

Integrating Functional Knowledge into Protein Design: A Novel Approach to Tokenization and Noise Injection for Function-Aware Protein Language Models

by

Adrina Tang

Bachelor of Science in Artificial Intelligence and Decision-Making, MIT, 2025

Submitted to the Department of Electrical Engineering and Computer Science in partial fulfillment of the requirements for the degree of

Master of Engineering in Electrical Engineering and Computer Science

at the

Massachusetts Institute of Technology

May 2025

© Adrina Tang, 2025. All rights reserved.

The author hereby grants to MIT a nonexclusive, worldwide, irrevocable, royalty-free license to exercise any and all rights under copyright, including to reproduce, preserve, distribute, and publicly display copies of the thesis, or release the thesis under an open-access license.

Authored by: **Adrina Tang**

Department of Electrical Engineering and Computer Science
May 2025

Certified by: **Bonnie Berger**

Department of Mathematics
Thesis supervisor

Accepted by: **Katrina LaCurts**

Chair
Master of Engineering Thesis Committee

This page is intentionally left blank

Integrating Functional Knowledge into Protein Design: A Novel Approach to Tokenization and Noise Injection for Function-Aware Protein Language Models

by

Adrina Tang

Submitted to the Department of Electrical Engineering and Computer Science on
May 12th, 2025 in partial fulfillment of the requirements for the degree of
Master of Engineering in Electrical Engineering and Computer Science

ABSTRACT

Designing novel proteins with specific biological functions remains a fundamental challenge in computational biology. While recent advances in protein language models have enabled powerful sequence-based representations, most models, including state-of-the-art systems like ESM3, fall short in effectively encoding functional context during protein generation. In this work, we present a multimodal protein co-design framework that conditions sequence generation on fine-grained functional annotations, specifically leveraging residue-level Gene Ontology (GO) term labels on sequences from the UniRef100 database. By explicitly associating functional signals with residue elements of proteins, our model learns to generate function-conditioned protein sequences that are biologically plausible and semantically consistent. Unlike prior approaches, which treat function as a secondary feature or a classification task, our method focuses on joint reasoning over function and sequence during the design process. This closes a critical gap in the current landscape of protein design tools, offering a scalable and generalizable approach to co-designing protein sequences with user-specified functional profiles.

Thesis Supervisor: Bonnie Berger
Title: Simons Professor of Mathematics

This page is intentionally left blank

Acknowledgements

I want to thank my mentors, Bowen Jing and Anna Sappington, for their guidance and mentorship throughout my Masters studies. I would also like to thank my advisor, Professor Bonnie Berger, for providing the opportunity to work with and learn from such a talented and insightful group of people, as well as the resources to carry out our ambitious work.

I would also like to thank Sam Sledzieski, for first welcoming me to the Berger Lab and inspiring my research in the realm of protein design and modeling, and Anand Chandrasekar, for being my first research mentor and opening my eyes to what computation can do for biology and health care. To all the professors and teaching assistants who have helped me in my academic journey, a huge thank you to you all as well.

To all my friends who have accompanied me in the past few years and spent countless late nights with me doing more talking than studying, the biggest thanks goes out to you as well. Finally, thank you to my family—without whom I would not be who I am today.

This page is intentionally left blank

Contents

Title	1
Abstract	3
Acknowledgements	5
Table of Contents	7
List of Figures	9
List of Tables	11
1 Introduction	12
1.1 Motivation	12
1.2 Related Works	15
1.2.1 Protein Language Models and ESM3	15
1.2.2 Structure Prediction and Structure-Conditioned Design	16
1.2.3 Functional Prediction Models	17
1.2.4 Annotation Resources: InterPro and Gene Ontology	18
2 Methods	19
2.1 Data	19
2.1.1 Uniref	19
2.1.2 InterPro	20
2.1.3 Gene Ontology	21
2.2 Model and Sequence Generation	23
2.2.1 Model Architecture	23
2.2.2 Noising and Denoising	25
2.3 Function Conditioning	26

2.3.1	ESM3 Function Tokenization	26
2.3.2	GO Function Tokenization	28
2.4	Sampling for Sequence Generation	29
2.5	Evaluation	32
2.5.1	Folding Confidence	32
2.5.2	Sequence Entropy	33
2.5.3	DeepFRI	35
3	Discussion	37
3.1	Results	37
3.1.1	Structural Integrity	37
3.1.2	Sequence Diversity	39
3.1.3	Functional Adherence	40
3.2	Key Findings	41
3.3	Limitations	43
4	Future Work	45
4.1	Structure Track Implementation	45
4.2	Feedback-Driven Protein Generation	46
4.3	Encoding Functional Semantics	46
4.4	Contextualizing Functional Origin	47
4.5	Incorporating Active Site Embeddings	47
4.6	Toward Experimentally Viable Proteins	47
A	Code Availability & Compute Resources	50
B	Model Hyperparameters	51
C	Additional Results	52
	Bibliography	55

