Scaling Up ESM2 Architectures for Long Protein Sequences Analysis: Long and Quantized Approaches

Gabriel Bianchin de Oliveira[1], **Helio Pedrini**[1], **and Zanoni Dias**[1]

[1]Institute of Computing, University of Campinas, Campinas, SP, Brazil

{gabriel.oliveira, helio, zanoni}@ic.unicamp.br

Abstract. *Various approaches utilizing Transformer architectures have achieved state-of-the-art results in Natural Language Processing (NLP). Based on this success, numerous architectures have been proposed for other types of data, such as in biology, particularly for protein sequences. Notably among these are the ESM2 architectures, pre-trained on billions of proteins, which form the basis of various state-of-the-art approaches in the field. However, the ESM2 architectures have a limitation regarding input size, restricting it to 1,022 amino acids, which necessitates the use of preprocessing techniques to handle sequences longer than this limit. In this paper, we present the long and quantized versions of the ESM2 architectures, doubling the input size limit to 2,048 amino acids.*

1. Introduction

Transformer-based models have become state-of-the-art in various Natural Language Processing (NLP) tasks, such as context analysis, text generation, and translation. Recently, tools that utilize Transformers, such as ChatGPT[1] and Copilot[2], have become instrumental in assisting users with their tasks.

The success of architectures that employ Transformers stems from attention modules, capable of learning the relationships between words in a sentence autonomously (self-attention) [Vaswani et al. 2017]. In the pre-training phase, these models are exposed to the context of the language they are being trained in, which typically consists of collections ranging from millions to billions of documents, enabling them to learn and adapt to the nuances of the language. After this initial stage, which takes a substantial amount of time and requires significant processing power, users can fine-tune the model for specific tasks.

Following the success in Natural Language Processing, this approach is also being applied in other contexts, such as images [Arnab et al. 2021, Dosovitskiy et al. 2020] and audio [Ao et al. 2021]. In the biological domain, several Transformer-based architectures have also been developed, becoming state-of-the-art in tasks such as protein structure prediction [Abramson et al. 2024, Lin et al. 2023], protein representation extraction [Elnaggar et al. 2021], biological article analysis [Lee et al. 2020], and DNA sequence analysis [Zhou et al. 2023].

Considering the approaches for proteins, ESM family architectures, developed by the MetaAI group, with the most recent version being ESM2 [Lin et al. 2023], are

[1]https://chatgpt.com

[2]https://copilot.microsoft.com

among the state-of-the-art approaches in various tasks, such as protein function prediction [Zhapa-Camacho et al. 2024], protein family annotations [Vitale et al. 2024], and protein sequence conservation [Yeung et al. 2023]. The original ESM2 architecture has restrictions regarding the maximum sequence size of 1,022 amino acids, which, together with the `CLS` token, used to indicate the beginning of the sequence and utilized in classification tasks, and the `EOS` token, used to indicate the end of the sequence, total a 1,024-token input limit. However, there are protein sequences larger than this maximum size, forcing the approaches to use techniques such as truncation up to this limit, excluding larger proteins, or treatments with sliding window techniques to deal with longer sequences.

In this paper, we introduce the long versions of ESM2 architectures, which can process proteins with up to 2,048 amino acids without the need for additional preprocessing. Besides the standard long versions, we also present the quantized long versions, referred to simply as quantized, which apply quantization to reduce memory space required for model loading and to accelerate inference time.

Quantization is a technique commonly used in neural networks, including Transformers, to decrease the model's precision from 32-bit floating point to lower bit-width representations, such as 8-bit and 4-bit integers. This process significantly reduces memory usage and computational requirements, often with minimal impact on model accuracy. Consequently, model loading times and inference speeds are improved, making it a desirable option for deploying large models in resource-constrained environments.

During our evaluation, we assessed the ESM2 long and ESM2 quantized architectures for the task of protein function prediction. In most cases, these architectures demonstrated superior performance compared to the standard ESM2 architecture.

