessment and automated metrics, demonstrate that ProteinChat markedly outperforms various state-of-the-art baselines.

[*]Equal contribution [1]Department of Electrical and Computer Engineering, University of California San Diego [2]BioMap Research [3]School of Biological Sciences, University of California San Diego [4]Department of Integrative Structural and Computational Biology, The Scripps Research Institute [5]Division of Genetics, Department of Medicine, University of California San Diego [6]Howard Hughes Medical Institute, University of California San Diego [7]Mohamed bin Zayed University of Artificial Intelligence [8]Division of Biomedical Informatics, Department of Medicine, University of California San Diego. Correspondence to: Pengtao Xie <p1xie@ucsd.edu>.

Proceedings of the ICML 2025 Workshop on Multi-modal Foundation Models and Large Language Models for Life Sciences, Vancouver, Canada. 2025.

1. Introduction

Understanding protein functions and properties is crucial for advancing biological knowledge and driving innovations in drug discovery, disease treatment, and synthetic biology (Marcotte et al., 1999; Bepler & Berger, 2021; Watson et al., 2023; Listov et al., 2024; Kortemme, 2024). Predicting protein functions is a complex and challenging task due to the inherent diversity and intricate nature of proteins (Lee et al., 2007; Radivojac et al., 2013; Peled et al., 2016; Rives et al., 2021; Bileschi et al., 2022). Recent advancements in deep learning have demonstrated significant potential in improving the accuracy and efficiency of protein function prediction (Ryu et al., 2019; Wan & Jones, 2020; Gligorijević et al., 2021; Unsal et al., 2022; Wang et al., 2022; Zhou et al., 2022; Yu et al., 2023; Kulmanov et al., 2024). By leveraging extensive datasets of protein sequences, structures, and annotated functions, deep learning models can discern intricate patterns and relationships that often elude traditional computational methods.

However, existing deep learning-based methods for protein function prediction face significant limitations that prevent them from fully capturing the diverse range of protein functions. These methods typically predict protein functions as discrete categories (Radivojac et al., 2013; Wan & Jones, 2020; Gligorijević et al., 2021; Zhou et al., 2022; Yu et al., 2023; Kulmanov et al., 2024). This categorical formulation oversimplifies the complex and nuanced nature of proteins, which often perform multiple roles, engage in diverse interactions, and participate in intricate biological pathways.

Additionally, existing methods necessitate the development of specialized models for each prediction task, resulting in a fragmented approach that lacks efficiency and scalability (Peled et al., 2016; Gligorijević et al., 2021; Zhou et al., 2022; Wang et al., 2022; Yu et al., 2023; Kulmanov et al., 2024). The absence of a unified model capable of concurrently handling various prediction tasks limits a holistic understanding of protein functions. This fragmentation also increases the complexity and resource requirements for research and development, as developing, training, and maintaining multiple specialized models is significantly more challenging than managing a single, versatile model.

Large language models (LLMs) (Brown et al., 2020; Tou-

vron et al., 2023; Zhu et al., 2024) hold significant potential for addressing the limitations of current deep learning-based protein function prediction methods. These LLM models excel in generating high-quality text, making them well-suited for describing complex protein functions through comprehensive narratives. Furthermore, a single, pretrained LLM can perform a wide array of prediction tasks using task-specific user instructions or questions described in natural language (referred to as *prompts*) (Bubeck et al., 2023; Achiam et al., 2023; Wang et al., 2024), eliminating the necessity of training separate models for each task. Furthermore, LLMs facilitate interactive dialogues with human users (Chiang et al., 2023; Lee et al., 2023), enabling iterative refinement of generated textual predictions.

We developed ProteinChat, a multi-modal LLM that integrates two modalities - protein sequences and text. It takes an amino acid sequence and a prompt as inputs, and generates a detailed textual prediction of the protein's function. Unlike traditional methods that predict protein functions as discrete categories, ProteinChat generates coherent and comprehensive texts to predict the multifaceted functions of proteins, capturing the detailed roles, interactions, and biological context of proteins in a manner akin to human expert descriptions. Moreover, ProteinChat enables the use of diverse prompts for various prediction tasks that cover a wide range of protein functions and properties within this single tool, thereby streamlining the whole protein function exploration process without requiring new model training or extensive maintenance. Significantly outperforming current methods including GPT-4 (Achiam et al., 2023), BLASTp (Camacho et al., 2009), ProtNLM (Gane et al., 2024), and InstructProtein (Wang et al., 2024), ProteinChat can make accurate predictions across a broad spectrum of protein functions, which were evaluated using multiple metrics including assessments by human experts.

2. Method

ProteinChat accepts two types of inputs simultaneously: the amino acid sequence of a protein and a prompt tailored for easy, human-like dialogues with ProteinChat. For example, when given the prompt "describe the functions of this protein", ProteinChat generates a detailed free-form text describing the protein's various functions (Fig. 2a). Besides free-form prediction, ProteinChat can also predict specific function categories. For example when prompted with "What type of enzyme is this? Choose from [*a list of categories*]", ProteinChat chooses a specific answer from the list (Fig. 2a).

ProteinChat consists of three key modules: a protein encoder, an LLM, and an adaptor that bridges the two (Fig. 1). The protein encoder processes the amino acid sequence of the input protein, generating a representation vector for each amino acid, which captures the molecular characteristics of that amino acid. The adaptor aligns these representations with the LLM by transforming them into a format that is compatible with the LLM's input. Once this alignment is achieved, the LLM integrates the amino acid sequence with the prompt, and then utilizes this combined input to generate a textual prediction of the protein's function. We utilized xTrimoPGLM (Chen et al., 2025), a state-of-the-art protein language model, as the protein encoder, and Vicuna-13B (Chiang et al., 2023), fine-tuned from Llama-2 (Touvron et al., 2023), as the LLM of ProteinChat.

To train the ProteinChat model, we assembled a comprehensive dataset comprising (protein, prompt, answer) tripletts sourced from the Swiss-Prot database (UniProtKB, 2024), the expertly curated section of UniProt Knowledgebase (UniProtKB) (Consortium, 2022). The dataset contains approximately 1.5 million triplets from 567,467 proteins. In each triplet, the protein and prompt serve as inputs to the ProteinChat model, while the answer represents the desired output of ProteinChat. The answer can be either a detailed free-form text describing protein functions or a UniProtKB keyword representing a specific function category. This dataset comprehensively encompasses a diverse taxonomy of proteins and their various functions (Fig. 2b).

For the pretrained LLM (Vicuna-13B), we applied Low-Rank Adaptation (LoRA) (Hu et al., 2022) for fine-tuning. Specifically, a low-rank update matrix was added to each pretrained weight matrix. During fine-tuning, only the low-rank matrix was updated, while the original pretrained weight matrices remain fixed. For the pretrained protein encoder (xTrimoPGLM), full fine-tuning was utilized: all the pretrained weights were updated. The adaptor was trained from scratch. The trainable weights were optimized by minimizing the negative log-likelihood loss between the input data (proteins and prompts) and the corresponding output answers. Further details on the training of ProteinChat are provided in Section A.

3. Results

We evaluate our model's performance on two types of tasks – free-form protein function prediction (Section 3.1) and discrete function-category prediction (Section 3.2). Additional experimental results are provided in Appendix C.

3.1. ProteinChat achieved strong performance in free-form protein function prediction

Using the prompt "please describe the function of this protein", ProteinChat generated free-form text predictions for the functions of 700 test proteins from Swiss-Prot. These proteins were selected to ensure low sequence similarity with the training data, thereby mitigating the risk of information leakage and enabling evaluation of the model's

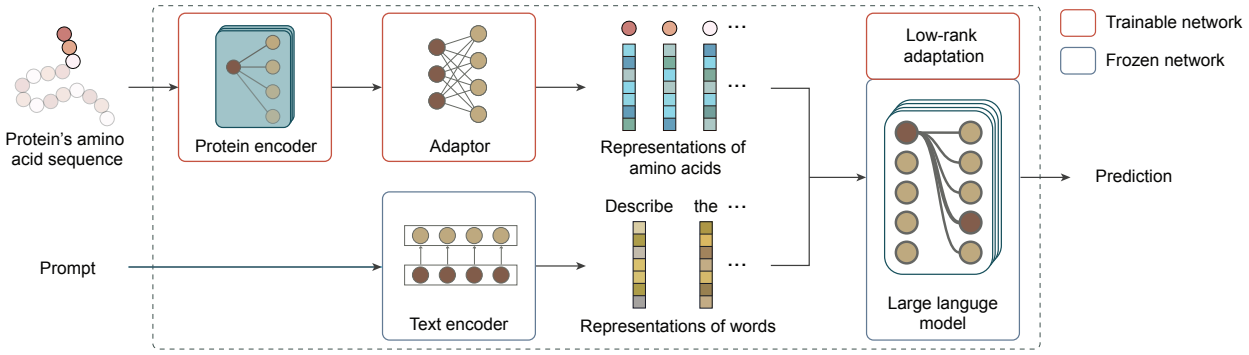

Figure 1: Model architecture of ProteinChat. It takes the amino acid sequence of a protein and a prompt as inputs, then generates a prediction in natural language. ProteinChat consists of a protein encoder that learns representation vectors for amino acids (AAs), an adaptor that transforms these representations into a format compatible with LLMs, and an LLM that generates the prediction based on the AAs' representations and the prompt.

ability to generalize to a diverse set of proteins deposited in Swiss-Prot across different time periods (Appendix A.1). The generated textual predictions offer more specific details about protein functions compared to discrete categories like Enzyme Commission (EC) numbers (Yu et al., 2023) and Gene Ontology terms (Ashburner et al., 2000; Consortium, 2019). As mentioned before, Swiss-Prot includes a textual description of each protein's function, which was used as ground truth in our evaluation.

We compared ProteinChat with two LLM-based models: GPT-4 and InstructProtein (Wang et al., 2024). GPT-4 is a flagship general-purpose large language model. InstructProtein was pretrained on protein sequences from UniRef100 (Suzek et al., 2015) and PubMed abstracts using the OPT-1.3B model (Zhang et al., 2022), and subsequently instruction-tuned on 5.2 million protein knowledge graph triples. We also evaluated ProteinChat against two similarity-based methods: BLASTp (Camacho et al., 2009) and ProtNLM (Gane et al., 2024).

We conducted a human evaluation of the predictions generated by ProteinChat and baseline models. Three domain experts specializing in proteins compared each prediction to its corresponding ground truth from Swiss-Prot. The evaluation considered two dimensions: correctness and completeness. Correctness measures the accuracy of the predicted function, analogous to precision — assessing whether the information is biologically valid and specific to the target protein. Completeness evaluates how comprehensively the prediction captures the ground truth function, analogous to recall — assessing whether the key functional elements are included. A detailed description of the assessment rubric can be found in Table 2.

ProteinChat achieved substantially higher scores in both correctness and completeness compared to all baseline methods (Figure 5a). For correctness, ProteinChat obtained an average score of 1.41, substantially outperforming GPT-4 (0.19), BLASTp (0.86), ProtNLM (0.72), and InstructProtein (0.45). For completeness, ProteinChat achieved an average score of 1.33, again substantially outperforming GPT-4 (0.16), BLASTp (0.71), ProtNLM (0.58), and InstructProtein (0.39). The superior performance of ProteinChat is evident not only in average scores but also in the distribution of evaluation scores. Compared to the baselines, ProteinChat received a substantially higher proportion of top scores (2). For correctness, 55.4% of ProteinChat's predictions received a score of 2, markedly higher than GPT-4 (3.6%), BLASTp (27.3%), ProtNLM (19.7%), and InstructProtein (12.0%). For completeness, 55.0% of ProteinChat's predictions received a score of 2, compared to 2.6% for GPT-4, 23.1% for BLASTp, 15.6% for ProtNLM, and 9.8% for InstructProtein.

In addition to human assessment, we employed widely used automated metrics to evaluate the similarity between predicted and ground truth functions. These include ROUGE-1, ROUGE-L (Lin, 2004), BLEU-1, BLEU-2 (Papineni et al., 2002), METEOR (Banerjee & Lavie, 2005), and SimCSE (Gao et al., 2021). ROUGE-1 measures unigram (word-level) overlap between predicted and reference texts, while ROUGE-L evaluates the longest common subsequence to capture sentence-level structural similarity. BLEU-n measures lexical similarity by comparing n-grams between the prediction and reference. METEOR computes a weighted harmonic mean of unigram precision and recall, incorporating synonymy and word order, and applies a penalty for fragmented alignments. SimCSE assesses semantic similarity by comparing the contextual embeddings of texts. All metrics produce scores where higher values indicate better alignment between prediction and ground truth. ROUGE, BLEU, and METEOR scores range from 0 to 1, while Sim-

CSE scores range from –1 to 1.

ProteinChat outperformed all five baseline methods across ROUGE, BLEU, METEOR, and SimCSE metrics (Fig. 5b). For example, it achieved a ROUGE-1 score of 0.25, compared to 0.14, 0.21, 0.18, and 0.11 for GPT-4, BLASTp, ProtNLM, and InstructProtein, respectively. Similarly, ProteinChat achieved a SimCSE score of 0.64, outperforming the baselines with scores of 0.42, 0.60, 0.50, and 0.56, respectively. For METEOR, ProteinChat attained a score of 0.16, compared to 0.10, 0.14, 0.13, and 0.11, respectively. These results further demonstrate the superior ability of ProteinChat to generate free-form protein function predictions.

