Benchmarking Biomolecular Foundation Models for Cross-Modal Genomics-Proteomics

Joseph G. Wakim[*], **Vinayak Gupta**[*], **Jose Manuel Marti**, **Jonathan E. Allen**, **Brian Bartoldson**, **Bhavya Kailkhura**[†]

Lawrence Livermore National Laboratory, Livermore, CA, USA
[*]Equal contribution; [†]Corresponding author: kailkhura1@llnl.gov

Abstract

Genome and protein foundation models (gFMs and pFMs) are designed to encode intricate patterns in nucleotide and amino acid sequences, respectively. While existing models are often trained on a single data modality, recent findings suggest that joint training on both nucleotide and amino acid sequences can improve a model's versatility. However, due to the limited availability of multimodal datasets, training and evaluating models across modalities pose a significant challenge. To address this gap, we introduce GENO-PROT, a new benchmark formed from matched gene and protein sequences mapped to nine attributes that capture structural, functional, and biomedical characteristics of the human proteome. The benchmark poses questions, such as: *Do proteins A and B interact? Do they co-localize in the same subcellular region? Do they bind a common ligand?* Using GENO-PROT, we evaluate leading gFMs, pFMs, and multimodal foundation models, along with ensembles that combine them. Our comprehensive analysis shows that, across all prediction tasks, ensembles tend to match or outperform the individual models they contain. Ultimately, GENO-PROT serves as a vital tool for model assessment, highlighting the predictive benefits of integrating both genomic and proteomic data. The code for this benchmark is available at https://github.com/LLNL/geno-prot.

Biomolecular foundation models (bioFMs) are trained on biological sequences (*e.g.*, proteins, DNA, and RNA) and have become powerful tools for scientists working on personalized medicine [1, 2, 3], drug discovery [4, 5], and protein engineering [6, 7]. Like traditional language models, bioFMs represent sequences with tokens. For example, they may tokenize DNA sequences on a per-nucleotide basis or protein sequences on a per-amino-acid basis. These tokens are later used as input to a self-attention model [8].

The majority of existing bioFMs are unimodal, meaning they are specifically trained on one type of biological data modality (*e.g.*, either DNA or protein sequences, but not both) [9, 10]. However, the central dogma of molecular biology links DNA and proteins, and jointly training on both sequence types can yield valuable insights [11, 12]. Assessing how well bioFMs capture the central dogma is difficult, as most training and evaluation datasets are also unimodal. Meanwhile, protein-to-DNA translation is unreliable due to codon degeneracy and ambiguity regarding introns. DNA-to-protein translation is also challenging; sequencing technologies generate short, randomly positioned DNA reads that may not align with complete protein-coding regions.

To address this gap, we present GENO-PROT, a benchmark suite for evaluating genome foundation models (gFMs) and protein foundation models (pFMs) in a cross-modal setting. Using sequences and annotations from RefSeq [13] and UniProt [14], GENO-PROT matches each gene's complete DNA sequence (from the annotated start codon to the stop codon, including introns) with its corresponding protein sequence. GENO-PROT includes nine modality-flexible downstream prediction tasks, relevant

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: 2nd Workshop on Multimodal Foundation Models and Large Language Models for Life Sciences.

to DNA sequences, protein sequences, or both. The prediction tasks are organized into three groups: (i) **structure and localization**, (ii) **functional relationships**, and (iii) **biomedical associations**. Using GENO-PROT, we evaluate bioFMs ranging from 6.6 million to seven billion parameters in size. We also compare these models to general-purpose large language models (LLMs), which likely exceed one trillion parameters in size. The key findings from our evaluations are as follows:

(1) Ensembles of gFMs and pFMs tend to exceed or match the performance of the individual models they contain. This result suggests that ensembles of bioFMs offer a robust approach for generating biological predictions.

(2) On our benchmarks, gFMs perform poorly compared to even much smaller pFMs. This observation is consistent with the findings of Ref. [15]. In GENO-PROT, the results may be influenced by the protein-relevant prediction tasks; on more gene-centric tasks, gFMs may still achieve competitive performance.

(3) State-of-the-art language models, like GPT-4.1 and GPT-4o, are not well suited for the biological tasks in GENO-PROT without further adaptation.

Overall, GENO-PROT serves as a valuable resource for developing cross-modal bioFMs and rigorously evaluating claims related to the central dogma of molecular biology.

The GENO-PROT Benchmark

The GENO-PROT benchmark organizes human proteins into positive and negative classes to support comparisons of unimodal bioFMs, multimodal bioFMs, and general-purpose LLMs. In all cases, the prediction task is to distinguish positive examples (labeled with `1`) from negative examples (labeled with `0`). Positive and negative classes are assigned based on nine attributes, described below. The attributes are organized into three thematic areas.

Structure and Localization

(1) **Shared Pfam Label.** Proteins are labeled based on whether they share a common Pfam domain [16]. We load the Pfam labels for human proteins with available sequences [14]. The proteins are organized into pairs. Protein pairs with shared Pfam domains are assigned to the positive class, while others are assigned to the negative class. To assess the classification performance of general-purpose LLMs, we ask: *"Do proteins A and B share a common Pfam label?"*

(2) **Shared Subcellular Location.** Proteins are annotated with their subcellular locations. Pairs of proteins that share at least one location are assigned to the positive class, while others are assigned to the negative class. We ask general-purpose LLMs: *"Do proteins A and B reside in at least one common subcellular location?"*

(3) **Shared Anatomical Location.** Proteins are annotated with their anatomical expression sites, such as organs and tissues [14]. We parse annotations for anatomical locations using named entity recognition (NER). We use the spaCy model for NER, trained on the BioNLP13CG corpus [17]. Protein pairs that share at least one anatomical location are assigned to the positive class, while others are assigned to the negative class. We ask general-purpose LLMs: *"Do proteins A and B reside in at least one common anatomical location?"*

Functional Relationships

(4) **Protein Function Similarity.** Protein function is represented using Gene Ontology (GO) annotations and GO2Vec embeddings [14, 18]. Each protein is represented by a set of GO embeddings. To quantify functional similarity between proteins, we apply a variant of the modified Hausdorff distance (MHD) to their GO embeddings, following Ref. [18]. The form of the MHD that we use leverages cosine similarities, such that greater values indicate more similar protein pairs. Protein pairs with MHDs above the median value are assigned to the positive class, while others are assigned to the negative class. We ask general-purpose LLMs: *"Do proteins A and B share similar functions?"*

(5) **Protein-Protein Interaction.** Protein-protein interactions are reported and scored in the STRING database [19]. Protein pairs with reported interactions are assigned to the positive

class, while others are assigned to the negative class. We ask general-purpose LLMs: *"Do proteins A and B interact?"*

(6) **Shared Biochemical Pathway.** Proteins are annotated with associated biochemical pathways in Reactome [20]. We isolate proteins associated with at least one biochemical pathway. Pairs of proteins sharing a pathway are assigned to the positive class, while others are assigned to the negative class. We ask general-purpose LLMs: *"Are proteins A and B involved in at least one common biochemical pathway?"*