The remainder of the paper is organized as follows. In Section 2, we describe the proposed architectural adaptation of ESM2 models to deal with sequences up to 2,048 amino acids. In Section 3, we evaluate and discuss the results for the protein function prediction task using the embeddings extracted from the long and quantized architectures and compare them with the standard ones. In Section 4, we present the main aspects of our work and indicate possible points for future research.

2. Methodology

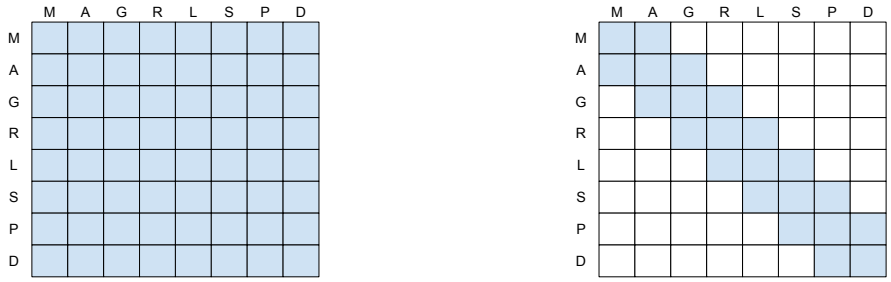
ESM2 architectures have configurations highlighted in Table 1. Each of these architectures employs the concept of self-attention memory modules with global mechanism, that is, each token (amino acid, `CLS`, or `EOS`) examines all other tokens in the sequence, as depicted in Figure 1. Taking into account memory and computational processing, the ESM2 architectures perform attention calculation in $\mathcal{O}(n^2)$, where n is the sequence length.

Inspired by LongFormer [Beltagy et al. 2020], we modified the attention mechanisms of the ESM2 architectures to local form, in which each token considers only the other tokens within a window of size k, as depicted in Figure 1. Consequently, the computational and memory complexity takes the form $\mathcal{O}(nk)$, where n is the sequence length and k is the window size.

To implement this adaptation, we copied the context representation of the ESM2 architectures to 2,050 positions, with 2,048 allocated for the amino acids and 2 positions

Table 1. Configuration of ESM2 architectures. Each architecture in the ESM2 family has n stacked layers, ranging from 6 in T6 up to 48 in T48.

	T6	T12	T30	T33	T36	T48
Number of Layers	6	12	30	33	36	48
Attention Heads	20	20	20	20	40	40
Embedding Dimension	320	480	640	1,280	2,560	5,120



(a) Global. (b) Local.

Figure 1. Self-attention mechanisms. In the global self-attention mechanism, each amino acid examines all the amino acids in the sequence. In the local self-attention mechanism, each amino acid examines the amino acids within a specific window.

for special tokens (`CLS` and `EOS`). We adopted this approach based on the results from Beltagy *et al.* [Beltagy et al. 2020], which demonstrated that context copying is more effective than random initialization. In addition to altering the context representation, we also modified the attention modules from global to local mechanisms.

Concerning the window size of attention, we maintained the window size at 1,024 and increased the sequence limit to 2,048 amino acids. Consequently, even though this increases the memory requirement compared to the original models, which only had an input size of 1,024 tokens, the memory needed for adapting the model to accommodate an input size of 2,050 tokens (up to 2,048 amino acids, `CLS`, and `EOS`) with global attention analysis was halved using our approach.

In addition to the long version, we transformed ESM2 long architectures into quantized versions. For this, we carried out the same process described for the long version, but during the architecture adaptation and pre-training stage, we loaded the models in the `int4` format [Dettmers and Zettlemoyer 2023], using LoRA [Yu et al. 2023] and `bfloat16` computation type. Unlike the standard representation of machine learning models, which is `float32`, representing each weight and network activation by 32 float values, the `int4` version performs this representation with only 4 integer values, reducing the memory required to load models by up to 8 times, while at a cost in terms of model performance.