Moreover, we leveraged a large language model to evaluate the free-form predictions. Specifically, given a free-form prediction and its corresponding ground truth annotation, we prompted the Claude 3.5 Sonnet (Anthropic, 2024) LLM to assign scores assessing how well the prediction matched the ground truth. The scoring rubric was identical to that used in the human evaluation. ProteinChat substantially outperformed all baseline methods, achieving an average correctness score of 0.75 and an average completeness score of 0.81 (Fig. 5c). These scores significantly exceeded those of the baselines: GPT-4 (0.04 correctness, 0.03 completeness), BLASTp (0.63 correctness, 0.73 completeness), ProtNLM (0.58 correctness, 0.65 completeness), and InstructProtein (0.12 correctness, 0.15 completeness). The Spearman correlation between Claude's evaluation scores and human evaluation scores was 0.88, indicating strong rank-level agreement. The results from human expert assessments and automated evaluations clearly demonstrates that ProteinChat significantly outperforms all baselines. This superior performance is primarily due to ProteinChat's enhanced ability in interpreting a fundamental language of biology, i.e., protein sequences (translated from DNA sequences), and the specialized training enables ProteinChat to offer precise annotations, identify functional domains, and predict potential interactions with high accuracy.

3.2. ProteinChat excels in predicting discrete function categories with high accuracy

While ProteinChat is designed as a general-purpose tool for generating detailed and nuanced descriptions of a protein's functions, it can also be customized for specific protein function prediction tasks where functions are categorized discretely. We applied ProteinChat to five specific protein function/property prediction tasks curated from UniProtKB, including catalytic function prediction, ligand binding function prediction, coenzyme-enzyme interaction prediction, biological process prediction, and cellular component compartmentalization prediction. These tasks encompass a broad spectrum of protein functions/properties (Appendix A.1).

To accomplish these tasks, we designed task-specific prompts (Appendix A.6) for ProteinChat, following a similar style. We employed accuracy, macro F1 score, and weighted F1 score as evaluation metrics, with F1 scores specifically accounting for both false positives and false negatives. We compared ProteinChat with InstructProtein and GPT-4. We also developed specialized classifier models, each designed to perform a specific prediction task, to evaluate how well ProteinChat, as a more general-purpose model, compares to these task-specific models.

Across all five prediction tasks, ProteinChat demonstrated near-optimal performance (Fig. 7a), achieving accuracy, macro F1, and weighted F1 scores between 0.94 and 0.99, and significantly outperforming InstructProtein and GPT-4. ProteinChat either outperformed or achieved comparable results of specialized classifiers, which is particularly remarkable given that ProteinChat employs a single model to handle all these prediction tasks, whereas the specialized classifiers are individually trained for each different task.

Next, we utilized ProteinChat to predict protein functions/properties represented by discrete Gene Ontology (GO) (Ashburner et al., 2000) categories and compared its performance against leading GO classifiers, including DeepGOPlus (Kulmanov & Hoehndorf, 2020) and NetGO 3.0 (Wang et al., 2023). Gene Ontology (GO) is a database that provides a hierarchical structure of categories widely used for annotating protein functions/properties. ProteinChat outperformed DeepGOPlus and NetGO 3.0 in predicting catalytic functions, biological processes, and cellular components (Fig. 7b). For instance, in predicting catalytic function, ProteinChat achieved a macro F1 score of 0.97, surpassing DeepGOPlus and NetGO, which obtained scores of 0.79 and 0.92, respectively. ProteinChat outperforms both DeepGOPlus and NetGO due to its ability in retaining and processing the entire sequence of amino acid representations using a protein language model. This ability allows ProteinChat to capture intricate relationships, positional context, and long-range dependencies within the sequence, which are essential for accurate protein function/properties prediction.

4. Conclusion

In conclusion, we present ProteinChat, a versatile tool for predicting protein functions represented in text using a multi-modal large language model. ProteinChat provides nuanced and in-depth predictions, surpassing both general-purpose LLMs and task-specific classifiers. Its ability in handling various prediction tasks within a single framework and facilitating interactive predictions allows for flexible, comprehensive, and in-depth analysis of protein functions. More in-indepth discussion can be found in Section D.

Software and Data

All data used in this study are available at https://drive.google.com/file/d/1ou2222905sAVljblc1VUH78Q4AQLYRjl/view?usp=sharing. The source code of this work is available at https://github.com/mignonjia/ProteinChat.

Impact Statement

This paper presents work whose goal is to advance the field of protein understanding using large language model and protein language model. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

BGC0001624 — mibig.secondarymetabolites.org. https://mibig.secondarymetabolites.org/repository/BGC0001624.3/#r1c1. [Accessed 15-03-2025].

Biosynthetic Gene Cluster Database with Functional Annotations — sr.iu.a.u-tokyo.ac.jp. https://sr.iu.a.u-tokyo.ac.jp/db/protein.pl?mibig_accession=BGC0001624&protein_id=APC57600.1#:~:text=419994cl23766fungal_TF_MHR%206.02e,component%20of%20the%20budding%20yeast. [Accessed 15-03-2025].

Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Anthropic. Claude 3.5 Sonnet. https://www.anthropic.com/news/claude-3-5-sonnet, June 2024. Large-language-model release note, accessed 11 May 2025.

Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., Davis, A. P., Dolinski, K., Dwight, S. S., Eppig, J. T., et al. Gene ontology: tool for the unification of biology. *Nature genetics*, 25(1):25–29, 2000.

Ashuach, T., Gabitto, M. I., Koodli, R. V., Saldi, G.-A., Jordan, M. I., and Yosef, N. Multivi: deep generative model for the integration of multimodal data. *Nature Methods*, 20(8):1222–1231, 2023.

Banerjee, S. and Lavie, A. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In Goldstein, J., Lavie, A., Lin, C.-Y., and Voss, C. (eds.), *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pp. 65–72, Ann Arbor, Michigan, June 2005. Association for Computational Linguistics. URL https://aclanthology.org/W05-0909/.

Bepler, T. and Berger, B. Learning the protein language: Evolution, structure, and function. *Cell systems*, 12(6): 654–669, 2021.

Bileschi, M. L., Belanger, D., Bryant, D. H., Sanderson, T., Carter, B., Sculley, D., Bateman, A., DePristo, M. A., and Colwell, L. J. Using deep learning to annotate the protein universe. *Nature Biotechnology*, 40(6):932–937, 2022.

Brown, D. W., Lee, S.-H., Kim, L.-H., Ryu, J.-G., Lee, S., Seo, Y., Kim, Y. H., Busman, M., Yun, S.-H., Proctor, R. H., et al. Identification of a 12-gene fusaric acid biosynthetic gene cluster in fusarium species through comparative and functional genomics. *Molecular Plant-Microbe Interactions*, 28(3):319–332, 2015.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901, 2020.

Bubeck, S., Chandrasekaran, V., Eldan, R., Gehrke, J., Horvitz, E., Kamar, E., Lee, P., Lee, Y. T., Li, Y., Lundberg, S., et al. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*, 2023.

Camacho, C., Coulouris, G., Avagyan, V., Ma, N., Papadopoulos, J., Bealer, K., and Madden, T. L. Blast+: architecture and applications. *BMC bioinformatics*, 10: 1–9, 2009.

Chen, B., Cheng, X., Li, P., Geng, Y.-a., Gong, J., Li, S., Bei, Z., Tan, X., Wang, B., Zeng, X., et al. xtrimopglm: unified 100-billion-parameter pretrained transformer for deciphering the language of proteins. *Nature Methods*, pp. 1–12, 2025.

Chiang, W.-L., Li, Z., Lin, Z., Sheng, Y., Wu, Z., Zhang, H., Zheng, L., Zhuang, S., Zhuang, Y., Gonzalez, J. E., Stoica, I., and Xing, E. P. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, March 2023. URL https://lmsys.org/blog/2023-03-30-vicuna/.

Consortium, G. O. The gene ontology resource: 20 years and still going strong. *Nucleic acids research*, 47(D1): D330–D338, 2019.

Consortium, T. U. UniProt: the Universal Protein Knowledgebase in 2023. *Nucleic Acids Research*, 51(D1): D523–D531, 11 2022. ISSN 0305-1048. doi: 10.1093/nar/gkac1052. URL https://doi.org/10.1093/nar/gkac1052.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, 2019.

Du, Z., Qian, Y., Liu, X., Ding, M., Qiu, J., Yang, Z., and Tang, J. Glm: General language model pretraining with autoregressive blank infilling. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 320–335, 2022.

Gane, A., Bileschi, M. L., Dohan, D., Speretta, E., Héliou, A., Meng-Papaxanthos, L., Zellner, H., Brevdo, E., Parikh, A., Martin, M. J., Orchard, S., Collaborators, U., and Colwell, L. J. Protnlm: Model-based natural language protein annotation. https://www.uniprot.org/help/ProtNLM, 2024.

Gao, T., Yao, X., and Chen, D. Simcse: Simple contrastive learning of sentence embeddings. In *EMNLP 2021-2021 Conference on Empirical Methods in Natural Language Processing, Proceedings*, 2021.

Gerke, J., Bayram, Ö., Feussner, K., Landesfeind, M., Shelest, E., Feussner, I., and Braus, G. H. Breaking the silence: protein stabilization uncovers silenced biosynthetic gene clusters in the fungus aspergillus nidulans. *Applied and environmental microbiology*, 78(23):8234–8244, 2012.

Gligorijević, V., Renfrew, P. D., Kosciolek, T., Leman, J. K., Berenberg, D., Vatanen, T., Chandler, C., Taylor, B. C., Fisk, I. M., Vlamakis, H., et al. Structure-based protein function prediction using graph convolutional networks. *Nature communications*, 12(1):3168, 2021.

Hadsell, R., Chopra, S., and LeCun, Y. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06)*, volume 2, pp. 1735–1742. IEEE, 2006.

Henikoff, S. and Henikoff, J. G. Amino acid substitution matrices from protein blocks. *Proceedings of the National Academy of Sciences*, 89(22):10915–10919, 1992.

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., and Chen, W. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

Hyman, A. A., Weber, C. A., and Jülicher, F. Liquid-liquid phase separation in biology. *Annual review of cell and developmental biology*, 30(1):39–58, 2014.

Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., et al. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, 2021.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

Kortemme, T. De novo protein design—from new structures to programmable functions. *Cell*, 187(3):526–544, 2024.

Kulmanov, M. and Hoehndorf, R. Deepgoplus: improved protein function prediction from sequence. *Bioinformatics*, 36(2):422–429, 2020.

Kulmanov, M., Guzmán-Vega, F. J., Duek Roggli, P., Lane, L., Arold, S. T., and Hoehndorf, R. Protein function prediction as approximate semantic entailment. *Nature Machine Intelligence*, 6(2):220–228, 2024.

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

Lee, P., Bubeck, S., and Petro, J. Benefits, limits, and risks of gpt-4 as an ai chatbot for medicine. *New England Journal of Medicine*, 388(13):1233–1239, 2023.

Li, J., Li, D., Savarese, S., and Hoi, S. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pp. 19730–19742. PMLR, 2023.

Lin, C.-Y. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pp. 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W04-1013.

Lin, P., Yan, Y., Tao, H., and Huang, S.-Y. Deep transfer learning for inter-chain contact predictions of transmembrane protein complexes. *Nature Communications*, 14(1):4935, 2023a.

Lin, Z., Akin, H., Rao, R., Hie, B., Zhu, Z., Lu, W., Smetanin, N., Verkuil, R., Kabeli, O., Shmueli, Y., et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023b.

Listov, D., Goverde, C. A., Correia, B. E., and Fleishman, S. J. Opportunities and challenges in design and optimization of protein function. *Nature Reviews Molecular Cell Biology*, pp. 1–15, 2024.

Liu, H., Li, C., Wu, Q., and Lee, Y. J. Visual instruction tuning. *Advances in neural information processing systems*, 36, 2024.

Liu, S., Nie, W., Wang, C., Lu, J., Qiao, Z., Liu, L., Tang, J., Xiao, C., and Anandkumar, A. Multi-modal molecule structure–text model for text-based retrieval and editing. *Nature Machine Intelligence*, 5(12):1447–1457, 2023.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

Lo, H.-C., Entwistle, R., Guo, C.-J., Ahuja, M., Szewczyk, E., Hung, J.-H., Chiang, Y.-M., Oakley, B. R., and Wang, C. C. Two separate gene clusters encode the biosynthetic pathway for the meroterpenoids austinol and dehydroaustinol in aspergillus nidulans. *Journal of the American Chemical Society*, 134(10):4709–4720, 2012.

Lovell, S. C. Are non-functional, unfolded proteins ('junk proteins') common in the genome? *FEBS letters*, 554(3): 237–239, 2003.

Marcotte, E. M., Pellegrini, M., Ng, H.-L., Rice, D. W., Yeates, T. O., and Eisenberg, D. Detecting protein function and protein-protein interactions from genome sequences. *Science*, 285(5428):751–753, 1999.