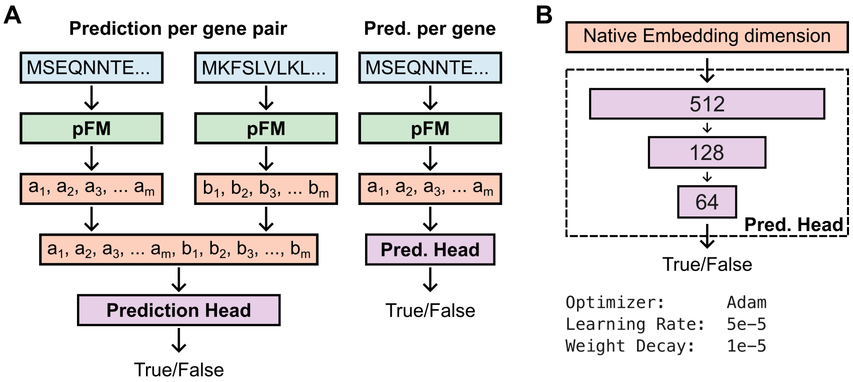
Biomedical Associations

(7) **Shared Drug Binding.** Protein-ligand interactions (PLIs) are reported in ChEMBL [21]. Heinzke *et al.* extract clinically relevant PLIs from the database [22]. Based on the filtered PLIs, we assign protein pairs to the positive class if they share at least one common ligand and to the negative class if they do not. Negative examples are restricted to proteins with at least one known interaction. We ask general-purpose LLMs: *"Do proteins A and B bind at least one common ligand?"*

(8) **Shared Disease Implication.** Mendelian Inheritance in Man (MIM) codes provide disease annotations for proteins [23]. We isolate proteins associated with at least one MIM code [14], then assign pairs of proteins to the positive class if they share at least one MIM code and to the negative class if they do not. We ask general-purpose LLMs: *"Are proteins A and B implicated in at least one common disease?"*

(9) **Melanoma Gene Dependency.** Using CRISPR knockout screens, gene dependency scores are obtained for melanoma cancer cell lines [24, 25]. Each dependency score indicates the relative fitness of tumor cells before and after a gene is knocked out. More negative scores indicate a stronger dependency on the gene. We binarize the dependency scores according to a threshold value of -0.5. Essential genes (with scores below the threshold) are assigned to the positive class, while less essential genes (with scores at or above the threshold) are assigned to the negative class. We ask general-purpose LLMs: *"Are melanoma tumor cells strongly dependent on protein A?"*

Most prediction tasks involve pairs of proteins. For these tasks, we concatenate embeddings for the proteins to represent the pair. For all tasks and bioFMs, we train separate fully connected prediction heads to generate binary outcomes (see Fig. 1). To prevent data leakage, we partition genes into non-overlapping but functionally equivalent training and evaluation folds. We train the prediction heads using the training folds and score model performance with the evaluation folds (see Sec. B.2 in the Appendix for more details). All folds of the datasets include a balanced number of positive and negative cases, sampled from the available examples.


A
Prediction per gene pair
Pred. per gene
MSEQNNTE...
MKFSLVLKL...
MSEQNNTE...
pFM
pFM
pFM
$a_1, a_2, a_3, \dots a_m$
$b_1, b_2, b_3, \dots b_m$
$a_1, a_2, a_3, \dots a_m$
$a_1, a_2, a_3, \dots a_m, b_1, b_2, b_3, \dots, b_m$
Pred. Head
Prediction Head
True/False
True/False
B
Native Embedding dimension
512
128
64
Pred. Head
True/False

```
Optimizer:      Adam
Learning Rate:  5e-5
Weight Decay:   1e-5
```

Figure 1: **Prediction pipeline. (A)** BioFMs embed gene or protein sequences, then embeddings are concatenated to represent protein pairs. For each embedding type and prediction task, a prediction head is prepared to generate binary outcomes. **(B)** The prediction head includes three dense hidden layers with 512, 128, and 64 nodes, followed by a single-node output layer. The prediction head is trained for 50 epochs with early stopping (patience=5).

Model Evaluations

We evaluate the performance of individual bioFMs, multimodal ensembles, and general-purpose LLMs using GENO-PROT. Our comparison includes prominent gFMs and pFMs from 2023 to 2025, many claiming state-of-the-art performance. The bioFMs included in our study are listed below:

- **gFMs:** hyena-160k, hyena-450k, and hyena-1m variants of HyenaDNA [9]; Evo [26]; evo2-7b variant of Evo2 [27]
- **pFMs:** Enzyme Commission (EC) number, GO, and Pfam variants of ProteInfer [28]; esmc-300m and esmc-600m variants of Evolutionary Scale Modeling Cambrian (ESMC) [29]
- **Multimodal bioFM:** LucaOne [30]

Details about these models are provided in Sec. C.1 of the Appendix. We form ensembles from Evo2, esmc-600m, all ProteInfer variants, and LucaOne. To mitigate biases, we use the same set of models in all ensembles. We implement five ensembling strategies: (1) S-ENSEM computes the mean of softmax probabilities; (2) H-ENSEM uses majority class voting; (3) W-ENSEM averages probabilities, weighting each model by its accuracy on a validation fold; (4) Med-ENSEM takes the median probability across models; and (5) Max-ENSEM selects the maximum probability per class. To provide a baseline, we evaluate GPT-4.1 and GPT-4o in a zero-shot setting, framing prediction tasks as True/False questions. The system prompt is shown in Listing C.1.

Results and Discussion

We plot the performance of gFMs, pFMs, and ensembles in Fig. 2.