Following the modifications to the architecture for the long and quantized versions, we pre-trained the networks considering all proteins (569,793) available in July 2023 in the UniProt database, Swiss-Prot version [The UniProt Consortium 2023]. We opted for this version given that the proteins in this set have been reviewed by laboratory

Table 2. Memory required to load each ESM2 architecture (in MB). Each architecture in the ESM2 family has n stacked layers, ranging from 6 in T6 up to 48 in T48.

	T6	T12	T30	T33	T36
Standard	31	136	595	2,673	11,643
Long	40	171	746	3,338	-
Quantized	314	328	384	664	1,750

methods compared to UniProtKB-TrEMBL. During this stage, we trained the models for 5 epochs, with a learning rate of 10^{-5} and the AdamW [Loshchilov and Hutter 2017] optimizer.

Table 2 presents the amount of memory required (in MB) to load each model. The memory requirement for small quantized models, such as T6, exceeds that of the standard and long configurations. However, as the model size increases, quantization proves to be memory-efficient, reducing the required memory by approximately four times for the largest ESM2 architecture that has both long and quantized versions (T33).

Due to computational limitations, we were unable to transform the ESM2 T36 architecture into the long version, nor the ESM2 T48 architecture into both the long and quantized versions. All the ESM2 long and quantized architectures are available in our HuggingFace webspace[3].

3. Results and Discussion

In order to assess the embedding representations of both the long and quantized versions of ESM2 architectures, we conducted an evaluation with respect to the task of protein function prediction.

With the recent advancements of the past decades, such as next-generation sequencing, numerous proteins have had their amino acid sequences defined by laboratory methods. However, determining the functions that each of these proteins performs remains quite costly, considering both the time and the financial resources required for this type of laboratory analysis. As a result, various computational methods have been proposed to reduce the gap between proteins that have a defined sequence but lack an annotated function [Cao and Shen 2021, Chua et al. 2024, Kulmanov and Hoehndorf 2019, Oliveira et al. 2023, Oliveira et al. 2024, Zhu et al. 2022].

Protein functions are typically classified using Gene Ontology (GO) [Ashburner et al. 2000]. This approach divides function annotation into three ontologies: Biological Process Ontology (BPO), which evaluates the overall process in which the protein is involved; Cellular Component Ontology (CCO), which indicates the location where the protein is executing its function; and Molecular Function Ontology (MFO), which analyzes the function performed at the molecular level. These three ontologies are organized in a directed acyclic graph, where deeper terms are more specific, while terms closer to the root term are more generic. Thus, if a protein performs a more specific function, it also performs all terms up to the root term, following the true path rule [Valentini 2010]. Moreover, each protein can perform more than one function at