Meta. Introducing meta llama 3: The most capable openly available llm to date. https://ai.meta.com/blog/meta-llama-3/, 2024.

Mirdita, M., Schütze, K., Moriwaki, Y., Heo, L., Ovchinnikov, S., and Steinegger, M. Colabfold: making protein folding accessible to all. *Nature methods*, 19(6):679–682, 2022.

Nielsen, M. L., Nielsen, J. B., Rank, C., Klejnstrup, M. L., Holm, D. K., Brogaard, K. H., Hansen, B. G., Frisvad, J. C., Larsen, T. O., and Mortensen, U. H. A genome-wide polyketide synthase deletion library uncovers novel genetic links to polyketides and meroterpenoids in aspergillus nidulans. *FEMS microbiology letters*, 321(2): 157–166, 2011.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pp. 311–318, 2002.

Peled, S., Leiderman, O., Charar, R., Efroni, G., Shav-Tal, Y., and Ofran, Y. De-novo protein function prediction using dna binding and rna binding proteins as a test case. *Nature communications*, 7(1):13424, 2016.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PMLR, 2021.

Radivojac, P., Clark, W. T., Oron, T. R., Schnoes, A. M., Wittkop, T., Sokolov, A., Graim, K., Funk, C., Verspoor, K., Ben-Hur, A., et al. A large-scale evaluation of computational protein function prediction. *Nature methods*, 10(3):221–227, 2013.

Rives, A., Meier, J., Sercu, T., Goyal, S., Lin, Z., Liu, J., Guo, D., Ott, M., Zitnick, C. L., Ma, J., 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.

Rost, B. Twilight zone of protein sequence alignments. *Protein engineering*, 12(2):85–94, 1999.

Ryu, J. Y., Kim, H. U., and Lee, S. Y. Deep learning enables high-quality and high-throughput prediction of enzyme commission numbers. *Proceedings of the National Academy of Sciences*, 116(28):13996–14001, 2019.

Sanchez, J. F., Chiang, Y.-M., Szewczyk, E., Davidson, A. D., Ahuja, M., Oakley, C. E., Bok, J. W., Keller, N., Oakley, B. R., and Wang, C. C. Molecular genetic analysis of the orsellinic acid/f9775 gene cluster of aspergillus nidulans. *Molecular Biosystems*, 6(3):587–593, 2010.

Schroeckh, V., Scherlach, K., Nützmann, H.-W., Shelest, E., Schmidt-Heck, W., Schuemann, J., Martin, K., Hertweck, C., and Brakhage, A. A. Intimate bacterial–fungal interaction triggers biosynthesis of archetypal polyketides in aspergillus nidulans. *Proceedings of the National Academy of Sciences*, 106(34):14558–14563, 2009.

Suzek, B. E., Wang, Y., Huang, H., McGarvey, P. B., Wu, C. H., and Consortium, U. Uniref clusters: a comprehensive and scalable alternative for improving sequence similarity searches. *Bioinformatics*, 31(6):926–932, 2015.

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

UniProtKB. Swiss-prot dataset. https://www.uniprot.org/uniprotkb?query=reviewed:true, 2024.

Unsal, S., Atas, H., Albayrak, M., Turhan, K., Acar, A. C., and Doğan, T. Learning functional properties of proteins with language models. *Nature Machine Intelligence*, 4 (3):227–245, 2022.

Van Der Lee, R., Buljan, M., Lang, B., Weatheritt, R. J., Daughdrill, G. W., Dunker, A. K., Fuxreiter, M., Gough, J., Gsponer, J., Jones, D. T., et al. Classification of intrinsically disordered regions and proteins. *Chemical reviews*, 114(13):6589–6631, 2014.

Van der Maaten, L. and Hinton, G. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Wan, C. and Jones, D. T. Protein function prediction is improved by creating synthetic feature samples with generative adversarial networks. *Nature Machine Intelligence*, 2(9):540–550, 2020.

Wang, J., Lisanza, S., Juergens, D., Tischer, D., Watson, J. L., Castro, K. M., Ragotte, R., Saragovi, A., Milles, L. F., Baek, M., et al. Scaffolding protein functional sites using deep learning. *Science*, 377(6604):387–394, 2022.

Wang, S., You, R., Liu, Y., Xiong, Y., and Zhu, S. Netgo 3.0: protein language model improves large-scale functional annotations. *Genomics, Proteomics & Bioinformatics*, 21 (2):349–358, 2023.

Wang, W., Yu, Y., Keller, N. P., and Wang, P. Presence, mode of action, and application of pathway specific transcription factors in aspergillus biosynthetic gene clusters. *International Journal of Molecular Sciences*, 22(16): 8709, 2021.

Wang, Z., Zhang, Q., Ding, K., Qin, M., Zhuang, X., Li, X., and Chen, H. InstructProtein: Aligning human and protein language via knowledge instruction. In Ku, L.-W., Martins, A., and Srikumar, V. (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1114–1136, Bangkok, Thailand, August 2024. Association for Computational Linguistics. URL https://aclanthology.org/2024.acl-long.62.

Watson, J. L., Juergens, D., Bennett, N. R., Trippe, B. L., Yim, J., Eisenach, H. E., Ahern, W., Borst, A. J., Ragotte, R. J., Milles, L. F., et al. De novo design of protein structure and function with rfdiffusion. *Nature*, 620(7976): 1089–1100, 2023.

Xu, M., Yuan, X., Miret, S., and Tang, J. Protst: Multimodality learning of protein sequences and biomedical texts. In *International Conference on Machine Learning*, pp. 38749–38767. PMLR, 2023.

Yu, T., Cui, H., Li, J. C., Luo, Y., Jiang, G., and Zhao, H. Enzyme function prediction using contrastive learning. *Science*, 379(6639):1358–1363, 2023.

Zhang, S., Roller, S., Goyal, N., Artetxe, M., Chen, M., Chen, S., Dewan, C., Diab, M., Li, X., Lin, X. V., et al. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*, 2022.

Zhou, X., Zheng, W., Li, Y., Pearce, R., Zhang, C., Bell, E. W., Zhang, G., and Zhang, Y. I-tasser-mtd: a deep-learning-based platform for multi-domain protein structure and function prediction. *Nature Protocols*, 17(10): 2326–2353, 2022.

Zhu, D., Chen, J., Shen, X., Li, X., and Elhoseiny, M. Minigpt-4: Enhancing vision-language understanding with advanced large language models. In *International Conference on Learning Representations*, 2024.

A. Method and experiment details

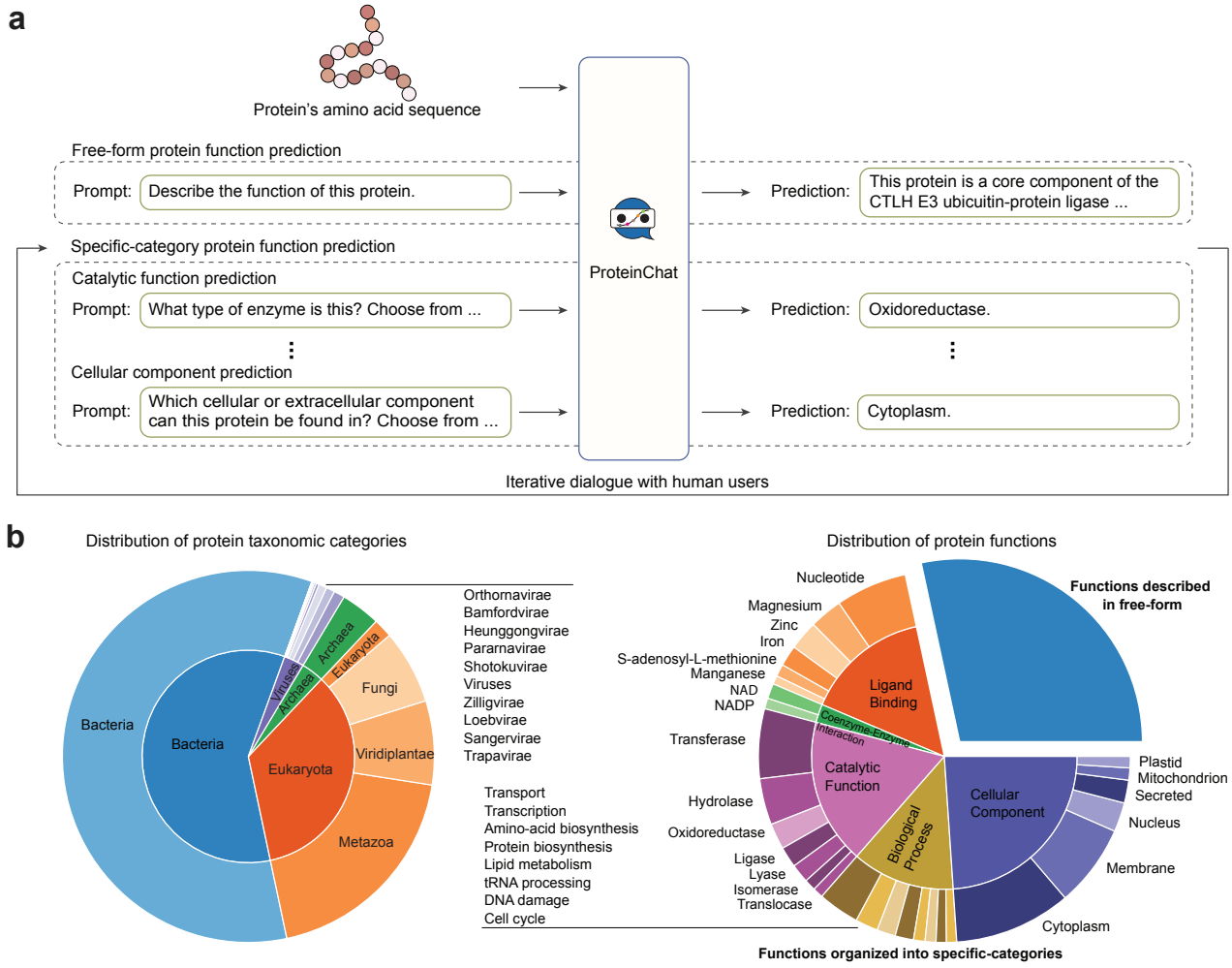

Figure 2: **ProteinChat is a multi-modal LLM capable of predicting protein functions represented either in free-form text or as specific categories. a**, ProteinChat enables versatile prediction of protein functions, allowing users to submit various requests in flexible natural language (known as prompts). By using task-specific prompts, ProteinChat can perform a variety of prediction tasks within a single framework without changing model parameters. ProteinChat facilitates interactive dialogues with users by retaining the conversation history, including prompts and corresponding predictions, allowing for in-depth analysis of a specific protein over multiple interactions. **b**, An extensive dataset, comprising proteins from various taxonomic groups, was constructed to train ProteinChat. In the left pie chart, the inner ring represents superkingdoms, while the outer ring represents kingdoms. ProteinChat was trained to make two types of predictions: one generates free-form textual descriptions, and the other predicts specific function categories. The pie chart on the right displays the relative proportions of the training data devoted to these two types.

A.1. Data preprocessing

We collected the amino acid sequences of proteins and their functions from Swiss-Prot (UniProtKB, 2024), the reviewed subset of proteins in UniProtKB (Consortium, 2022). The "Function" section in UniProtKB provides a textual description of a protein's functions. Additionally, the "Keywords" section offers a controlled vocabulary with a hierarchical structure that describes various aspects of protein functions, including activities, locations, interactions, and more. The Swiss-Prot database within UniProtKB, which was manually curated by experts, serves as a high-quality reference for protein functions.