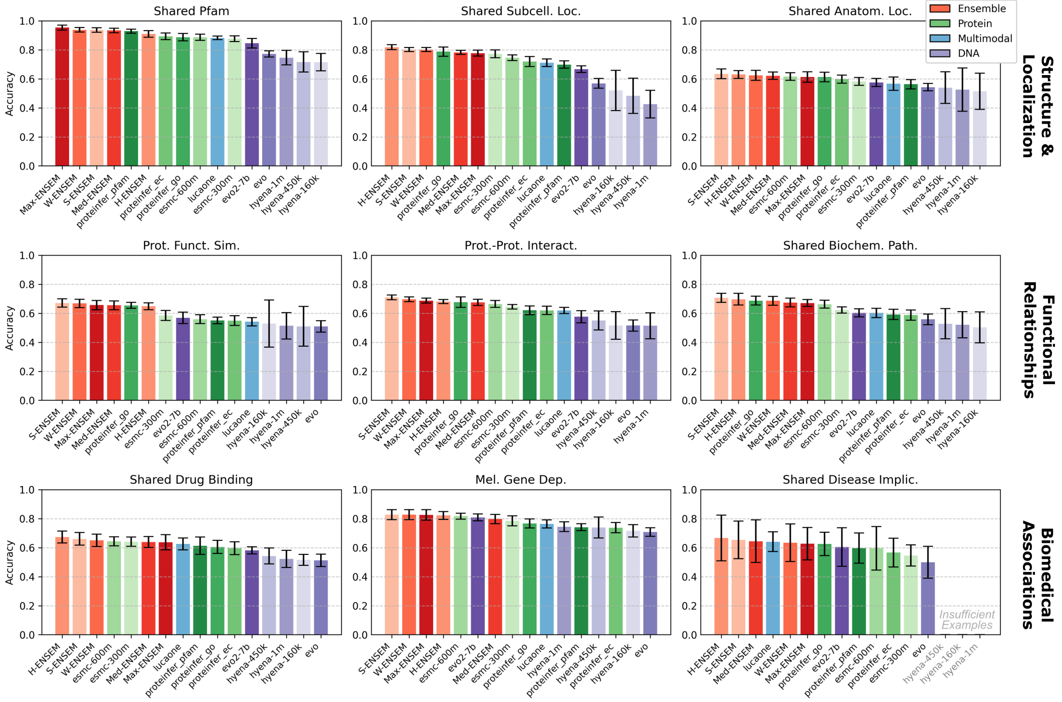


Figure 2: **Performance of models on prediction tasks.** Bars indicate average accuracy on the associated prediction task across ten evaluation samples. Error bars indicate standard deviation in accuracy across the evaluation samples. The performance of each model on individual evaluation samples is plotted in Fig. 7 in the Appendix. Bars are colored based on the class of foundation models they represent, with tones corresponding to specific models. We exclude Hyena models from the *Shared Disease Implication* evaluation due to a high frequency of out-of-memory errors observed with the examples.

The results suggest that no single model dominates all tasks, but pFMs generally outperform gFMs on most GENO-PROT prediction tasks. For the "melanoma gene dependency" prediction task, top gFMs and pFMs perform comparably. Ensembles consistently match or outperform the individual models they contain, though improvements are minor. However, the reliable performance of ensembles across the prediction tasks suggests that combining predictions from several bioFMs can lead to more robust results.

We then evaluate the zero-shot performance of GPT-4.1 and GPT-4o on the biological prediction tasks in GENO-PROT. To do so, we represent the tasks using natural language prompts, specifying proteins with their UniProt identifiers or sequences. The performance of the LLMs on the prediction tasks is reported in Fig. 3.

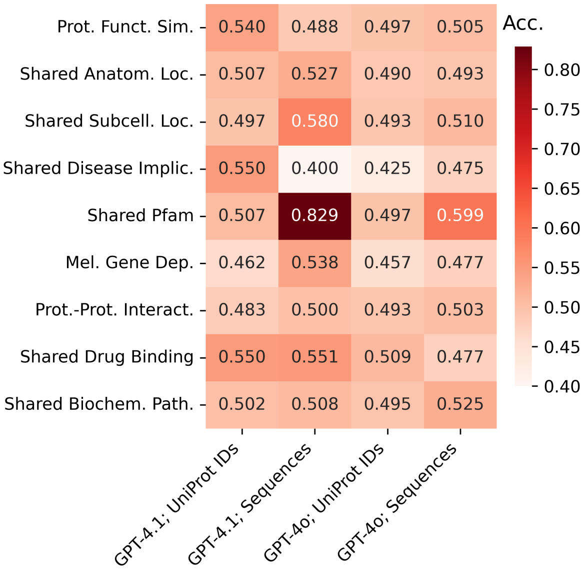


Figure 3: **LLM performance.** Zero-shot accuracies of GPT-4.1 and GPT-4o on GENO-PROT prediction tasks.

Without fine-tuning, LLMs achieve near-random performance on most prediction tasks. One notable exception is the 82.9% accuracy of GPT-4.1 with amino acid sequences on the "shared Pfam label" prediction task; this unexpected result warrants further investigation. However, more generally, the results suggest that LLMs should be used in conjunction with bioFMs for biomolecular predictions. This combined approach has already shown promise, as demonstrated in Refs. [31, 32].

Summary

Existing bioFM benchmarks are primarily unimodal, limiting head-to-head comparisons of gFMs and pFMs. In response, we present GENO-PROT, a cross-modal dataset of matched DNA and protein sequences mapped to nine biologically relevant prediction tasks. With GENO-PROT, we compare gFMs, pFMs, and ensembles that combine them. We find that pFMs generally outperform gFMs on our prediction tasks, while multimodal ensembles consistently match or exceed the performance of the individual models they contain. The results suggest that ensembles may enhance robustness, removing the need for model selection. We also translate prediction tasks into natural language prompts to evaluate the zero-shot performance of GPT-4.1 and GPT-4o. We find that bioFMs typically outperform the general-purpose LLMs. Overall, using our novel benchmark for comparing bioFMs, we highlight the value of multimodal ensembles in enhancing biological predictions.

Acknowledgments and Disclosure of Funding

This manuscript has been authored by Lawrence Livermore National Security, LLC under Contract No. DE-AC52-07NA27344 with the U.S. Department of Energy. This material is based upon work supported by the Department of Energy, Office of Science, Office of Advance Scientific Computing Research. The United States Government retains, and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes. Release Number: LLNL-CONF-2010645.

References

[1] Ming Y Lu, Bowen Chen, Drew FK Williamson, Richard J Chen, Ivy Liang, Tong Ding, Guillaume Jaume, Igor Odintsov, Long Phi Le, Georg Gerber, et al. A visual-language foundation

model for computational pathology. *Nature medicine*, 30(3):863–874, 2024.

[2] Yanrong Ji, Zhihan Zhou, Han Liu, and Ramana V Davuluri. Dnabert: pre-trained bidirectional encoder representations from transformers model for dna-language in genome. *Bioinformatics*, 37(15):2112–2120, 2021.

[3] Mohan Timilsina, Samuele Buosi, Muhammad Asif Razzaq, Rafiqul Haque, Conor Judge, and Edward Curry. Harmonizing foundation models in healthcare: A comprehensive survey of their roles, relationships, and impact in artificial intelligence's advancing terrain. *Computers in Biology and Medicine*, 189:109925, 2025.

[4] Kexin Huang, Payal Chandak, Qianwen Wang, Shreyas Havaldar, Akhil Vaid, Jure Leskovec, Girish N Nadkarni, Benjamin S Glicksberg, Nils Gehlenborg, and Marinka Zitnik. A foundation model for clinician-centered drug repurposing. *Nature Medicine*, 30(12):3601–3613, 2024.

[5] Julián N Acosta, Guido J Falcone, Pranav Rajpurkar, and Eric J Topol. Multimodal biomedical ai. *Nature medicine*, 28(9):1773–1784, 2022.

[6] Chai Discovery. Chai-1: Decoding the molecular interactions of life. *bioRxiv*, 2024.

[7] Thomas Hayes, Roshan Rao, Halil Akin, Nicholas J. Sofroniew, Deniz Oktay, Zeming Lin, et al. Simulating 500 million years of evolution with a language model. *Science*, 387(6736):850–858, 2025.

[8] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 2017.

[9] Eric Nguyen, Michael Poli, Marjan Faizi, Armin Thomas, Michael Wornow, Callum Birch-Sykes, Stefano Massaroli, Aman Patel, Clayton Rabideau, Yoshua Bengio, et al. Hyenadna: Long-range genomic sequence modeling at single nucleotide resolution. *Advances in Neural Information Processing Systems*, 36:43177–43201, 2023.

[10] Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023.