[3]https://huggingface.co/gabrielbianchin

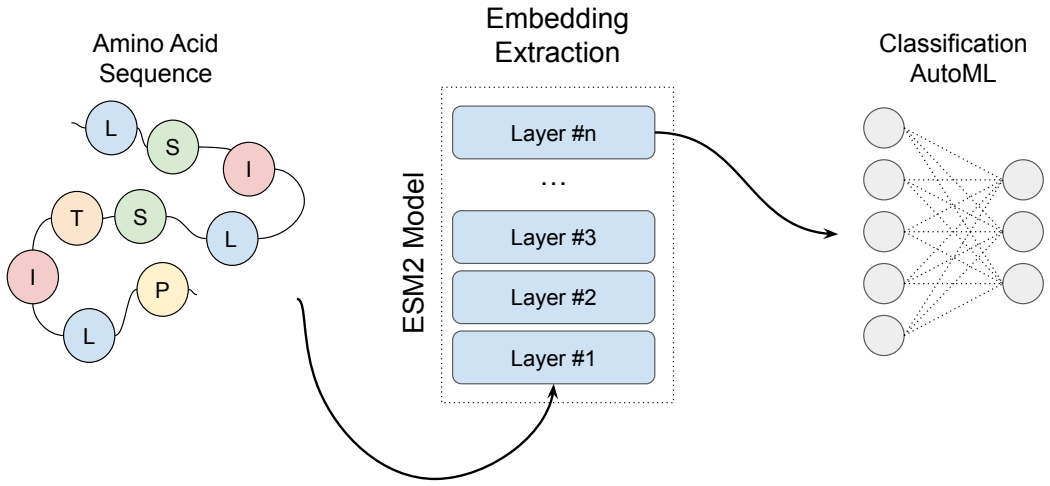


Figure 2. Pipeline for evaluating protein embeddings from ESM2 architectures. The method receives the amino acid sequence as input. Then, the features from the last layer of the backbone are used to train a classifier. During the classification step, the best classification model is identified using AutoML.

the same time, even if these functions have no shared ancestral terms. Due to this nature, the problem of classifying protein functions is considered a multi-label classification by computational methods.

To evaluate the representations (embeddings) extracted from ESM2 architectures, we employed the pipeline described by Oliveira et al. [Oliveira et al. 2024], illustrated in Figure 2. In the initial stage, referred to as embedding extraction, we extracted embeddings from the last layer of each architecture for every protein in the training, validation, and test sets. Due to the length of the sequences and the maximum input size of the standard, long, and quantized architectures, if a protein sequence exceeds 1,022 (standard) or 2,046 (long and quantized) amino acids, we applied the sliding window technique to segment the sequence into non-overlapping slices that fit within the models' maximum input size. For example, if a protein has 3,000 amino acids, for the standard configuration, there will be two slices of 1,022 amino acids (the first from amino acid 1 to amino acid 1,022, and the second from amino acid 1,023 to amino acid 2,044) and one slice with 956 amino acids (from amino acid 2,045 to 3,000). For the long and quantized versions, there will be two slices, one with 2,046 amino acids (from amino acid 1 to 2,046) and another with 954 amino acids (the remaining ones).

After extracting the embeddings, if a protein was separated into slices, we aggregated all representations by averaging the feature vectors position-wise. If a protein sequence length was less than the model input limit, no preprocessing steps were applied. At the end of this process, each protein in the training, validation, and test sets is represented by real values, with the representation vector matching the embedding dimension specified in Table 1.

Table 3. Number of proteins and terms for BPO, CCO, and MFO.

	BPO	CCO	MFO
Training	73,768	74,328	62,909
Validation	9,221	9,292	7,864
Test	9,221	9,292	7,864
Terms	500	498	499

With the extracted embeddings, for each scenario – considering architecture (T6, T12, T30, or T36), input size (standard, long, or quantized), and ontology (BPO, CCO, or MFO) – we obtained a classifier using AutoML from the AutoKeras [Jin et al. 2023] package with 50 trials, selecting the best classifier for each configuration based on the validation set. Finally, the best classifier identified for each scenario was evaluated on the test set.

The dataset utilized for the evaluation of ESM2 embeddings was chosen as outlined in the work of Oliveira et al. [Oliveira et al. 2024], which is derived from the CAFA5 challenge [Friedberg et al. 2023]. The number of proteins in the training, validation, and test sets, as well as the number of terms in each ontology, are detailed in Table 3. With respect to proteins consisting of more than 1,024 amino acids, this dataset comprises approximately 12% for BPO, 11% for CCO, and 10% for MFO. In the case of proteins with more than 2,048 amino acids, it is approximately 2% for each ontology.