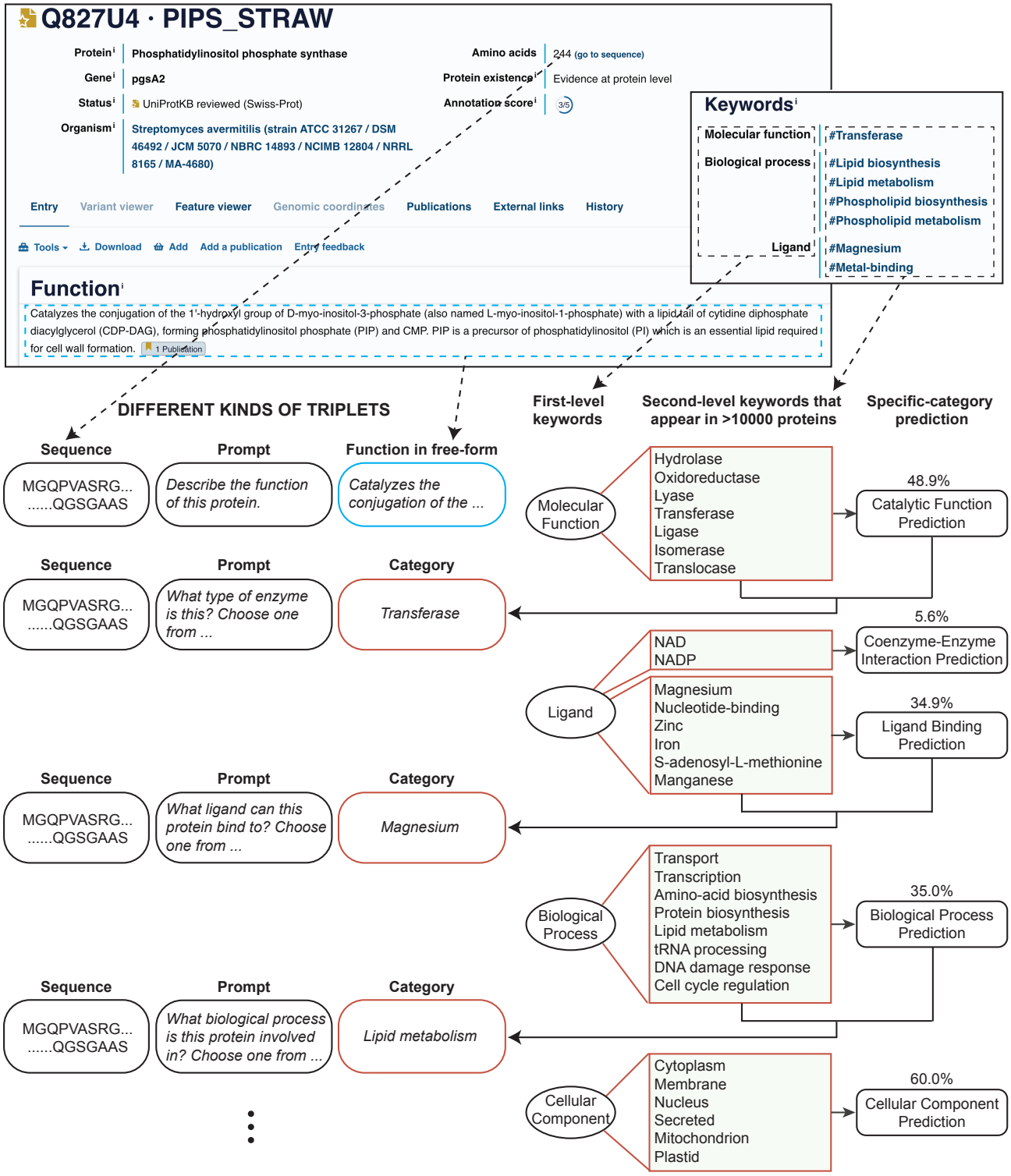

Figure 3: An illustration of the process used to curate (protein sequence, prompt, answer) triplets from the Swiss-Prot database. The percentages represent the proportion of protein entries in Swiss-Prot that include keywords corresponding to the listed categories.

The data used in this study was based on the UniProt 2023_02 version, released on May 2nd, 2023[1]. We downloaded the metadata in JSON format and extracted the protein functions by filtering entries where `commentType` is set to "Function". We excluded all functions that contain the `molecule` field, indicating that the function pertains to a subsequence of amino acids after clipping rather than the entire protein sequence. This exclusion is necessary because the protein can serve as a precursor to various chains or peptides. UniProtKB specifies the role of each peptide separately under distinct `molecule`[2] entries. As a result, functions for 2,049 proteins were excluded, reducing the total to 567,467 proteins.

For the free-form prediction task, we selected 700 proteins to form the test set. For each discrete classification task, 1,000 proteins were randomly selected as the test set. The remaining proteins were used for model training. Among the 700 proteins used for evaluating free-form prediction, 350 were deposited in UniProtKB-SwissProt prior to February 2023, and 350 were deposited after February 2023. No sequences added after this date were included in the training set. To ensure sufficient sequence diversity and minimize potential overlap with the training data, we applied the following criteria for test set selection:

- BLASTp query coverage ≤ 0.4. To limit sequence similarity between the test and training sets, we applied a strict threshold on BLASTp query coverage. For a given test protein, we computed the BLASTp alignment against the training set and defined query coverage as the proportion of the test sequence that aligns to any training sequence in the highest-scoring alignment segment. A query coverage of 0.4 means that no more than 40% of the test sequence can be aligned to a sequence seen during training. This constraint ensures that the majority of the test sequence remains unseen during training, thereby reducing the potential for the model to make predictions based on memorized sequence fragments. By enforcing this low-coverage threshold, we encourage the evaluation to reflect true generalization to novel or weakly homologous proteins, rather than pattern recall from closely related sequences.

- For sequences deposited prior to February 2023, we explicitly excluded from the test set any entries that also appear in the training set.

These criteria mitigate the risk of data leakage and help ensure that the test set includes a representative mix of both previously known and more recently annotated proteins. In the pre-February 2023 subset, the average query coverage is 0.1245 and the average percentage of identical matches is 0.1865. In the post-February 2023 subset, these values are 0.0083 and 0.3468, respectively. Overall, the combined test set has an average query coverage of 0.0649 and an average sequence identity of 0.2667. According to the sequence similarity guidelines outlined in (Rost, 1999), these values reflect a low degree of similarity between the training and test sequences, thereby substantially reducing the risk of information leakage. The original training set was further split into a new training set and a validation set in a 9:1 ratio.

From the training proteins and their associated textual descriptions of functions, we curated the training dataset for ProteinChat (Figure 3). For each training protein p, we created a training example represented as a triplet (protein's amino acid sequence, prompt, answer). The amino acid sequence and the prompt serve as the inputs to ProteinChat, while the answer is the expected output. Specifically, the amino acid sequence of p serves as the first element in the triplet, the prompt "Describe the function of this protein" forms the second element, and the textual description of p's function acts as the third element. To enhance ProteinChat's robustness against linguistic variations, we also employed other semantically equivalent prompts during the training process (Zhu et al., 2024). Additionally, we generated training triplets based on UniProtKB keywords, which are organized into a hierarchy. There are 10 first-level keywords, and we selected 4 that are relevant to protein functions, including molecular functions, binding properties, biological processes, and cellular localization. Furthermore, we chose 29 second-level keywords associated with over 10,000 proteins. These keywords cover 84% of all proteins in Swiss-Prot. Table 1 was used to curate training triplets from keywords. For a given protein p associated with a keyword k, the corresponding prompt t for k was identified from this table. For example, if the keyword is KW-0808 ("Transferase"), the corresponding prompt is "What type of enzyme is this? Choose one from the following options: hydrolase, oxidoreductase, lyase, transferase, ligase, isomerase, and translocase." This forms the triplet (p, t, k). On average, 2.7 triplets were curated per protein. The final training dataset for ProteinChat was formed by combining triplets curated from textual descriptions of functions and keywords. Similarly, a validation set of triplets was curated from the validation proteins.

[1] https://www.uniprot.org/release-notes/2023-05-03-release

[2] https://www.uniprot.org/help/function

A.2. ProteinChat model

ProteinChat employs xTrimoPGLM-1B (Chen et al., 2025) as the protein sequence encoder and Vicuna-13B (Chiang et al., 2023) as the large language model. The xTrimoPGLM-1B model comprises 24 Transformer (Vaswani et al., 2017) layers, 32 attention heads, and an embedding dimension of 2048. It was pretrained on the Uniref90 (Suzek et al., 2015) and ColabFoldDB (Mirdita et al., 2022) datasets using two strategies: masked language modeling (MLM) (Devlin et al., 2019) and general language modeling (GLM) (Du et al., 2022). The MLM strategy enhances xTrimoPGLM-1B's understanding of protein sequences, while the GLM strategy improves its generative capabilities. Vicuna-13B, fine-tuned from Llama2-13B (Touvron et al., 2023), retains the same architecture as Llama2-13B including 40 Transformer layers, 40 attention heads, and an embedding dimension of 5120. Vicuna-13B was trained by fine-tuning Llama2-13B on a dataset of 70K user-shared dialogues collected from ShareGPT.com.

For an input protein $\mathbf{x}_p$, we utilize the pretrained xTrimoPGLM-1B encoder g to generate a protein embedding $g(\mathbf{x}_p)$ of size $l \times 2048$, with l to be the length of the amino acid sequence. A linear layer (i.e., adaptor) $\mathbf{W}$ is applied to map these protein embeddings to the LLM input embedding space, resulting in a new embedding $\mathbf{h}_p = g(\mathbf{x}_p) \times \mathbf{W}$ of size $l \times 5120$. This embedding can be directly input into the LLM to represent the protein. To combine the protein embedding with the textual prompt, we design the LLM Input and Response fields following the conversational format of Vicuna (Chiang et al., 2023):

- (LLM Input) Human: <Protein> `ProteinHere` </Protein>`Prompt` Assistant:
- (LLM Response) `Answer`

As previously mentioned, each training example consists of a (protein, prompt, answer) triplet. We replace the placeholders `Prompt` and `Answer` with the corresponding elements from the triplet. All text in the LLM input, except for `ProteinHere`, is referred to as the *auxiliary prompt*, including the special characters <, >, and /. We denote the tokenized auxiliary prompt as $\mathbf{x}_{\text{aux}}$. Next, we use the LLM to embed $\mathbf{x}_{\text{aux}}$, resulting in the auxiliary prompt embedding $\mathbf{h}_{\text{aux}}$. After obtaining this embedding, we replace `ProteinHere` with the protein embedding $\mathbf{h}_p$ generated by the adaptor and feed the entire prompt into the LLM.

The model is trained using a language modeling task, where it learns to generate successive tokens by considering the preceding context. During the training process, the main objective is to optimize the log-likelihood of these tokens. In ProteinChat, only the `Answer` part is used to compute the loss. By explicitly adding an ending symbol to the answer, the model is also trained to predict where to stop. Specifically, for a target answer $\mathbf{x}_a$ of length l, we compute the probability of generating $\mathbf{x}_a$ by:

$$p(\mathbf{x}_a \mid \mathbf{x}_p, \mathbf{x}_{\text{aux}}) = \prod_{i=0}^{l} p_{\boldsymbol{\theta}} \left(\mathbf{x}_a^{(i)} \mid \mathbf{x}_p, \mathbf{x}_{\text{aux}}, \mathbf{x}_a^{<(i)} \right), \tag{1}$$

where $\mathbf{x}_p$ is the protein sequence and $\mathbf{x}_{\text{aux}}$ is the auxiliary prompt in tokens. $\mathbf{x}_a$ is the answer to be trained on. We use $\mathbf{x}_a^{(i)}$ and $\mathbf{x}_a^{<(i)}$ to denote the i-th token and all tokens before the i-th one. $\boldsymbol{\theta}$ denotes the trainable model parameters.

A.3. Training details of ProteinChat

We used the Adam (Kingma & Ba, 2015) optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and a weight decay of 0.05. We applied a cosine learning rate decay with a peak learning rate of $1e$-5 and a linear warm-up of 2000 steps. The minimum learning rate was $1e$-6. Due to the high memory consumption required for fine-tuning the encoder and LLM, we utilized a mini-batch size of one per GPU and limited the protein length to a maximum of 600 residues. Notably, 87.1% of the proteins had sequence lengths within this limit. For protein sequences longer than this limit, we truncated the excess length. We used 8 NVIDIA A100 GPUs, with 4 accumulation steps, resulting in an effective batch size of 32. We trained the model for 210K steps. In LoRA, we set the rank to 8, LoRA alpha to 16, and dropout rate to 0.05.

A.4. Evaluation metrics

We employed SimCSE (Gao et al., 2021) to assess the semantic similarity between the ground truth protein function and the predicted function. SimCSE leverages a contrastive learning framework (Hadsell et al., 2006) and utilizes the RoBERTa-base (Liu et al., 2019) model (denoted by f_θ) to generate sentence embeddings. The semantic similarity is quantified by calculating the cosine similarity of these embeddings, with scores ranging from -1 to 1, where higher values

signify greater semantic alignment. Specifically, let s and s' represent the ground truth protein function and the predicted function, respectively. The SimCSE score is computed as:

$$\cos_{\text{sim}}(f_\theta(s), f_\theta(s')),$$

where $f_\theta(s)$ and $f_\theta(s')$ are the embeddings of s and s' extracted by the RoBERTa-base model f_θ. $\cos_{\text{sim}}(\cdot,\cdot)$ denotes the cosine similarity operation.

BLEU (Papineni et al., 2002) is computed using a set of modified n-gram precisions. Specifically,

$$\text{BLEU} = \text{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right), \tag{2}$$

where p_n is the modified precision for n-gram, $w_n > 0$ and $\sum_{n=1}^{N} w_n = 1$. The brevity penalty (BP) is applied to penalize short generated text. Let c be the length of the generated text and r be the length of the ground truth. BP is computed as follows:

$$\text{BP} = \left\{ \begin{array}{ll} 1 & \text{if } c > r \\ \exp(1 - \frac{r}{c}) & \text{if } c \leq r \end{array} \right\} \tag{3}$$

The weighted F1 score is calculated by averaging the F1 score of all categories, weighted by the true instances (support) in each category. The macro F1 score is computed as a simple arithmetic mean of the F1 scores across all categories, without considering their support.

ROUGE-1 (Lin, 2004) evaluates the unigram (i.e., word-level) overlap between a candidate text C and a set of reference texts R. It is defined as the recall of overlapping unigrams:

$$\text{ROUGE-1} = \frac{\sum_{r \in R} \sum_{w \in r} \min\left(\text{Count}_C(w), \text{Count}_r(w)\right)}{\sum_{r \in R} \sum_{w \in r} \text{Count}_r(w)} \tag{4}$$

Here, $\text{Count}_C(w)$ denotes the number of times word w appears in the candidate text C, and $\text{Count}_r(w)$ denotes the number of times w appears in a reference $r \in R$. The numerator sums the minimum unigram counts across candidate and reference texts, while the denominator sums the total unigram counts in the references, yielding a recall-oriented measure.