[11] Eugene V Koonin. Why the central dogma: on the nature of the great biological exclusion principle. *Biology direct*, 10(1):52, 2015.

[12] Francis Crick. Central dogma of molecular biology. *Nature*, 227(5258):561–563, 1970.

[13] Tamara Goldfarb, Vamsi K. Kodali, Shashikant Pujar, Vyacheslav Brover, Barbara Robbertse, et al. Ncbi refseq: reference sequence standards through 25 years of curation and annotation. *Nucleic Acids Research*, 53(D1):D243–D257, 2025.

[14] The UniProt Consortium. Uniprot: the universal protein knowledgebase in 2025. *Nucleic Acids Research*, 53(D1):D609–D617, 2024.

[15] Aman Patel, Arpita Singhal, Austin Wang, Anusri Pampari, Maya Kasowski, and Anshul Kundaje. Dart-eval: A comprehensive dna language model evaluation benchmark on regulatory dna. *Advances in Neural Information Processing Systems*, 37:62024–62061, 2024.

[16] Jaina Mistry, Sara Chuguransky, Lowri Williams, Matloob Qureshi, Gustavo A. Salazar, et al. Pfam: The protein families database in 2021. *Nucleic Acids Research*, 49(D1):D412–D419, 2020.

[17] Mark Neumann, Daniel King, Iz Beltagy, and Waleed Ammar. ScispaCy: Fast and Robust Models for Biomedical Natural Language Processing. In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 319–327, Florence, Italy, August 2019. Association for Computational Linguistics.

[18] Xiaoshi Zhong, Rama Kaalia, and Jagath C. Rajapakse. Go2vec: transforming go terms and proteins to vector representations via graph embeddings. *BMC Genomics*, 20(9):918, 2020.

[19] Damian Szklarczyk, Rebecca Kirsch, Mikaela Koutrouli, Katerina Nastou, Farrokh Mehryary, et al. The string database in 2023: protein–protein association networks and functional enrichment analyses for any sequenced genome of interest. *Nucleic Acids Research*, 51(D1):D638–D646, 2022.

[20] Marija Milacic, Deidre Beavers, Patrick Conley, Chuqiao Gong, Marc Gillespie, Johannes Griss, Robin Haw, Bijay Jassal, Lisa Matthews, Bruce May, Robert Petryszak, Eliot Ragueneau, Karen Rothfels, Cristoffer Sevilla, Veronica Shamovsky, Ralf Stephan, Krishna Tiwari, Thawfeek Varusai, Joel Weiser, Adam Wright, Guanming Wu, Lincoln Stein, Henning Hermjakob, and Peter D'Eustachio. The reactome pathway knowledgebase 2024. *Nucleic Acids Research*, 52(D1):D672–D678, 11 2023.

[21] Barbara Zdrazil, Eloy Felix, Fiona Hunter, Emma J Manners, James Blackshaw, Sybilla Corbett, Marleen de Veij, Harris Ioannidis, David Mendez Lopez, Juan F Mosquera, Maria Paula Magarinos, Nicolas Bosc, Ricardo Arcila, Tevfik Kizilören, Anna Gaulton, A Patrícia Bento, Melissa F Adasme, Peter Monecke, Gregory A Landrum, and Andrew R Leach. The ChEMBL Database in 2023: a drug discovery platform spanning multiple bioactivity data types and time periods. *Nucleic Acids Research*, 52(D1):D1180–D1192, January 2024.

[22] A. Lina Heinzke, Barbara Zdrazil, Paul D. Leeson, Robert J. Young, Axel Pahl, Herbert Waldmann, and Andrew R. Leach. A compound-target pairs dataset: differences between drugs, clinical candidates and other bioactive compounds. *Scientific Data*, 11(1):1160, October 2024.

[23] Joanna S Amberger, Carol A Bocchini, Alan F Scott, and Ada Hamosh. Omim.org: leveraging knowledge across phenotype–gene relationships. *Nucleic Acids Research*, 47(D1):D1038–D1043, 11 2018.

[24] Aviad Tsherniak, Francisca Vazquez, Phil G. Montgomery, Barbara A. Weir, Gregory Kryukov, et al. Defining a cancer dependency map. *Cell*, 170(3):564–576.e16, 2017.

[25] DepMap, Broad. DepMap Public 25Q2. Dataset, 2025.

[26] Eric Nguyen, Michael Poli, Matthew G. Durrant, Brian Kang, Dhruva Katrekar, David B. Li, Liam J. Bartie, Armin W. Thomas, Samuel H. King, Garyk Brixi, et al. Sequence modeling and design from molecular to genome scale with evo. *Science*, 386(6723):eado9336, 2024.

[27] Garyk Brixi, Matthew G. Durrant, Jerome Ku, Michael Poli, Greg Brockman, Daniel Chang, Gabriel A. Gonzalez, Samuel H. King, David B. Li, Aditi T. Merchant, et al. Genome modeling and design across all domains of life with evo 2. *bioRxiv*, page 2025.02.18.638918, 2025.

[28] Theo Sanderson, Maxwell L Bileschi, David Belanger, and Lucy J Colwell. Proteinfer, deep neural networks for protein functional inference. *Elife*, 12:e80942, 2023.

[29] ESM Team. Esm cambrian: Revealing the mysteries of proteins with unsupervised learning, 2024.

[30] Yong He, Pan Fang, Yongtao Shan, Yuanfei Pan, Yanhong Wei, Yichang Chen, et al. Generalized biological foundation model with unified nucleic acid and protein language. *Nature Machine Intelligence*, 7(6):942–953, 2025.

[31] Samir Char, Nathaniel Corley, Sarah Alamdari, Kevin K Yang, and Ava P Amini. Protnote: a multimodal method for protein–function annotation. *Bioinformatics*, 41(5):btaf170, 04 2025.

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

[33] Hugo Dalla-Torre, Liam Gonzalez, Javier Mendoza-Revilla, Nicolas Lopez Carranza, Adam Henryk Grzywaczewski, Francesco Oteri, et al. Nucleotide transformer: building and evaluating robust foundation models for human genomics. *Nature Methods*, 22(2):287–297, 2025.

[34] Michael Poli, Armin W Thomas, Eric Nguyen, Pragaash Ponnusamy, Björn Deiseroth, Kristian Kersting, Taiji Suzuki, Brian Hie, Stefano Ermon, Christopher Re, et al. Mechanistic design and scaling of hybrid architectures. In *Forty-first International Conference on Machine Learning*, 2024.

[35] Jerome Ku, Eric Nguyen, David W. Romero, Garyk Brixi, Brandon Yang, et al. Systems and algorithms for convolutional multi-hybrid language models at scale, 2025.

[36] Xin Hou, Yong He, Pan Fang, Shi-Qiang Mei, Zan Xu, Wei-Chen Wu, et al. Using artificial intelligence to document the hidden rna virosphere. *Cell*, 187(24):6929–6942.e16, 2024.

[37] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021.

[38] Frederikke Isa Marin, Felix Teufel, Marc Horlacher, Dennis Madsen, Dennis Pultz, Ole Winther, and Wouter Boomsma. BEND: Benchmarking DNA language models on biologically meaningful tasks. In *The Twelfth International Conference on Learning Representations*, 2024.