As an evaluation metric, we utilized $F_{\max}$, which is the most commonly employed metric in the task of protein function prediction [Radivojac 2013, Zhou et al. 2019]. $F_{\max}$ assesses the maximum F-score considering the thresholds τ ranging from 0.01 to 1.00, applying the harmonic mean between precision and recall at each τ. Equations 1, 2, and 3 present $F_{\max}$, precision at τ (denoted by $\mathrm{pr}(\tau)$), and recall at τ (denoted by $\mathrm{rc}(\tau)$). In these formulas, T_i represents the ground-truth of a protein i, $P_i(\tau)$ is the set of terms predicted for a protein i at a threshold τ, $m(\tau)$ indicates the number of proteins with at least one term predicted with a score equal to or greater than τ, and n is the number of proteins in the evaluation set.

$$F_{\max} = \max_{\tau} \left\{ \frac{2 \times \mathrm{pr}(\tau) \times \mathrm{rc}(\tau)}{\mathrm{pr}(\tau) + \mathrm{rc}(\tau)} \right\} \tag{1}$$

$$\mathrm{pr}(\tau) = \frac{1}{m(\tau)} \sum_{i=1}^{m(\tau)} \frac{|P_i(\tau) \cap T_i|}{|P_i(\tau)|} \tag{2}$$

$$\mathrm{rc}(\tau) = \frac{1}{n} \sum_{i=1}^{n} \frac{|P_i(\tau) \cap T_i|}{|T_i|} \tag{3}$$

The results of the models using ESM2 standard, long, and quantized embeddings for the protein function prediction task on the test set are presented in Table 4. These results indicate that the optimal values, or the highest results for each type of architecture, from ESM2 T6 to ESM2 T33, are attained by long and/or quantized architectures, with the exception in ESM2 T30 for BPO. With respect to ESM2 T36, the quantized version

Table 4. $F_{\max}$ **of ESM2 standard, long and quantized embeddings on the test set.**

Method	BPO	CCO	MFO
ESM2 T6			
Standard	0.505	0.723	0.754
Long	**0.509**	**0.733**	**0.757**
Quantized	0.498	0.727	0.744
ESM2 T12			
Standard	0.505	0.728	0.762
Long	**0.532**	**0.734**	**0.778**
Quantized	0.505	0.729	0.777
ESM2 T30			
Standard	**0.539**	0.739	0.770
Long	0.527	0.742	0.766
Quantized	0.509	**0.743**	**0.778**
ESM2 T33			
Standard	0.540	0.736	0.773
Long	0.512	**0.751**	0.782
Quantized	**0.549**	0.747	**0.783**
ESM2 T36			
Standard	**0.555**	0.755	**0.793**
Quantized	0.531	**0.760**	0.785

achieved the best results for CCO, while the standard architecture surpassed it in the other two ontologies.

Next, we assessed the performance of each approach by focusing exclusively on proteins with more than 1,024 amino acids in the test set. Table 5 presents the results, indicating that the long and/or quantized embeddings of ESM2 T6, T12, T30, and T33 architectures achieved the highest $F_{\max}$ scores for BPO, CCO, and MFO compared to the standard configuration. For T36 embeddings, the quantized version yielded the best results for CCO and MFO. These findings lead us to conclude that ESM2 long and/or quantized embeddings are better suited for handling sequences with more than 1,024 amino acids compared to the corresponding standard models in most cases.

4. Conclusions

In this study, we introduce an adaptation of ESM2 architectures for sequences encompassing up to 2,048 amino acids, effectively doubling the input size that the original ESM2 models can handle. In terms of the results in the protein function prediction task, the classifiers utilizing embeddings derived from long or quantized versions have outperformed the standard ESM2 configuration during our evaluation in most cases.