ROUGE-L (Lin, 2004) captures sentence-level structure similarity by computing the length of the longest common subsequence (LCS) between the candidate and reference sequences. Given a candidate sequence C and a reference sequence r, let $\text{LCS}(C, r)$ be the length of their longest common subsequence. The precision P, recall R, and F-measure F are defined as:

$$P = \frac{\text{LCS}(C, r)}{|C|}, \quad R = \frac{\text{LCS}(C, r)}{|r|}, \quad F = \frac{(1 + \beta^2) \cdot P \cdot R}{R + \beta^2 \cdot P} \tag{5}$$

Here, $|C|$ and $|r|$ denote the lengths of the candidate and reference sequences, respectively. The parameter β (typically set to 1.2) balances the relative importance of recall and precision. The final ROUGE-L score is computed by taking the maximum F-measure over all reference sequences for each candidate.

METEOR (Metric for Evaluation of Translation with Explicit ORdering) (Banerjee & Lavie, 2005) evaluates the quality of a candidate sentence by aligning it to reference sentences based on exact matches, stemmed matches, synonym matches, and paraphrase matches. Once an alignment is established between the candidate and reference, METEOR computes unigram-level precision and recall, followed by a harmonic mean that favors recall. Let m denote the number of mapped unigrams between the candidate C and reference R, $|C|$ the number of unigrams in the candidate, and $|R|$ the number of unigrams in the reference. The precision P and recall R are computed as:

$$P = \frac{m}{|C|}, \quad R = \frac{m}{|R|} \tag{6}$$

The harmonic mean F_α of precision and recall, weighted by a parameter α (typically set to 0.9), is given by:

$$F_\alpha = \frac{P \cdot R}{\alpha \cdot P + (1 - \alpha) \cdot R} \tag{7}$$

To penalize alignments with many chunks (i.e., non-contiguous matched segments), a fragmentation penalty is applied. Let ch be the number of such chunks. The penalty is defined as:

$$Penalty = \gamma \left(\frac{ch}{m} \right)^{\theta} \tag{8}$$

where γ and θ are parameters (commonly $\gamma = 0.5$, $\theta = 3.0$). The final METEOR score is:

$$\text{METEOR} = (1 - Penalty) \cdot F_{\alpha} \tag{9}$$

This formulation encourages alignments that are both accurate (high precision and recall) and fluent (low fragmentation), making METEOR more sensitive to word order and synonymy compared to n-gram based metrics.

When using Claude 3.5 Sonnet to assess free-form predictions, we used the following prompt: "You are a biologist specializing in protein functions. Evaluate the predicted function '[predicted function]' against the ground truth function '[ground truth function]' in terms of correctness and completeness, using the following scoring scales..." The detailed scoring scales are provided in Table 2.

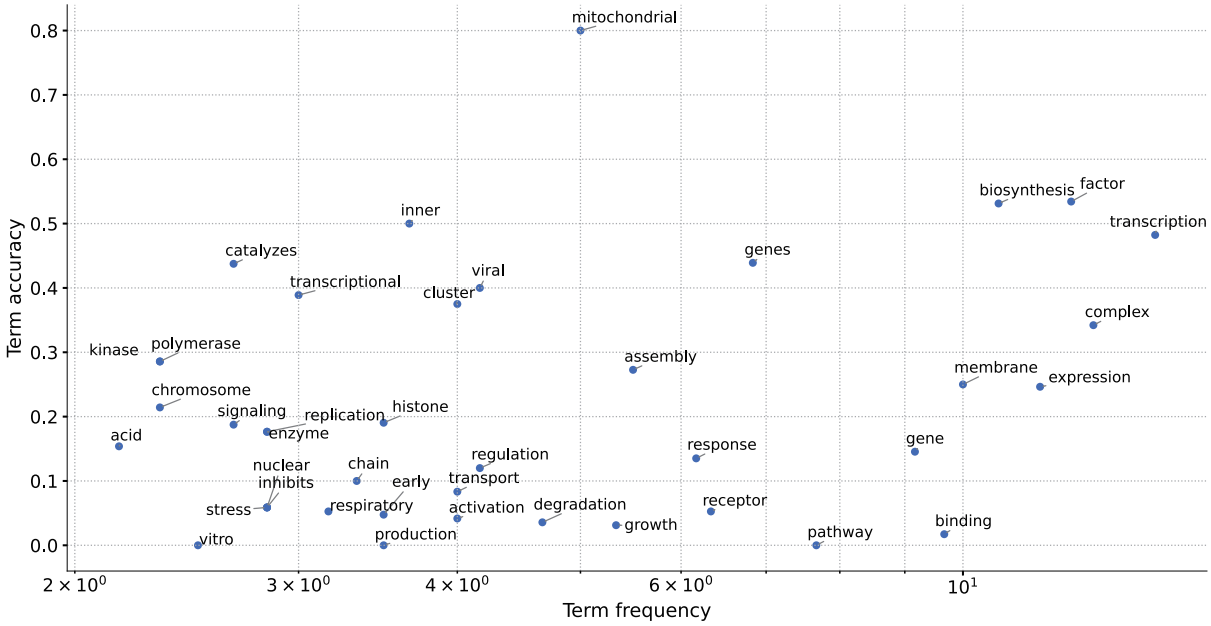

Figure 4: ProteinChat's prediction accuracy for biological terms across varying frequencies.

A.5. Experimental details of baselines

To solicit function predictions from GPT-4 using the amino acid sequence of a protein, we used the following prompt: "Given the sequence of a protein: [*a string of amino acid letters such as MARYFRRRKFCRFTAEGVQEIDYKDIATLKNYITES-GKIVPSRITGTRAKYQRQLARAIKRARYLSLLPYTDRHQ*], please describe the function of this protein."

To obtain function predictions using BLASTp, we compared each query protein sequence against a database of protein sequences from the training set, using an E-value threshold of 0.05 and the BLOSUM62 scoring matrix (Henikoff & Henikoff, 1992). We ranked the BLASTp results in descending order of bit-score and selected the top hit as the predicted match. The function annotation of this top match was then retrieved from the training set and used as the BLASTp-predicted function.

To obtain function predictions using ProtNLM, we first used ProtNLM to predict the protein name from the input amino acid sequence. Given the ProtNLM-generated name, we queried UniProt and retrieved the function annotation of the top matching entry, which we used as the ProtNLM-predicted function.

The InstructProtein model was pretrained on textual amino acid sequences and natural language data, followed by instruction tuning. We fine-tuned InstructProtein on our free-form function dataset for two epochs using the recommended hyperparameters, then performed inference on our test set to generate predictions for both free-form prediction and discrete category classification tasks. We used the following prompt: "Please describe the function of the protein [*<protein>MLPSKRVPFLFTIILFLAGLGQHTTESVL-PDCVLYPRCLITKDPCCM</protein>*]", where "<protein></protein>" is a specialized token representing the amino acid sequence.

A.6. Experimental details for specific prediction Tasks

Predicting enzyme catalytic functions involves determining which of the seven categories of chemical reactions a given enzyme can catalyze. These categories include hydrolase, oxidoreductase, lyase, transferase, ligase, isomerase, and translocase. The prompt for this prediction task was "What type of enzyme is this? Choose from [*the list of categories above*]". Similarly, predicting ligand binding entails identifying the specific ligand a protein can bind to, while predicting coenzyme-enzyme interactions focuses on determining which coenzyme interacts with a given enzyme. The prompts for these tasks are outlined in Table 1. In the biological process prediction task, the goal is to predict the biological processes in which a protein is involved, including molecule transport, DNA to mRNA transcription, amino acid biosynthesis, protein biosynthesis from mRNA molecules, lipid metabolism, tRNA processing, DNA damage response, and cell cycle regulation. Cellular component prediction involves determining the cellular localization of proteins (Consortium, 2019). While cellular localization does not directly define protein functions, it is often intrinsically linked to the roles proteins play within the cell. For example, proteins involved in energy production, such as those in the electron transport chain, are typically located within the mitochondria. We evaluated ProteinChat's ability in identifying proteins' cellular localization from six categories: cytoplasm, membrane, nucleus, secreted, mitochondrion, and plastid, using the following prompt: "What is the cellular localization of this protein? Choose from [*a list of the six categories*]".

For each of these specific prediction tasks, we developed a specialized classifier. Each classifier includes a protein encoder based on the pretrained xTrimoPGLM-1B and a classification head based on a multi-layer perceptron. Given the amino acid sequence of a protein, the protein encoder extracts representations for each amino acid. These representations are then averaged into a single vector, which is subsequently fed into the classification head to predict the class label. The classification head is a Multilayer Perceptron (MLP) with two layers. For all classification tasks, the first layer of the MLP contains 128 hidden units. The second layer's number of hidden units corresponds to the number of categories specific to the task. For each classifier, we trained two variants: 1) keeping the pretrained protein encoder fixed and only training the classification head (referred to as Classifier 1), and 2) training both the protein encoder and the classification head (referred to as Classifier 2). The weights of the MLP were initialized using the Kaiming initialization method. We used the same learning rate and optimizer as in the ProteinChat training configurations. The batch size was set to 32, and a checkpoint was saved every 2500 iterations. The checkpoint with the best performance on 300 randomly selected validation examples was then chosen. For each task, there were 1000 test proteins. The training data for the specialized classifiers was curated from the UniProtKB database. The number of training examples for the classifiers in the tasks of predicting catalytic functions, ligand binding, coenzyme-enzyme interactions, biological processes, and cellular components were 277548, 198215, 31672, 198661, and 340276 respectively.

The two Gene Ontology (GO) classifiers - DeepGOPlus (Kulmanov & Hoehndorf, 2020) and NetGO 3.0 (Wang et al., 2023) - utilize online web services to predict GO terms with rankings. A prediction is considered correct if the ground truth GO term holds the highest rank among all possible answers for the given question.

B. Related works

To better analyze, annotate, and predict protein functions, significant research has been conducted in recent years. The Critical Assessment of Function Annotation (CAFA) competition (Radivojac et al., 2013) is designed to develop machine learning models for predicting the Gene Ontology (GO) categories associated with protein functions. As of 2023, this competition has been held five times, yielding diverse solutions such as comparing unsolved sequences with known proteins, integrating multiple data sources, and applying machine learning algorithms with insights into biological processes to decipher protein functions. Notable work has focused on predicting GO functions, including DeepGOPlus (Kulmanov & Hoehndorf, 2020; Kulmanov et al., 2024) and NetGO 3.0 (Wang et al., 2023). These methods typically train separate models for each sub-ontology in GO, which encompasses molecular function ontology (MFO), biological process ontology (BPO), and cellular component ontology (CCO). Recent deep learning methods have demonstrated great efficacy in predicting

specific protein functions. These include Graph Neural Networks (Gligorijević et al., 2021), diffusion models (Watson et al., 2023), transfer learning (Lin et al., 2023a), and contrastive learning (Yu et al., 2023). These methods focus on predicting protein functions represented as discrete categories, but they are unable to predict functions described in free-form text, which typically contains more detailed information than category labels.

Multi-modal learning, particularly in image-text applications, has seen significant advancements recently. The CLIP model (Radford et al., 2021) employs contrastive learning to align image and text embeddings effectively. The BLIP-2 framework (Li et al., 2023) integrates images and text prompts to generate relevant responses using large language models. Building on BLIP-2, MiniGPT-4 (Zhu et al., 2024) enhances performance by incorporating the more powerful Llama-2 model. Additionally, LLaVA (Liu et al., 2024) combines a vision encoder with a large language model for various visual-textual tasks, including scientific question answering. In the scientific domain, multi-modal learning has gained increasing attention. MoleculeSTM (Liu et al., 2023) utilizes contrastive learning to simultaneously learn representations for chemical structures and textual descriptions of molecules. ProtST (Xu et al., 2023) employs contrastive learning and multi-modal mask prediction to align protein sequences with their textual descriptions, enabling zero-shot classification and text-protein retrieval. In contrast to ProtST, ProteinChat offers free-form protein function prediction, a feature not available in ProtST. Additionally, MultiVI (Ashuach et al., 2023) is a deep generative model that integrates multi-modal single-cell datasets, facilitating the joint analysis of chromatin accessibility and gene expression measurements.