[39] Pascal Notin, Mafalda Dias, Jonathan Frazer, Javier Marchena-Hurtado, Aidan N. Gomez, Debora Marks, and Yarin Gal. Tranception: Protein fitness prediction with autoregressive transformers and inference-time retrieval. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162, pages 16990–17017, 2022.

[40] Vinayak Gupta, Brian Bartoldson, Joseph G. Wakim, Jonathan E. Allen, Jose Marti Martinez, Tianlong Chen, and Bhavya Kailkhura. Cost-effective biological data analysis via a benchmark and ensemble of large language models. Technical report, Lawrence Livermore National Laboratory, Livermore, CA, USA, 2025.

[41] Yue Cao, Thomas Andrew Geddes, Jean Yee Hwa Yang, and Pengyi Yang. Ensemble deep learning in bioinformatics. *Nature Machine Intelligence*, 2(9):500–508, 2020.

[42] Yang Qu, Zitong Niu, Qiaojiao Ding, Taowa Zhao, Tong Kong, Bing Bai, et al. Ensemble learning with supervised methods based on large-scale protein language models for protein mutation effects prediction. *International Journal of Molecular Sciences*, 24(22):16496, 2023.

[43] Roshan Rao, Nicholas Bhattacharya, Neil Thomas, Yan Duan, Xi Chen, John Canny, et al. Evaluating protein transfer learning with tape. *Advances in Neural Information Processing Systems*, 32:9689–9701, 2019.

[44] Amina Mollaysa, Artem Moskale, Pushpak Pati, Tommaso Mansi, Mangal Prakash, and Rui Liao. Biolangfusion: Multimodal fusion of dna, mrna, and protein language models, 2025.

[45] Jiayang Chen, Zhihang Hu, Siqi Sun, Qingxiong Tan, Yixuan Wang, et al. Interpretable rna foundation model from unannotated data for highly accurate rna structure and function predictions, 2022.

[46] Mufan Qiu, Sukwon Yun, Ruichen Zhang, Joseph G. Wakim, Jonathan E. Allen, Jose Manuel Marti, Vinayak Gupta, Brian Bartoldson, Bhavya Kailkhura, and Tianlong Chen. Enhancing biological insights with knowledge-driven multi-modal rna models. Technical report, Lawrence Livermore National Laboratory, Livermore, CA, USA, 2025.

[47] Sizhen Li, Saeed Moayedpour, Ruijiang Li, Michael Bailey, Saleh Riahi, Milad Miladi, et al. Codonbert: Large language models for mrna design and optimization. *bioRxiv*, page 2023.09.09.556981, 2023.

[48] Mehdi Yazdani-Jahromi, Mangal Prakash, Tommaso Mansi, Artem Moskalev, and Rui Liao. Helm: Hierarchical encoding for mrna language modeling, 2025.

[49] Genome Reference Consortium. GRCh38.p13 Human Reference Genome Assembly. NCBI Assembly, 2019. Patch release 13 of GRCh38, available at https://www.ncbi.nlm.nih.gov/assembly/GCF_000001405.39/.

[50] The Gene Ontology Consortium, Suzi A. Aleksander, James Balhoff, Seth Carbon, J. Michael Cherry, et al. The gene ontology knowledgebase in 2023. *Genetics*, 224(1), 2023.

[51] Michael Ashburner, Catherine A. Ball, Judith A. Blake, David Botstein, Heather Butler, et al. Gene ontology: tool for the unification of biology. *Nature Genetics*, 25(1):25–29, 2000.

[52] Michael Heinzinger, Konstantin Weissenow, Joaquin Gomez Sanchez, Adrian Henkel, Milot Mirdita, Martin Steinegger, and Burkhard Rost. Bilingual language model for protein sequence and structure. *NAR Genomics and Bioinformatics*, 6(4):lqae150, 11 2024.

[53] Michael Poli, Stefano Massaroli, Eric Nguyen, Daniel Y Fu, Tri Dao, Stephen Baccus, Yoshua Bengio, Stefano Ermon, and Christopher Ré. Hyena hierarchy: Towards larger convolutional language models. In *International Conference on Machine Learning*, pages 28043–28078. PMLR, 2023.

Technical Appendices for:

"Benchmarking Biomolecular Foundation Models for Cross-Modal Genomics-Proteomics"

A Background and Related Work

DNA sequences form a "biological language" that encodes cellular structure and function. Although composed of just four types of nucleotides, this language is complex and encompasses diverse phenotypes. Triads of nucleotides form codons, which map to amino acids. Amino acids with distinct physicochemical properties interact with their environment and fold into proteins. Proteins function as "molecular machines" that provide structure or catalyze reactions. Despite its critical importance to all life, the biological language remains poorly understood.

Since the Human Genome Project, billions of DNA and protein sequences have been measured and made publicly available. Hundreds of millions of these sequences are annotated with functional information. This vast amount of data, alongside recent advances in artificial intelligence, creates a new opportunity to decipher the genetic code. gFMs and pFMs leverage architectures and training strategies from natural language processing (NLP) to learn complex patterns in DNA and protein sequences. These models have proven their versatility across applications in biology and medicine.

DNABERT, Nucleotide Transformer, and Evo2 (with its predecessors Evo and HyenaDNA) represent groundbreaking gFMs [2, 33, 27, 26, 9]. These models demonstrate remarkable abilities to predict both functional and regulatory outcomes, including chromatin profiles, effects of clinical variants, and gene dependencies. While DNABERT and Nucleotide Transformer are based on transformer architectures, Evo2 uses a specialized StripedHyena architecture, which leverages long convolutions to support a broader context window [34, 35]. pFMs like ESMC and LucaProt are trained on functional regions of the genome and excel at tasks like protein structure prediction and function annotation [29, 36]. Multimodal foundation models can achieve greater predictive performance by drawing information from both DNA and protein sequences. For example, LucaOne is concurrently trained on multimodal inputs and excels in tasks like taxonomic classification, subcellular protein localization, and non-coding RNA family detection [30].

Given the variety of gFMs and pFMs available, benchmarks are critical for comparing model performance. NLP offers several examples of effective benchmarks for ranking models. For example, Hendrycks *et al.* developed the Massive Multitask Language Understanding dataset, providing 57 tasks to evaluate NLP model performance across subject areas [37]. For bioFMs, Marin *et al.* and Patel *et al.* developed benchmarks for nucleotide-scale prediction tasks relating to gene regulation, epigenetic modifications, and variant effects [38, 15]. Notin *et al.* published the ProteinGym dataset, a benchmark of protein fitness scores associated with around 1.8 million mutations obtained from deep mutational scanning assays [39]. Here, we introduce the GENO-PROT benchmark, focusing on protein-relevant regions of the genome and enabling simultaneous evaluations of gFMs and pFMs [40]. We use our benchmark to compare single bioFMs with ensembles.