For future research, we highlight the adaptation of several architectures for long sequences, such as ProtT5 [Elnaggar et al. 2021]. Furthermore, we recognize the significance of analyzing ESM2 long and quantized architectures across different tasks. Additionally, since the long and quantized versions are pre-trained on protein data, we encour-

Table 5. $F_{\max}$ **of ESM2 standard, long and quantized embeddings for proteins with more than 1,024 amino acids on the test set.**

Method	BPO	CCO	MFO
ESM2 T6			
Standard	0.501	0.686	0.731
Long	**0.516**	**0.712**	**0.750**
Quantized	0.505	0.702	0.732
ESM2 T12			
Standard	0.493	0.684	0.743
Long	**0.533**	**0.698**	0.767
Quantized	0.516	0.693	**0.771**
ESM2 T30			
Standard	**0.525**	0.698	0.754
Long	**0.525**	0.711	0.758
Quantized	0.506	**0.717**	**0.771**
ESM2 T33			
Standard	0.517	0.690	0.761
Long	0.511	**0.718**	0.767
Quantized	**0.556**	0.716	**0.771**
ESM2 T36			
Standard	**0.533**	0.717	0.770
Quantized	0.528	**0.733**	**0.771**

age the application of these architectures in fine-tuning processes for specific tasks, such as protein secondary structure and contact map prediction.

Acknowledgements

The authors would like to thank the São Paulo Research Foundation (2017/12646-3), the National Council for Scientific and Technological Development (161015/2021-2, 304380/2018-0, 309330/2018-1), the Coordination for the Improvement of Higher Education Personnel, Santander Bank - Brazil, LNCC/MCTI for providing HPC resources of the SDumont supercomputer, and Centro Nacional de Processamento de Alto Desempenho em São Paulo (CENAPAD-SP) for providing computational resources.

References

Abramson, J., Adler, J., Dunger, J., Evans, R., Green, T., Pritzel, A., Ronneberger, O., Willmore, L., Ballard, A. J., Bambrick, J., Bodenstein, S. W., Evans, D. A., Hung, C.-C., O'Neill, M., Reiman, D., Tunyasuvunakool, K., Wu, Z., Žemgulytė, A., Arvaniti, E., Beattie, C., Bertolli, O., Bridgland, A., Cherepanov, A., Congreve, M., Cowen-Rivers, A. I., Cowie, A., Figurnov, M., Fuchs, F. B., Gladman, H., Jain, R., Khan, Y. A., Low, C. M. R., Perlin, K., Potapenko, A., Savy, P., Singh, S., Stecula, A., Thillaisundaram, A., Tong, C., Yakneen, S., Zhong, E. D., Zielinski, M., Žídek, A., Bapst, V., Kohli, P., Jaderberg, M., Hassabis, D., and Jumper, J. M. (2024). Accurate structure prediction of biomolecular interactions with AlphaFold 3. *Nature*, pages 1–3.

Ao, J., Wang, R., Zhou, L., Wang, C., Ren, S., Wu, Y., Liu, S., Ko, T., Li, Q., Zhang, Y., Wei, Z., Qian, Y., Li, J., and Wei, F. (2021). SpeechT5: Unified-Modal Encoder-Decoder Pre-Training for Spoken Language Processing. *arXiv:2110.07205*, pages 1–16.

Arnab, A., Dehghani, M., Heigold, G., Sun, C., Lučić, M., and Schmid, C. (2021). ViViT: A Video Vision Transformer. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6836–6846.

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., Harris, M. A., Hill, D. P., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J. C., Richardson, J. E., Ringwald, M., Rubin, G. M., and Sherlock, G. (2000). Gene Ontology: Tool for the Unification of Biology. *Nature Genetics*, 25(1):25–29.

Beltagy, I., Peters, M. E., and Cohan, A. (2020). Longformer: The long-document transformer. *arXiv:2004.05150*, pages 1–17.

Cao, Y. and Shen, Y. (2021). TALE: Transformer-based protein function Annotation with joint sequence–Label Embedding. *Bioinformatics*, 37(18):2825–2833.

Chua, Z. M., Rajesh, A., Sinha, S., and Adams, P. D. (2024). PROTGOAT: Improved automated protein function predictions using Protein Language Models. *bioRxiv*, pages 1–15.