C. Additional experiments

C.1. ProteinChat predicts novel protein functions beyond existing annotations

To assess whether ProteinChat's predictions contain novel functional insights beyond what is present in the ground truth annotations, we randomly selected a few test proteins for qualitative analysis. Fig. 6a presents the predictions of ProteinChat and BLASTp, along with the corresponding ground truth annotations, for two illustrative examples. For protein `A0A1J0HSR1`, ProteinChat predicts that it mediates fusaric acid biosynthesis. Notably, this functional role is not mentioned in the ground truth annotation, suggesting a novel prediction. To evaluate its plausibility, our domain experts conducted a literature review and bioinformatic analysis. The results support the biological validity of ProteinChat's prediction. Specifically, `A0A1J0HSR1` contains a conserved fungal transcription factor middle homology region (MHR) domain, indicating that it likely belongs to the $Zn(II)_2Cys_6$ family of transcriptional regulators (uto). Furthermore, `A0A1J0HSR1` — also known as gene *iacI* in *Pestalotiopsis fici* — is located within a 12-gene biosynthetic gene cluster (BGC) that includes two putative regulatory proteins (iacI and iacK) alongside multiple enzymatic genes (sec). This gene cluster architecture closely resembles that of the fusaric acid (FUB) cluster in *Fusarium*, which also contains two pathway-specific $Zn(II)_2Cys_6$ regulators (Brown et al., 2015). Taken together, these findings suggest that `A0A1J0HSR1` likely functions as a pathway-specific transcription factor regulating fusaric acid biosynthesis, paralleling the role of FUB cluster regulators in *Fusarium*. This novel role was neither captured in the ground truth annotation nor inferred by BLASTp.

For another protein, `Q5AUX9`, ProteinChat predicts a role in meroterpenol biosynthesis, which is not mentioned in the ground truth annotation. The ground truth describes this protein as a transcription factor regulating genes responsible for the production of DHMBA, a specific polyketide intermediate. However, literature indicates that DHMBA biosynthesis occurs as part of a larger, coordinated meroterpenoid biosynthetic system. In *Aspergillus nidulans*, the production of meroterpenoids requires the interplay of two gene clusters: one synthesizes the polyketide precursor (e.g., DHMBA), and the other performs essential tailoring modifications to produce the final meroterpenoid compound (Schroeckh et al., 2009; Nielsen et al., 2011). Crucially, these gene clusters are not regulated in isolation. Regulatory proteins such as DbaA and DbaG have been shown to coordinate gene expression across the entire dba biosynthetic cluster — the same cluster responsible for DHMBA production — indicating that regulators involved in early-stage precursor synthesis often also govern downstream steps in meroterpenoid biosynthesis (Gerke et al., 2012; Wang et al., 2021). This integrated regulatory architecture supports the idea that a transcription factor initially associated with DHMBA may also influence the broader meroterpenol biosynthetic pathway. ProteinChat's broader prediction is further supported by studies showing that such pathway-wide regulation is common in fungal secondary metabolism. Lo et al. (Lo et al., 2012) demonstrated that transcription factors in *Aspergillus nidulans* can activate entire meroterpenoid biosynthetic pathways, coordinating expression across both core and tailoring genes. Sanchez et al. (Sanchez et al., 2010) similarly showed that biosynthetic gene clusters are often controlled by regulators that function at the pathway level, rather than being restricted to individual metabolite-specific roles. These findings reinforce the plausibility of ProteinChat's functional assignment: by predicting that `Q5AUX9` regulates the full meroterpenol pathway — not just DHMBA — it mirrors known biological patterns of

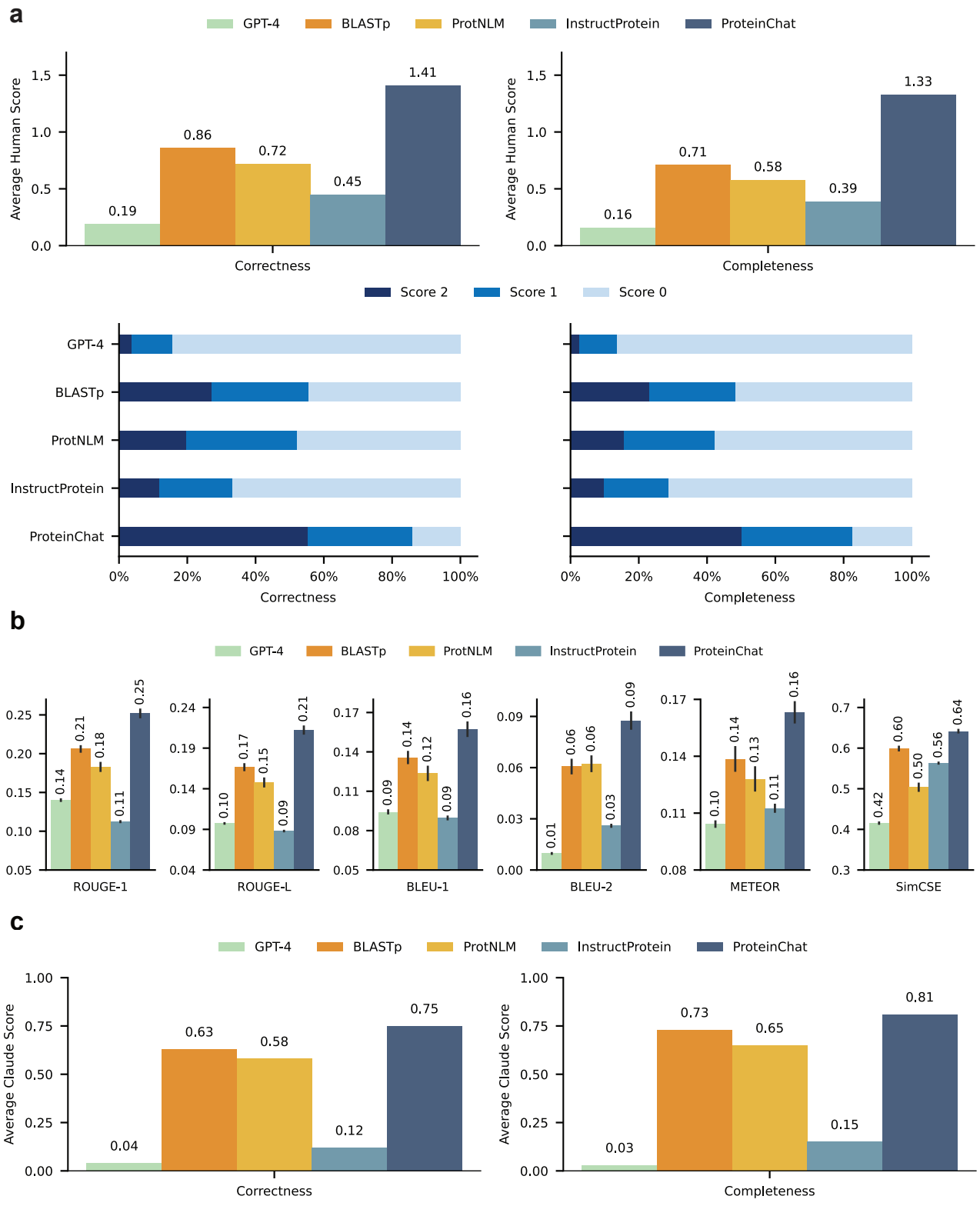

Figure 5: **ProteinChat accurately predicts protein functions expressed in textual descriptions and outperforms GPT-4, BLASTp, ProtNLM, and InstructProtein**. **a**, In the human evaluation of correctness and completeness, ProteinChat significantly outperformed all baselines in both average scores and score distributions. **b**, In automated evaluation metrics including ROUGE-1, ROUGE-L, BLEU-1, BLEU-2, METEOR, and SimCSE, ProteinChat demonstrated significantly superior performance compared to baselines. **c**, In the Claude 3.5 Sonnet evaluation, ProteinChat outperformed all baseline methods.

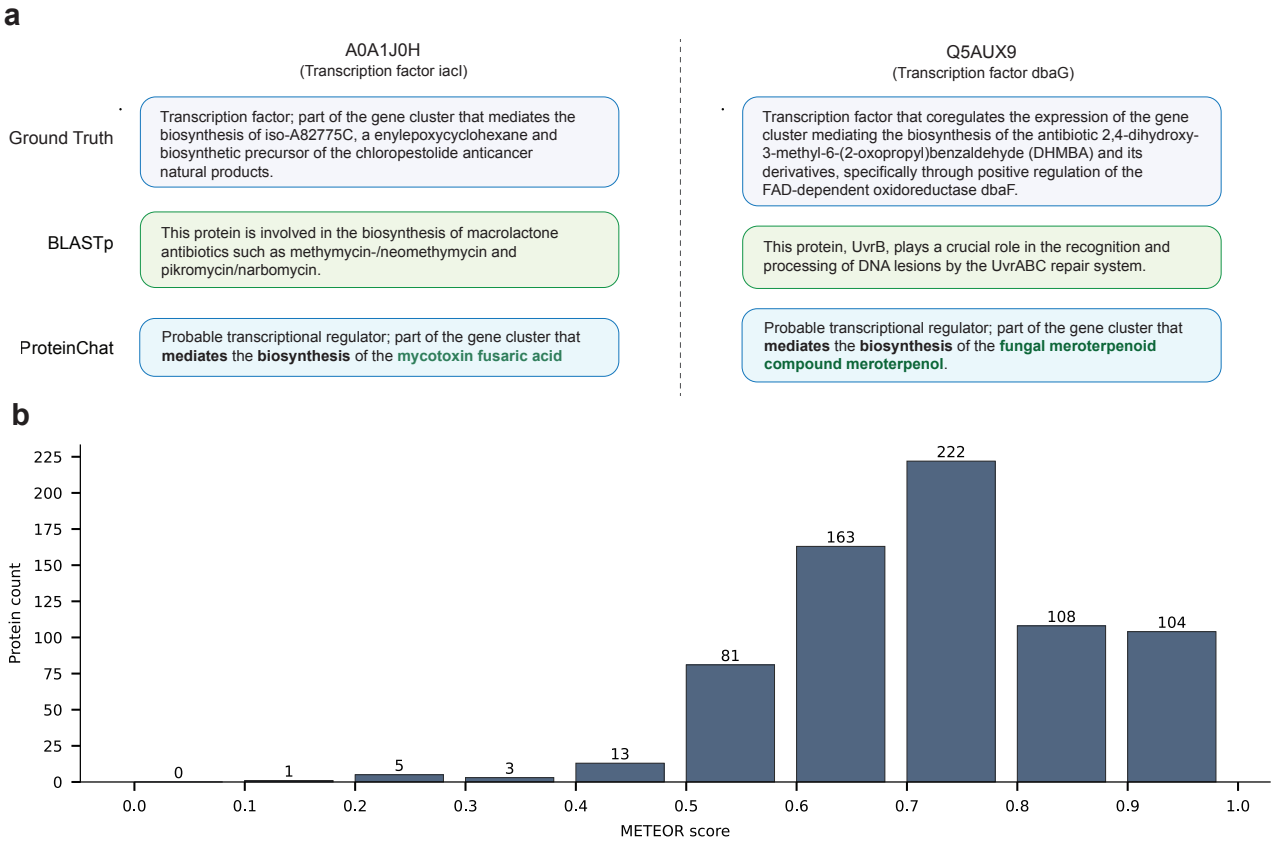

Figure 6: **ProteinChat predicts novel protein functions beyond existing annotations. a**, On two exemplar test proteins, ProteinChat predicted biologically plausible functions that are not present in the ground truth annotations, whereas BLASTp failed to identify such functions. **b**, The METEOR scores between ProteinChat's predictions on test proteins and ground truth annotations in the training data mostly fall between 0.5 and 0.9, indicating that ProteinChat generates novel information rather than memorizing training data.

pathway-level transcriptional control in fungal secondary metabolism. In contrast, BLASTp does not associate this protein with meroterpenol biosynthesis. These examples demonstrate that ProteinChat goes beyond reproducing known annotations by generating novel, biologically plausible predictions. This qualitative analysis highlights ProteinChat's potential utility in uncovering previously unannotated functions and in generating testable hypotheses to guide future experimental validation.

In addition, to further evaluate whether ProteinChat generates novel free-form function descriptions rather than memorizing its training data, we analyzed the similarity between its predictions on the test set and the function annotations seen during training. We computed the METEOR score for each test prediction against all function descriptions in the training set and recorded the highest score per prediction. As shown in Fig. 6b, the resulting scores are concentrated between 0.5 and 0.9, with very few predictions achieving scores near 1.0. This range reflects moderate similarity, suggesting that the predictions contain novel information not present in the training data. These findings demonstrate that ProteinChat is not simply memorizing its training data, but rather generalizing from it to produce novel function descriptions.