The use of ensembles in computational biology represents an emerging strategy for improving predictive performance. Cao *et al.* review supervised and unsupervised ensembling approaches, focusing on applications in bioinformatics [41]. Qu *et al.* leverage ensembles of pFMs to predict the effects of mutations reported in the ProteinGym dataset [42, 39, 43]. Mollaysa *et al.* evaluate several fusion strategies for combining DNA, RNA, and protein embeddings generated by Nucleotide Transformer, RNA-FM, and ESM-2, respectively [44, 33, 45, 10, 46]. The group evaluates their models on benchmarks mapping messenger RNA (mRNA) sequences to phenotypic outcomes across species and applications [47, 48]. In this work, we advance the development and evaluation of multimodal ensembles of bioFMs by including more advanced models and designing benchmarks that relate DNA and protein sequences to attributes in humans.

B Detailed Specifications of Geno-Prot

The Geno-Prot benchmark includes matched gene and protein sequences for human proteins listed in UniProt. We map the sequences to binary outcomes, creating nine prediction tasks used to compare gFMs, pFMs, and multimodal models. In the subsections below, we describe the sources of our data and the methods with which we organize them into prediction tasks.

B.1 DNA and Protein Sequences

Our benchmark characterizes human gene and protein sequences derived from the GRCh38.p13 version of the human genome reference assembly [49]. Using annotations provided by RefSeq, we extract amino acid sequences associated with each gene [13]. Fig. 4 plots the distribution of sequence lengths and the proportion of DNA sequences that encode amino acids. We map RefSeq protein accessions to UniProt IDs, then load experimentally verified GO annotations for each gene [14, 50, 51]. We isolate genes with available ProtT5 embeddings [52]. ProtT5 embeddings are used to partition our dataset into roughly equivalent folds (see Sec. B.2). Ultimately, we store the UniProt ID, RefSeq accessions, nucleotide sequences, amino acid sequences, and GO annotations associated with 12,448 distinct proteins, which we reference for the prediction tasks.

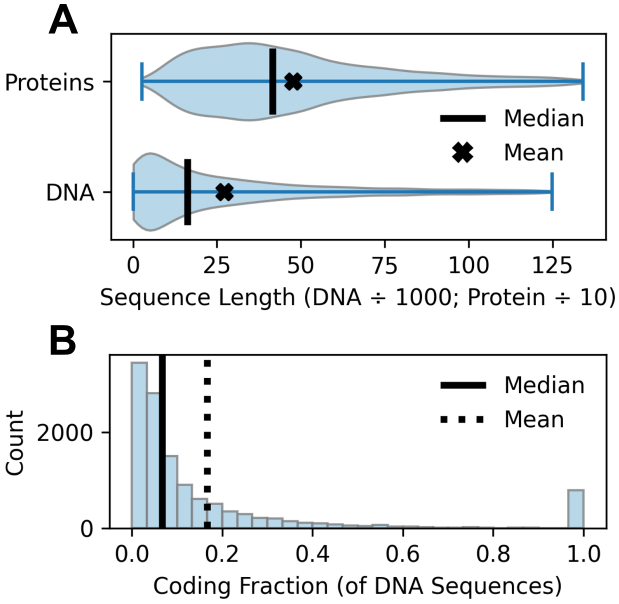


Figure 4: **Features of sequences in the dataset.** **(A)** Distributions of DNA and protein sequence lengths, excluding outliers beyond 1.5 × IQR from the first or third quartile (Tukey rule). To enable visualization on a common scale, DNA sequence lengths are divided by 1,000 and protein sequence lengths are divided by 10. **(B)** Distribution of coding nucleotide fractions in DNA sequences.

B.2 Data Partitioning

To evaluate prediction tasks, we train prediction heads that take embeddings from gFMs or pFMs and predict binary labels. We isolate a subset of the human proteome for the purpose of training the prediction heads. This subset includes 8,448 proteins, leaving a non-overlapping subset of 4,000 proteins reserved for evaluation. To ensure that the training and evaluation subsets are equivalent, we embed the proteins using ProtT5 [52], then partition the proteins according to Algorithm 1.

Algorithm 1 Partitioning the human proteome into non-overlapping training and evaluation subsets.

Inputs:

- `ProteinList`: List of UniProt IDs for human proteins
- `EmbeddingMap`: Map from UniProt ID to ProtT5 embedding
- `N`: Desired number of proteins in the evaluation subset

Outputs:

- `TrainFold`: Set of proteins for training
- `EvalFold`: Set of proteins for evaluation

Pseudocode:

```
Initialize EvalFold $\leftarrow \emptyset$
Initialize TrainFold $\leftarrow$ ProteinList
Randomly select a protein $P_0$ from TrainFold
Add $P_0$ to EvalFold
Remove $P_0$ from TrainFold
while len(EvalFold) $<$ N do
    Initialize empty list MinDists
    for each protein $P_i$ in TrainFold do
        Compute squared Euclidean distances from $P_i$ to each protein in EvalFold using ProtT5 embeddings
        from EmbeddingMap
        Let $d_i$ be the minimum of these squared distances
        Append $d_i$ to MinDists
    end for
    Normalize MinDists to form a probability distribution $\pi$ over TrainFold, where $\pi_i \propto d_i$
    Sample a protein $P_{\text{sample}}$ from TrainFold according to probabilities $\pi$
    Add $P_{\text{sample}}$ to EvalFold
    Remove $P_{\text{sample}}$ from TrainFold
end while
return TrainFold, EvalFold
```

After partitioning the proteins into training and evaluation subsets, we verify that the UMAP projections of the ProtT5 embeddings for proteins in each subset approximately represent the full dataset (see Fig. 5). We also verify that feature vectors encoding the probability distributions of subcellular locations, anatomical locations, and biochemical pathways are well correlated.

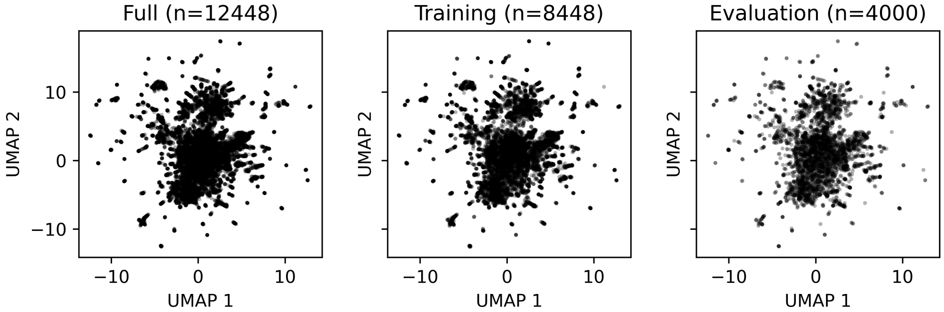


Figure 5: **UMAP projections of ProtT5 embeddings for proteins in the training and evaluation subsets of the dataset.** The consistency between the UMAP projections suggests that the training and evaluation subsets are representative of the full dataset.

To mitigate data leakage, we prepare separate training and evaluation datasets for each prediction task from the corresponding protein subsets. To ensure class balance within each benchmark dataset, we randomly select some number of positive and negative examples according to Tab. 1, based on the prevalence of the minority class.

Table 1: **Dataset Sizes.** An equal number of random examples were drawn from the positive and negative classes in each benchmark dataset. The values reported in this table represent the number of examples drawn from each class. Each evaluation dataset is sampled 10 times, enabling quantification of variability in scored model performance. Evaluation dataset sizes are reported on a per-sample basis.