Dettmers, T. and Zettlemoyer, L. (2023). The case for 4-bit precision: k-bit Inference Scaling Laws. In *40th International Conference on Machine Learning (ICML)*, pages 7750–7774.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. (2020). An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv:2010.11929*, pages 1–22.

Elnaggar, A., Heinzinger, M., Dallago, C., Rehawi, G., Wang, Y., Jones, L., Gibbs, T., Feher, T., Angerer, C., Steinegger, M., Bhowmik, D., and Rost, B. (2021). ProtTrans: Toward understanding the language of life through self-supervised learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):7112–7127.

Friedberg, I., Radivojac, P., Paolis, C. D., Piovesan, D., Joshi, P., Reade, W., and Howard, A. (2023). CAFA 5 Protein Function Prediction.

Jin, H., Chollet, F., Song, Q., and Hu, X. (2023). AutoKeras: An AutoML Library for Deep Learning. *Journal of Machine Learning Research*, 24(6):1–6.

Kulmanov, M. and Hoehndorf, R. (2019). DeepGOPlus: Improved Protein Function Prediction from Sequence. *Bioinformatics*, 36(2):422–429.

Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C. H., and Kang, J. (2020). BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Lin, Z., Akin, H., Rao, R., Hie, B., Zhu, Z., Lu, W., Smetanin, N., Verkuil, R., Kabeli, O., Shmueli, Y., Costa, A. d. S., Fazel-Zarandi, M., Sercu, T., Candido, S., and Rives,

A. (2023). Evolutionary-Scale Prediction of Atomic-Level Protein Structure with a Language Nodel. *Science*, 379(6637):1123–1130.

Loshchilov, I. and Hutter, F. (2017). Decoupled Weight Decay Regularization. *arXiv:1711.05101*, pages 1–19.

Oliveira, G. B., Pedrini, H., and Dias, Z. (2023). TEMPROT: Protein Function Annotation using Transformers Embeddings and Homology Search. *BMC Bioinformatics*, 24(1):1–16.

Oliveira, G. B., Pedrini, H., and Dias, Z. (2024). Integrating Transformers and AutoML for Protein Function Prediction. In *46th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 1–5. IEEE.

Radivojac, P. (2013). A (not so) quick introduction to protein function prediction. *Indiana University, USA*.

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

Valentini, G. (2010). True Path Rule Hierarchical Ensembles for Genome-Wide Gene Function Prediction. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 8(3):832–847.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *30th Advances in Neural Information Processing Systems (NeurIPS)*, pages 5998–6008.

Vitale, R., Bugnon, L. A., Fenoy, E. L., Milone, D. H., and Stegmayer, G. (2024). Evaluating large language models for annotating proteins. *Briefings in Bioinformatics*, 25(3):bbae177.

Yeung, W., Zhou, Z., Li, S., and Kannan, N. (2023). Alignment-free estimation of sequence conservation for identifying functional sites using protein sequence embeddings. *Briefings in Bioinformatics*, 24(1):bbac599.

Yu, Y., Yang, C.-H. H., Kolehmainen, J., Shivakumar, P. G., Gu, Y., Ren, S. R. R., Luo, Q., Gourav, A., Chen, I.-F., Liu, Y.-C., Dinh, T., Gandhe, A., Filimonov, D., Ghosh, S., Stolcke, A., Rastow, A., and Bulyko, I. (2023). Low-rank adaptation of large language model rescoring for parameter-efficient speech recognition. In *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 1–8. IEEE.

Zhapa-Camacho, F., Tang, Z., Kulmanov, M., and Hoehndorf, R. (2024). Predicting protein functions using positive-unlabeled ranking with ontology-based priors. *bioRxiv*, pages 1–9.