C.2. ProteinChat enables interactive and iterative predictions of protein functions

ProteinChat facilitates interactive dialogues between users and the system. After obtaining the initial predictions from ProteinChat, users can input more detailed and specific prompts to further refine and expand these predictions. Fig. 8 presents three example dialogues between ProteinChat and human users, corresponding to proteins `Q9U281`, `Q9XZG9`, and `Q9LU44` in UniProtKB. The dialogue on the left pertains to `Q9U281`, where the user inquires about the general function of this protein. ProteinChat identifies it as a histone protein involved in modulating DNA accessibility. Subsequently, the user inquires

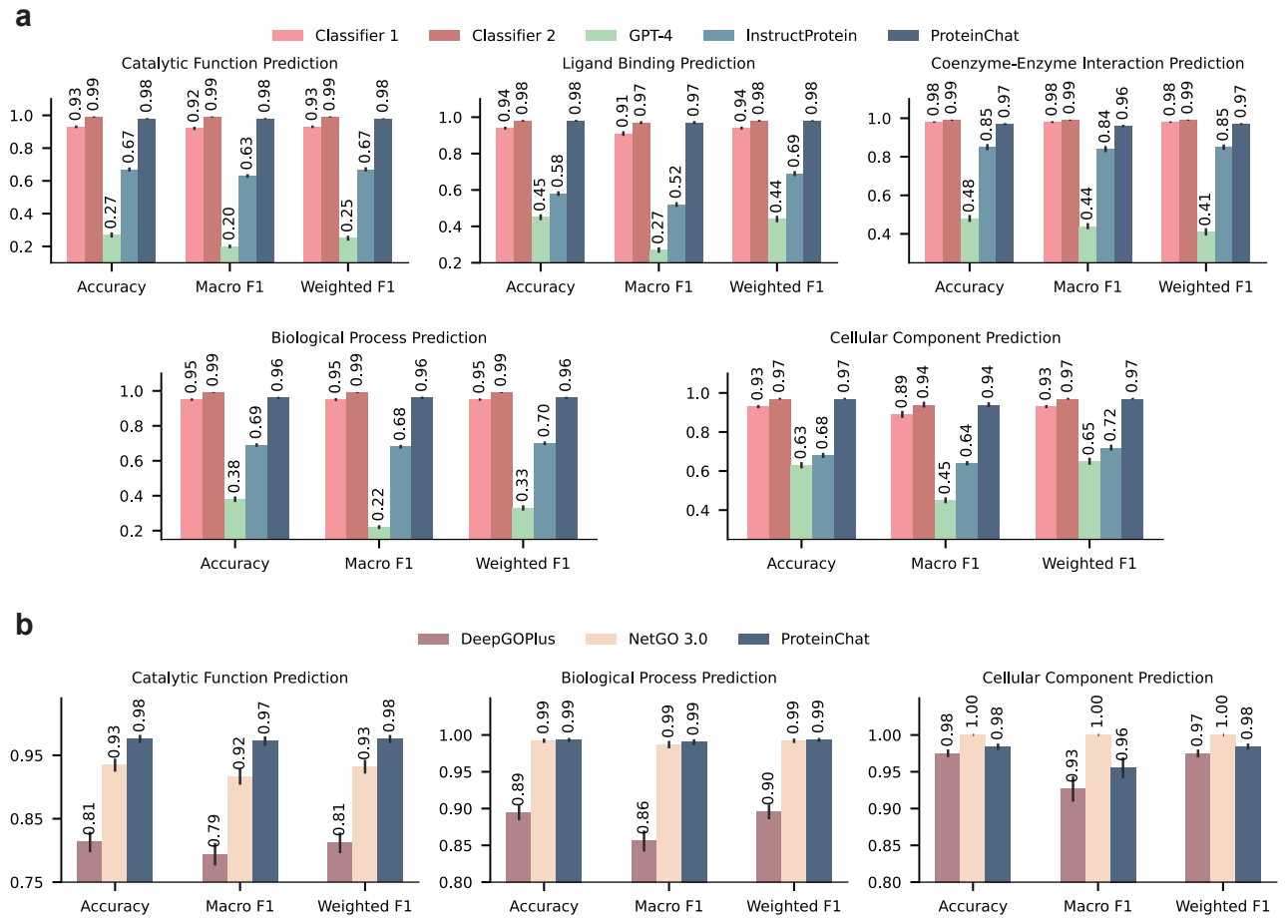

Figure 7: **ProteinChat demonstrates strong accuracy in specific-category prediction tasks. a**, In five curated tasks from UniProt — catalytic function prediction, ligand binding prediction, coenzyme–enzyme interaction prediction, biological process prediction, and cellular component prediction — where protein functions are represented as discrete categories, ProteinChat achieved significantly higher accuracy, macro F1, and weighted F1 scores compared to GPT-4 and InstructProtein, and performed on par with specialized classifiers. **b**, In predicting protein functions represented using Gene Ontology (GO) categories, ProteinChat outperformed or matched the performance of two state-of-the-art GO classifiers, DeepGOPlus and NetGO 3.0.

about the specific functions of this histone protein, and ProteinChat provides detailed predictions, highlighting the protein's roles in transcription regulation and post-translational modifications. The top right dialogue pertains to `Q9XZG9`, where ProteinChat initially predicts that the protein has antibacterial function. Based on the user's further prompt, ProteinChat then accurately predicts the protein can inhibit the growth of both Gram-positive and Gram-negative bacteria. The bottom right example focuses on `Q9LU44`. When inquired about general functions, ProteinChat predicts that the protein is involved in pre-mRNA splicing. Upon further inquiry into specific molecular functions, such as metal binding, ProteinChat predicts that the protein binds zinc ions. This dynamic interaction between ProteinChat and users facilitates continuous, in-depth analysis of the same protein, in contrast to previous methods that offer only single-shot predictions. Users can delve deeper into the specifics of protein functions, exploring intricate details and nuances that single-shot predictions might miss. This ensures that the predictions are not only more accurate but also more comprehensive, uncovering complex protein behaviors and mechanisms.

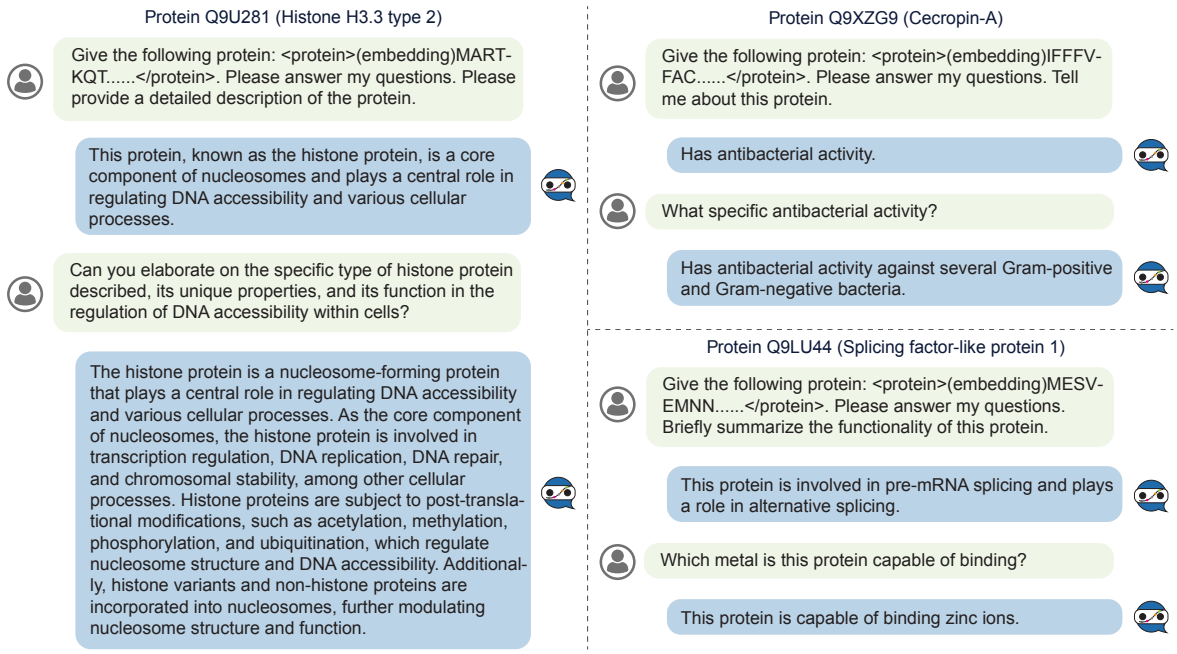

Figure 8: Interactive dialogues between ProteinChat and human users about proteins Q9U281, Q9XZG9, and Q9LU44.

C.3. Proteins with identical functions are located close to each other in the representation space of ProteinChat

To better understand how ProteinChat predicts protein functions, we visualized its learned protein representations in a 2D space using t-SNE (Van der Maaten & Hinton, 2008). For each input protein's amino acid sequence, we utilized the trained xTrimoPGLM (Chen et al., 2025) protein encoder and the trained adaptor in ProteinChat to extract a representation vector for each amino acid. We then computed the overall representation of the entire protein by averaging the representations of all the amino acids. We projected the protein representation vectors into a 2D space using t-SNE (Van der Maaten & Hinton, 2008) for visualization. Fig. 9 presents a visualization of all $n = 20,426$ human proteins from the Swiss-Prot dataset. Each dot in the figure represents a protein. In Fig. 9a, we have highlighted proteins with ground truth labels for three cellular localizations: nucleus ($n = 5,617$), secreted ($n = 2,113$), and mitochondrion ($n = 1,309$). As observed, proteins with the same cellular localization are clustered together in the representation space. Similar patterns can be observed in Fig. 9b-d.

In addition, we quantitatively assessed the quality of clustering in the learned protein embedding space. Specifically, we computed intra-cluster and inter-cluster distances among protein embeddings based on functional or subcellular annotations, such as cellular compartments (e.g., nucleus, secreted, mitochondrial) and ligand-binding categories (e.g., nucleotide-binding, zinc-binding). For each annotation category, we calculated the average pairwise distance between embeddings within the same group (intra-cluster) and between different groups (inter-cluster).

The results in Table 3 show a consistent pattern: proteins sharing the same annotation exhibit substantially lower intra-cluster distances than inter-cluster distances. For example, proteins localized to the nucleus have an average intra-cluster distance of 227.79, compared to an inter-cluster distance of 246.68 when paired with secreted proteins. Similarly, nucleotide- and zinc-binding proteins show intra-cluster distances ranging from 212.68 to 226.62, which are markedly smaller than the inter-cluster distances of 232.58–235.57. These results provide quantitative evidence that ProteinChat's learned embeddings effectively capture biologically meaningful patterns, supporting our observation from the t-SNE visualizations. The consistent separation of proteins by function or localization in the embedding space underscores the model's capacity to encode relevant biological semantics.

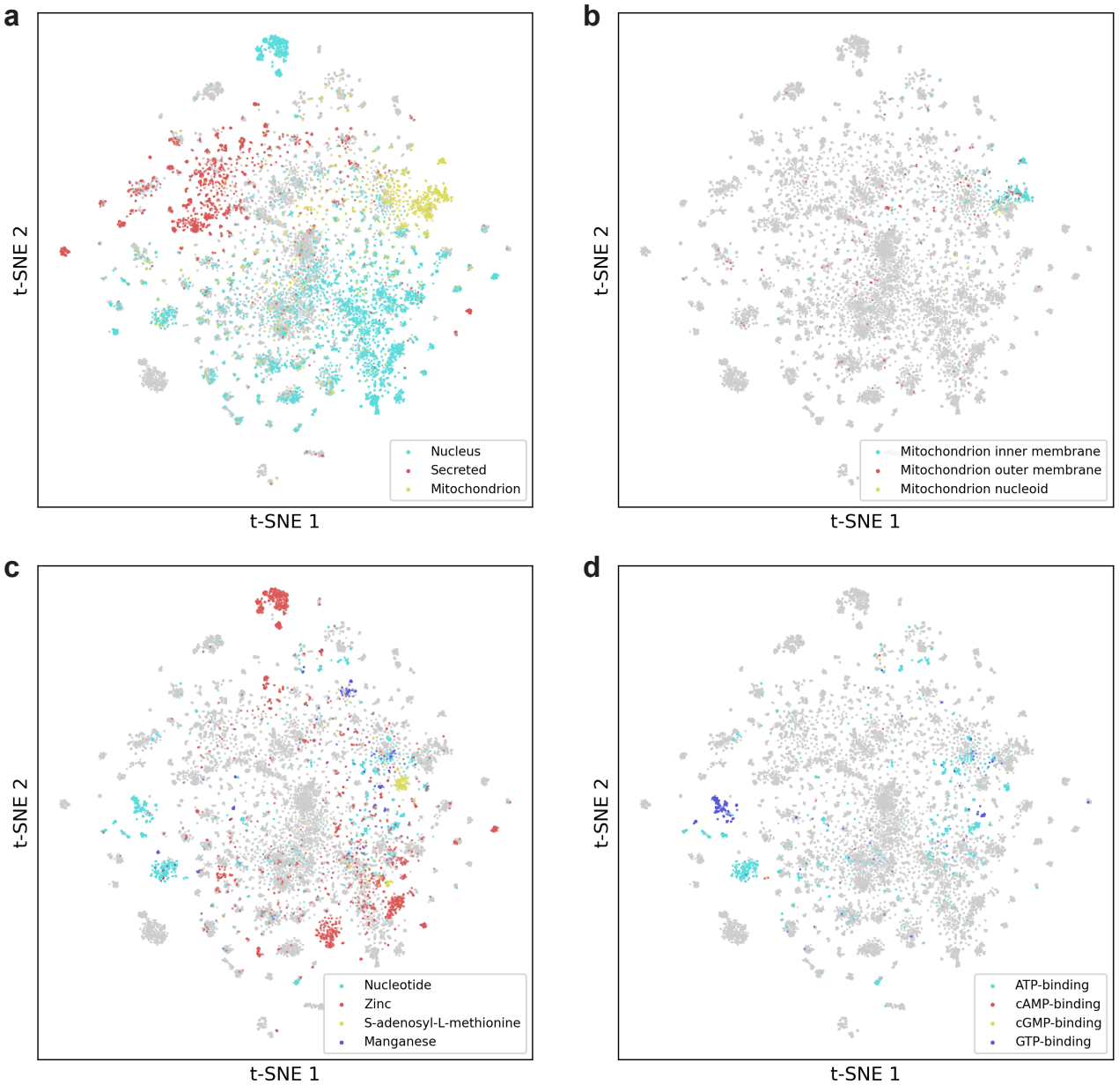

Figure 9: **t-SNE visualization of protein representations extracted by the protein encoder and adaptor of ProteinChat.** **a**, Proteins located in three cellular locations, including nucleus, secreted, and mitochondrion, are highlighted. **b**, Proteins located in three mitochondrial components - inner membrane, outer membrane, and nucleoid - are highlighted. **c**, Proteins that bind with four ligands - nucleotide, zinc, S-adenosyl-L-methionine, and manganese - are highlighted. **d**, Proteins binding with ATP, cAMP, cGMP, and GTP are highlighted.