Prediction Task	**Training**	**Evaluation**
Shared Pfam	25,000	150
Shared Subcell. Loc.	100,000	150
Shared Anatom. Loc.	100,000	150
Prot. Funct. Sim.	100,000	150
Prot.-Prot. Interact.	50,000	150
Shared Biochem. Path.	100,000	150
Shared Drug Binding	35,000	150
Mel. Gene Dep.	800	100
Shared Disease Implic.	400	20

The full data partitioning process is depicted in Fig. 6. To estimate the uncertainty in each model's performance, we randomly sample each evaluation dataset 10 times and evaluate the accuracies across these samples.

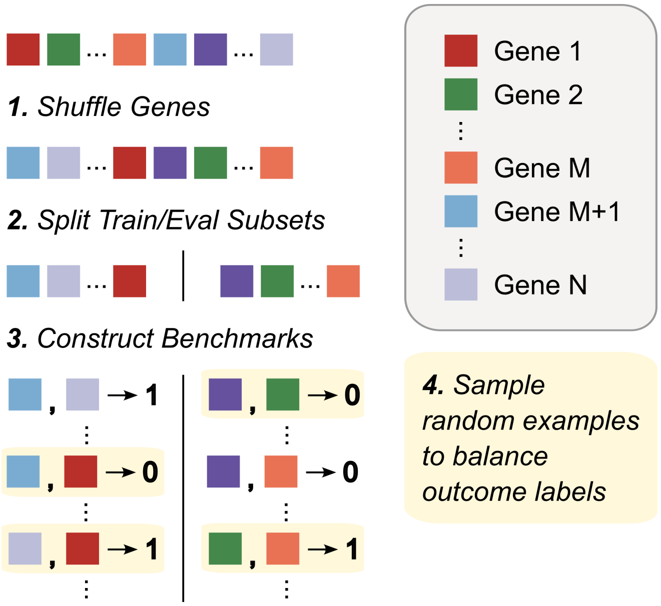


Figure 6: **Partitioning training/evaluation datasets.** We randomly partition proteins into roughly equivalent training and evaluation subsets. We separately sample pairs of proteins to form each benchmark from the training and evaluation folds to mitigate information leakage.

C Experiments and Discussions

We use GENO-PROT to compare individual bioFMs, ensembles of bioFMs, and general-purpose LLMs. Most evaluations were performed on a machine using AMD MI-250X GPUs with 128 GB of memory. However, since Evo and Evo2 required CUDA-specific drivers, evaluations involving those models were run on a machine using H100 GPUs with 80 GB of memory.

C.1 Model Details

Descriptions of the models included in our evaluation are listed below and summarized in Tab. 2.

C.1.1 Gene Foundation Models

1. **HyenaDNA** [9]: HyenaDNA processes long DNA sequences, up to one million nucleotides in length, using the efficient Hyena operator [53], which replaces attention with fast long convolutions and gating. This design enables the model to learn and understand genetic information faster and with less computing power than previous bioFMs. We evaluate three versions of HyenaDNA, each with eight layers but different context sizes: small (160 thousand tokens), medium (450 thousand tokens), and large (one million tokens). We label these models "hyena-160k," "hyena-450k," and "hyena-1m," respectively.
2. **Evo** [26]: Evo is a gFM developed by the Arc Institute that leverages a novel StripedHyena architecture to efficiently process long sequences (up to 131 thousand bases in length). The model is trained on trillions of nucleotides from diverse organisms, enabling accurate predictions of genetic functions and mutations across domains of life.
3. **Evo2** [27]: Evo2 is a recently published state-of-the-art gFM building on Evo. The model is trained on over 9.3 trillion nucleotide tokens from more than 128 thousand genomes. Evo2 demonstrates impressive capabilities in zero-shot gene function prediction, gene essentiality assessment, and CRISPR system design. The model's architecture includes layers categorized into Short Explicit (SE), Medium Regularized (MR), and Long Implicit (LI), allowing it to handle extensive genomic data effectively. Evo2 represents a significant advancement in genomic AI, providing researchers with a powerful tool for exploring and manipulating genetic information across diverse organisms. We use the seven-billion parameter variant of Evo2, denoted "evo2-7b."

C.1.2 Protein Foundation Models

1. **ProteInfer** [28]: ProteInfer is developed by Google Research to predict functions directly from protein sequences. The model leverages deep convolutional neural networks trained on protein sequences and annotations. We use three variants of ProteInfer trained on different annotation types: EC numbers, GO terms, and Pfam labels. We denote these variants as "proteinfer_ec," "proteinfer_go," and "proteinfer_pfam," respectively.
2. **ESMC** [29]: ESMC is a pFM trained on massive protein sequence data in an unsupervised manner. The model captures important biological patterns to predict protein structure and function. ESMC is an embedding-focused counterpart to ESM3, a state-of-the-art pFM that has been used to design fluorescent proteins with chain-of-thought reasoning [7]. We use two publicly available variants of ESMC: the 300-million-parameter and 600-million-parameter versions. We refer to these models as "esmc-300m" and "esmc-600m," respectively.

C.1.3 Multimodal Model

1. **LucaOne** [30]: LucaOne is a unified bioFM trained on DNA, RNA, and protein sequences from over 169 thousand species. The model uses an enhanced transformer with token-type embeddings to distinguish biomolecular modalities, which has enabled it to encode the central dogma of molecular biology. LucaOne supports few-shot learning and achieves strong performance across diverse genomic and proteomic tasks.

C.1.4 Ensembles of BioFMs

We combine the predictions of leading gFMs and pFMs to incorporate multimodal information. All ensembles leverage Evo2, esmc-600m, proteinfer_ec, proteinfer_go, proteinfer_pfam, and LucaOne to make predictions. Below, we describe five strategies for combining predictions.

1. **Soft Voting (S-ENSEM):** To perform soft voting, we convert logits to probabilities using a softmax function, then average probabilities across models. The final class prediction is selected based on the highest average probability. This method considers model confidence and typically offers smoother predictions.

2. **Hard Voting (H-ENSEM):** In hard voting, each model contributes a single class label, and the final prediction corresponds to the majority class. This approach does not account for prediction confidence and assigns equal weight to all models. Hard voting is most effective when the individual models tend to produce consistent predictions.
3. **Weighted Voting (W-ENSEM):** In weighted voting, each model is assigned a weight based on its performance on a validation fold of each benchmark dataset. Class probabilities are averaged according to these weights, and the class with the greatest probability is selected for the prediction. This approach extends soft voting by incorporating model-specific weightings for each task. In Tab. 3, we report the weights assigned to individual models for each task.
4. **Median Voting (Med-ENSEM):** In median voting, the median predicted probability across all models is determined for each class. The class with the greatest median probability is selected. This approach reduces the impact of outliers or overconfident models, enhancing robustness.
5. **Max Voting (Max-ENSEM):** In max voting, the maximum predicted probability across all models is determined for each class, and the class with the greatest maximum probability is selected. This approach prioritizes the most confident model, favoring high-certainty predictions.