Zhou, N., Jiang, Y., Bergquist, T. R., Lee, A. J., Kacsoh, B. Z., Crocker, A. W., Lewis, K. A., Georghiou, G., Nguyen, H. N., Hamid, M. N., Davis, L., Dogan, T., Atalay, V., Rifaioglu, A. S., Dalkıran, A., Cetin Atalay, R., Zhang, C., Hurto, R. L., Freddolino, P. L., Zhang, Y., Bhat, P., Supek, F., Fernández, J. M., Gemovic, B., Perovic, V. R., Davidović, R. S., Sumonja, N., Veljkovic, N., Asgari, E., Mofrad, M. R., Profiti, G., Savojardo, C., Martelli, P. L., Casadio, R., Boecker, F., Schoof, H., Kahanda, I., Thurlby, N., McHardy, A. C., Renaux, A., Saidi, R., Gough, J., Freitas, A. A., Antczak, M., Fabris, F., Wass, M. N., Hou, J., Cheng, J., Wang, Z., Romero, A. E., Paccanaro,

A., Yang, H., Goldberg, T., Zhao, C., Holm, L., Törönen, P., Medlar, A. J., Zosa, E., Borukhov, I., Novikov, I., Wilkins, A., Lichtarge, O., Chi, P.-H., Tseng, W.-C., Linial, M., Rose, P. W., Dessimoz, C., Vidulin, V., Dzeroski, S., Sillitoe, I., Das, S., Lees, J. G., Jones, D. T., Wan, C., Cozzetto, D., Fa, R., Torres, M., Warwick Vesztrocy, A., Rodriguez, J. M., Tress, M. L., Frasca, M., Notaro, M., Grossi, G., Petrini, A., Re, M., Valentini, G., Mesiti, M., Roche, D. B., Reeb, J., Ritchie, D. W., Aridhi, S., Alborzi, S. Z., Devignes, M.-D., Koo, D. C. E., Bonneau, R., Gligorijević, V., Barot, M., Fang, H., Toppo, S., Lavezzo, E., Falda, M., Berselli, M., Tosatto, S. C., Carraro, M., Piovesan, D., Ur Rehman, H., Mao, Q., Zhang, S., Vucetic, S., Black, G. S., Jo, D., Suh, E., Dayton, J. B., Larsen, D. J., Omdahl, A. R., McGuffin, L. J., Brackenridge, D. A., Babbitt, P. C., Yunes, J. M., Fontana, P., Zhang, F., Zhu, S., You, R., Zhang, Z., Dai, S., Yao, S., Tian, W., Cao, R., Chandler, C., Amezola, M., Johnson, D., Chang, J.-M., Liao, W.-H., Liu, Y.-W., Pascarelli, S., Frank, Y., Hoehndorf, R., Kulmanov, M., Boudellioua, I., Politano, G., Di Carlo, S., Benso, A., Hakala, K., Ginter, F., Mehryary, F., Kaewphan, S., Björne, J., Moen, H., Tolvanen, M. E., Salakoski, T., Kihara, D., Jain, A., Šmuc, T., Altenhoff, A., Ben-Hur, A., Rost, B., Brenner, S. E., Orengo, C. A., Jeffery, C. J., Bosco, G., Hogan, D. A., Martin, M. J., O'Donovan, C., Mooney, S. D., Greene, C. S., Radivojac, P., and Friedberg, I. (2019). The CAFA challenge reports improved protein function prediction and new functional annotations for hundreds of genes through experimental screens. *Genome Biology*, 20(1):244.

Zhou, Z., Ji, Y., Li, W., Dutta, P., Davuluri, R., and Liu, H. (2023). DNABERT-2: Efficient foundation model and benchmark for multi-species genomes. *arXiv:2306.15006*, pages 1–23.

Zhu, Y.-H., Zhang, C., Yu, D.-J., and Zhang, Y. (2022). Integrating Unsupervised Language Model with Triplet Neural Networks for Protein Gene Ontology Prediction. *PLoS Computational Biology*, 18(12):e1010793.