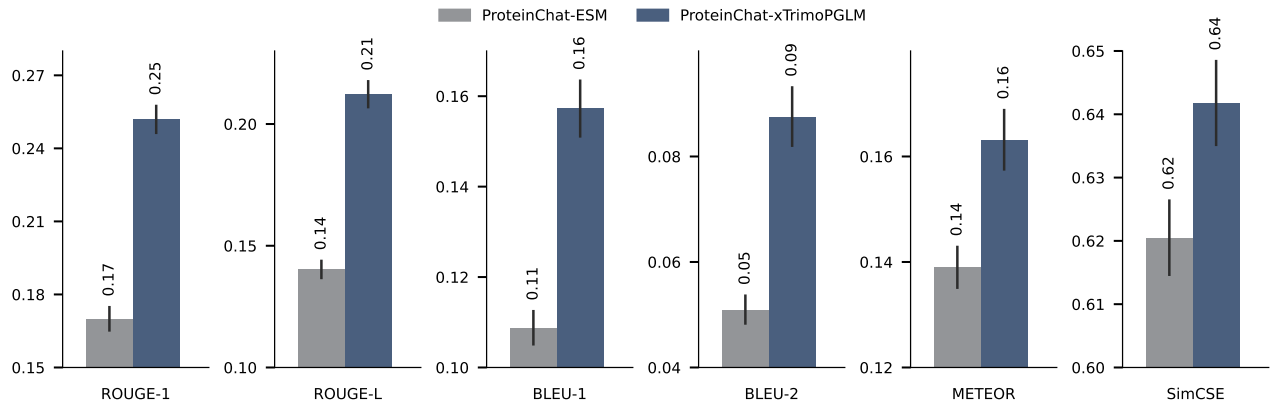

Figure 10: **Ablation study on the protein encoder.** Using xTrimoPGLM as the protein encoder yielded better performance in free-form function prediction compared to ESM-2.

C.4. Ablation study on protein encoder

To assess the impact of the protein sequence encoder on ProteinChat's performance, we conducted an ablation study comparing xTrimoPGLM (Chen et al., 2025) and ESM-2 (650M) (Lin et al., 2023b) within our framework. Specifically, we replaced xTrimoPGLM with ESM-2 and evaluated both variants on free-form protein function prediction. As shown in Figure 10, the model using xTrimoPGLM consistently outperforms the ESM-2 variant across multiple evaluation metrics, including ROUGE-1 and SimCSE similarity.

We attribute the superior performance of xTrimoPGLM to its dual-objective pretraining strategy, which combines masked language modeling (MLM) and generative language modeling (GLM). This approach enables xTrimoPGLM to learn both bidirectional representations and autoregressive generation capabilities — properties that are essential for modeling long-range dependencies and nuanced contextual patterns in protein sequences. In contrast, ESM-2 adopts an encoder-only architecture trained solely with an MLM objective, which is less suited for capturing sequential dependencies and global semantic coherence. This limitation reduces its ability to model the full functional landscape of protein sequences, especially in tasks like free-form function prediction.

C.5. Robustness to incomplete category lists

In practical applications of protein function prediction, the list of candidate function categories provided to a model may be incomplete. For example, certain relevant categories may be missing due to annotation gaps or upstream errors in data processing. Therefore, it is important for a model like ProteinChat not only to choose the correct category when it is available but also to recognize when the appropriate category is absent from the provided options. Such robustness is critical for real-world usability, especially in exploratory or low-confidence settings. To assess this capability, we conducted a controlled experiment to evaluate ProteinChat's behavior when the ground truth category is deliberately excluded from the prompt. We modified the prompt to include the instruction: "If you believe the provided list does not contain the correct category, output 'Other'." This setup allows us to test whether ProteinChat can accurately detect the absence of the correct category and appropriately respond. We first randomly selected 1,000 proteins and removed the ground truth category from the list provided in the prompt. ProteinChat responded with "Other" in 874 cases (87.4%), demonstrating a strong ability to recognize when the correct category is missing. To evaluate whether ProteinChat would erroneously respond with "Other" when the correct category is present, we conducted a complementary experiment using another 1,000 randomly selected proteins, this time ensuring that the ground truth category was included in the list. In 878 cases (87.8%), ProteinChat correctly selected the ground truth category instead of "Other", confirming that it does not mistakenly reject valid options. These results indicate that ProteinChat is robust to prompt-level noise and can reliably identify when the correct category is missing or present. This behavior is important for maintaining prediction reliability in real-world scenarios where user-provided function lists may be incomplete.

D. Additional discussion

ProteinChat illustrates two important concepts. Firstly, the fundamental language of biology - amino acid sequences - encodes highly rich information about underlying biological processes. This information is both computable and predictive, suggesting that this language can be harnessed to develop powerful predictive models in other areas of biology, as demonstrated by ProteinChat. Secondly, achieving a balance is crucial when designing deep learning models for biological applications. While highly specialized models like DeepGo or NetGo are effective in specific tasks, they may overlook the complex, multi-tasking nature of proteins that are involved in multiple biological pathways. On the other hand, overly generalized models, such as GPT-4, might lack the precision needed for accurate, domain-specific predictions. ProteinChat strikes a balance between these extremes, offering broad generalization across proteomics while maintaining high accuracy and specificity, as demonstrated in Fig. 5 and 7.

ProteinChat is designed to minimize the need for continuous user training while allowing for periodic updates and enhancements by us, the developers. For example, we plan to integrate more advanced versions of Llama (e.g., Llama-3 (Meta, 2024)) as the textual LLM component of ProteinChat, improving the quality of human-like interactions. Additionally, incorporating newer versions of xTrimoPGLM will further enhance ProteinChat's accuracy and specificity. These planned improvements will ensure that ProteinChat remains both competitive and up-to-date. Furthermore, ProteinChat's versatility enables seamless integration with other deep-learning models, such as those based on structure prediction like AlphaFold (Jumper et al., 2021), allowing it to predict the functions of proteins in the context of their 3D structures.

Some predictions made by ProteinChat, currently labeled as incorrect by human experts, may actually uncover previously unidentified properties and functions of these proteins. As a result, the scores we assigned to ProteinChat could potentially be even higher. More importantly, predictions deemed incorrect might actually offer new insights or hypotheses that warrant further experimental validation. For many proteins, only a portion of their amino acid sequences have been fully understood, with the remainder still elusive and sometimes labeled as "junk" - sequences that seemingly do not contribute significantly to the protein's main function (Lovell, 2003). ProteinChat has the potential to shed light on these currently uninterpretable sequences. Additionally, large portions of proteins can consist of disordered segments - sequences that do not fold into a stable structure (Van Der Lee et al., 2014). Historically, these segments have often been truncated in structural and biophysical studies, leading to incomplete characterizations. However, recent research indicates that these disordered segments are crucial for the phase separation of proteins into specific cellular compartments, where they carry out their functions (Hyman et al., 2014). ProteinChat, which can analyze the entire protein sequence, could be particularly effective in interpreting these disordered segments and predicting their phase-separating characteristics. This capability may already be reflected in ProteinChat's predictions related to cellular compartmentalization.

Table 1: Prompts linked to keywords and the number of curated triplets for each keyword.

Catalytic function

Prompt: *What type of enzyme is this? Choose one from the following options: hydrolase, oxidoreductase, lyase, transferase, ligase, isomerase, and translocase.*

Function category	Number of triplets	UniProtKB keyword	GO term
Transferase	98540	KW-0808	GO:0016740
Hydrolase	65580	KW-0378	GO:0016787
Oxidoreductase	36864	KW-0560	GO:0016491
Ligase	29379	KW-0436	GO:0016874
Lyase	26546	KW-0456	GO:0016829
Isomerase	16283	KW-0413	GO:0016853
Translocase	14708	KW-1278	-

Ligand binding

Prompt: *What ligand can this protein bind to? Choose one from the following options: magnesium, nucleotide-binding, zinc, iron, S-adenosyl-L-methionine, and manganese.*

Function category	Number of triplets	UniProtKB keyword	GO term
Nucleotide-binding	101082	KW-0547	GO:0000166
Magnesium	46675	KW-0460	-
Zinc	41464	KW-0862	-
Iron	29555	KW-0408	-
S-adenosyl-L-methionine	17332	KW-0949	-
Manganese	12067	KW-0464	-

Coenzyme-enzyme interaction

Prompt: *What coenzyme does this enzyme interact with? Choose one from the following options: nicotinamide adenine dinucleotide (NAD) and nicotinamide adenine dinucleotide phosphate (NADP).*

Function category	Number of triplets	UniProtKB keyword	GO term
Nicotinamide adenine dinucleotide (NAD)	21502	KW-0520	-
Nicotinamide adenine dinucleotide phosphate (NADP)	15102	KW-0521	-

Biological process

Prompt: *What biological process is this protein involved in? Choose one from the following options: molecule transport, DNA to mRNA transcription, amino acid biosynthesis, protein biosynthesis from mRNA molecules, lipid metabolism, tRNA processing, DNA damage response, and cell cycle regulation.*

Function category	Number of triplets	UniProtKB keyword	GO term
Molecule transport	58648	KW-0813	-
DNA to mRNA transcription	32127	KW-0804	-
Amino acid biosynthesis	26272	KW-0028	GO:0008652
Protein biosynthesis from mRNA molecules	26063	KW-0648	GO:0006412
Lipid metabolism	16282	KW-0443	GO:0006629
tRNA processing	15380	KW-0819	GO:0008033
DNA damage response	14565	KW-0227	GO:0006974
Cell cycle regulation	14474	KW-0131	GO:0007049

Cellular component

Prompt: *What is the cellular localization of this protein? Choose one from the following options: cytoplasm, membrane, nucleus, secreted, mitochondrion, and plastid.*

Function category	Number of triplets	UniProtKB keyword	GO term
Cytoplasm	165882	KW-0963	GO:0005737
Membrane	116756	KW-0472	GO:0016020
Nucleus	41431	KW-0539	GO:0005634
Secreted	32360	KW-0964	GO:0005576
Mitochondrion	17206	KW-0496	GO:0005739
Plastid	15990	KW-0934	GO:0009536

Table 2: Rubric used for both human expert evaluation and Claude 3.5 Sonnet assessment of predicted protein functions.

Score	Correctness score description	Completeness score description
2	Prediction is mostly or fully accurate.	Prediction captures all key aspects of the ground truth function.
1	Prediction captures the correct functional category but includes substantial inaccuracies (e.g., correct substrate but incorrect function).	Prediction includes some relevant functional information but is partially incomplete.
0	Prediction is entirely incorrect or lacks relevant biological content.	Prediction does not include any relevant functional information.

Table 3: Intra- and inter-cluster distances and permutation test results.

		Intra-cluster	Inter-cluster Comparison		
		Distance	Comparison	Distance	p-value
Cellular Localization	Nucleus (5617)	227.79	Nucleus vs Secreted	246.68	0.0099
	Secreted (2113)	236.60	Nucleus vs Mitochondrion	252.16	0.0099
	Mitochondrion (1309)	245.97	Secreted vs Mitochondrion	258.61	0.0099
Mitochondrion Locations	Inner membrane (332)	243.64	Inner vs Outer membrane	255.39	0.0099
	Outer membrane (154)	232.29	Inner vs Nucleoid	248.93	0.0099
	Nucleoid (19)	216.10	Outer vs Nucleoid	234.39	0.3366
Binding Ligands	Nucleotide (1985)	226.62	Nucleotide vs Zinc	233.65	0.0099
	Zinc (2449)	219.12	Nucleotide vs S-adenosyl-L-methionine	234.60	0.0099
	S-adenosyl-L-methionine (188)	212.68	Nucleotide vs Manganese	232.58	0.0099
	Manganese (222)	223.19	Zinc vs S-adenosyl-L-methionine	233.11	0.0099
			Zinc vs Manganese	235.57	0.0099
			S-adenosyl-L-methionine vs Manganese	233.09	0.0099
ATP/cAMP/cGMP/GTP Binding	ATP-binding (1393)	224.44	ATP vs cAMP-binding	223.16	0.5842
	cAMP-binding (17)	180.55	ATP vs cGMP-binding	219.80	0.7723
	cGMP-binding (9)	162.75	ATP vs GTP-binding	231.93	0.0099
	GTP-binding (357)	209.94	cAMP vs cGMP-binding	182.46	0.1287
			cAMP vs GTP-binding	229.95	0.0099
			cGMP vs GTP-binding	227.95	0.0297