Table 2: **Model specifications.** LucaOne and ESMC can process long sequences by batching and aggregating embeddings. ProteInfer variants use convolutional architectures without a context window, relying on fixed-size filters.

Model Name	Size	Context	Type
hyena-160k	-	160k	gFM
hyena-450k	-	450k	gFM
hyena-1m	6.6M	1M	gFM
evo	7B	131k	gFM
evo2-7b	7B	1M	gFM
esmc-300m	300M	2048	pFM
esmc-600m	600M	2048	pFM
proteinfer_ec	-	-	pFM
proteinfer_go	-	-	pFM
proteinfer_pfam	-	-	pFM
lucaone	1.8B	1280	gFM, pFM

Table 3: **W-ENSEM weights.** For the weighted voting ensemble, the class probabilities of individual models are aggregated with a weighted average. The weight of each model is selected according to its performance on a validation fold associated with each prediction task. Weights for individual models and prediction tasks are listed below.

Task	esmc-600m	evo2-7b	proteinfer_ec	proteinfer_go	proteinfer_pfam	lucaone
Shared Pfam	0.166	0.158	0.167	0.166	0.174	0.165
Shared Subcell. Loc.	0.172	0.154	0.166	0.181	0.161	0.164
Shared Anatom. Loc.	0.174	0.162	0.169	0.173	0.159	0.160
Prot. Funct. Sim.	0.163	0.166	0.160	0.191	0.160	0.158
Prot.-Prot. Interact.	0.175	0.152	0.164	0.179	0.164	0.163
Shared Biochem. Path.	0.177	0.161	0.157	0.183	0.158	0.161
Shared Drug Binding	0.175	0.158	0.162	0.165	0.167	0.170
Mel. Gene Dep.	0.176	0.174	0.159	0.165	0.159	0.164
Shared Disease Implic.	0.164	0.166	0.155	0.172	0.164	0.176

Listing C.1 System prompt for evaluating LLMs on Geno-Prot prediction tasks.

```
SYSTEM_PROMPT = """
You are a bioinformatics model trained to assess relationships or properties of
    proteins. You will be given one or two protein identifiers or amino acid
    sequences in fill-in-the-blank format, where <UNIPROT:####...#> indicates a
    UniProt ID and <SEQ:####...#> indicates an amino acid sequence. You will also
    receive a question that communicates the true/false prediction task.

Your response must follow these strict rules:
- Only return the answer in pure JSON format, with a single key "result" and a
    boolean value (true or false).
- Do not explain your answer or include any other text.
- Do not describe the sequences, IDs, or the question.
- Do not provide any formatting, logging, or annotation.
- Your output must be machine-readable and strictly JSON-only.

Input format (examples):

Are proteins <UNIPROT:P19086> and <UNIPROT:P60880> involved in at least one common
    biochemical pathway?

Do proteins <UNIPROT:P12345> and <UNIPROT:Q67890> share at least one common Pfam
    label?

Are melanoma tumor cells strongly dependent on protein <UNIPROT:Q9Y6K9>?

Do proteins <SEQ:MVLSPADKTNVKAA...W*> and <SEQ:MKVIFLAGKQLEDGR...T*> share similar
    functions?

Expected output format:
{"result": true}
or
{"result": false}
"""
```

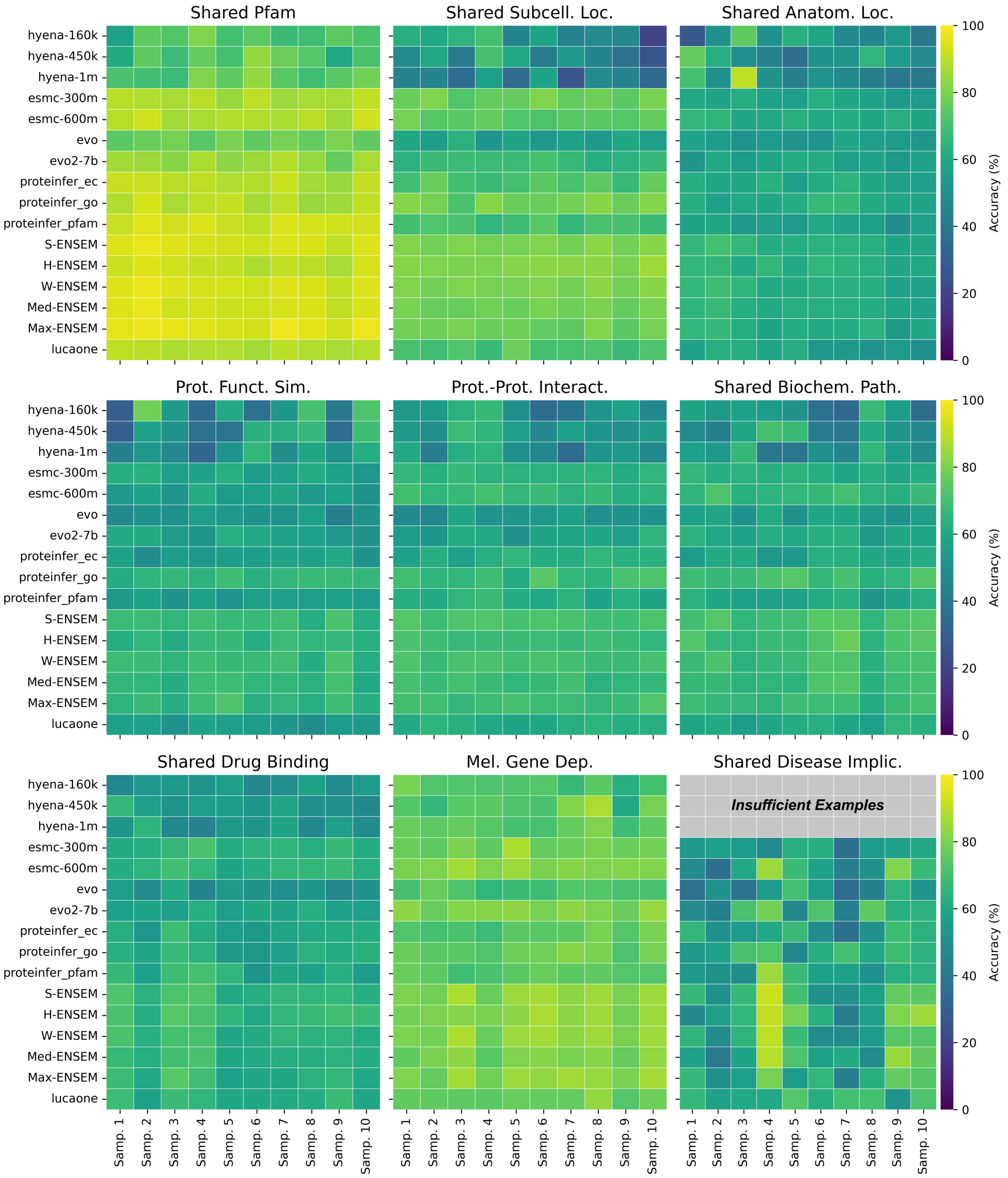


Figure 7: **Model performance across evaluation samples.** We plot the accuracy of each model across ten samples from our evaluation dataset.