List of Figures

2.1	Example of GO term inheritance and GO function annotations from InterProScan on random peptide chain.	21
2.2	Distribution of MF GO terms in the GO ontology and their distance from MF root term.	22
2.3	Distribution of MF GO term annotations and their distance from MF root term for a sample of 6 Million UniRef100 sequence.	23
2.4	Model architecture, consisting of two parallel tracks for sequence and function, followed by transformer blocks with multi-model geometric attention heads.	24
2.5	Example of ESMFold evaluation on sequence chunk taken from Uniref100. Sequence shown is a component of the acetyl coenzyme A carboxylase (ACC) complex [1].	33
3.1	pLDDT evaluation against baseline. Baseline metrics calculated from a sample of naturally occurring proteins in the UniRef100 dataset . . .	38

3.2	Sequence Entropy evaluation results against baseline. Baseline metrics calculated from a sample of naturally occurring proteins in the UniRef100 dataset	39
3.3	DeepFRI prediction results as a percentage of baseline. Baseline metrics calculated from a sample of naturally occurring proteins with similar annotations in the UniRef100 dataset	40
C.1	Loss and perplexity of models during training	52

List of Tables

2.1	Gene Ontology (GO) categories with their statistics and root nodes. .	22
3.1	pLDDT Values	38
3.2	Sequence Entropy Values	39
3.3	DeepFRI Metrics as Percentages of Baseline	41
B.1	Final hyperparameters for training and generation. These hyperpa- rameters are shared between the different conditioning methods and model dimensions examined.	51
C.1	Raw Comprehensive Performance Metrics	53

Chapter 1

Introduction

Proteins are the workhorses of life. They orchestrate nearly every biological process, from catalyzing chemical reactions to transporting molecules, maintaining cellular structures, and facilitating communication within and between cells. The sequence of amino acids that make up a protein defines its three-dimensional structure, and in turn, that structure determines the protein's function. Given this central role, the ability to design proteins with novel or enhanced functions has vast implications across medicine, biotechnology, and materials science [2].

1.1 Motivation

Despite its promise, protein design remains an exceptionally complex problem, often referred to as searching for functional needles in the astronomically large haystack of possible sequences. Traditional methods in the lab require intensive experimental iteration and domain-specific intuition. In recent years, the emergence of deep learning has offered new tools for navigating this space more efficiently and systematically, with the hope of making protein design more principled, scalable, and automated [3, 4].

Breakthroughs in structure prediction, most notably with AlphaFold2, have revolutionized how we understand protein folding [5]. By training deep neural networks on

co-evolutionary information and structural databases, AlphaFold2 demonstrated that it is possible to predict high-resolution protein structures directly from amino acid sequences. This achievement brought structural biology into the computational age and laid the foundation for a new wave of machine learning models aimed at understanding and engineering proteins.

In parallel, protein language models (PLMs) such as the ESM (Evolutionary Scale Modeling) series have adapted techniques from natural language processing to the protein domain [6, 7]. Trained on hundreds of millions of sequences from databases like UniRef, these models learn to represent sequences in high-dimensional embeddings that encode rich information about local structure, global fold, and even some functional signals. The ESM series includes ESMFold, another highly accurate structural predictor that uses the ESM model’s sequence embeddings to produce structures. The latest iteration, ESM3, goes beyond masked language modeling by incorporating generative capabilities, including the ability to condition generation on desired outputs such as structural templates or functional labels.

However, while ESM3 and related models represent a significant step toward function-conditioned protein design, their effectiveness in this task remains limited. The conditioning mechanisms are often restricted to global function tags, which do not capture the residue-level specificity required for many applications. Moreover, these models are typically trained on weak or implicit supervision of function, making it difficult to disentangle whether generated sequences truly encode the desired activity or merely approximate it based on learned priors. Consequently, while ESM3 can generate plausible sequences, its ability to precisely and controllably design for function remains an open challenge.

This thesis aims to address that gap. We introduce a multimodal co-design framework that conditions protein generation on fine-grained functional annotations, using Gene Ontology (GO) terms mapped directly to residues from the UniRef100 database [1]. This approach allows the model not only to generate sequences that conform to overall struc-

tural or evolutionary norms, but to explicitly embed functional intent into specific regions of the sequence. In contrast to previous models that treat function as a post-hoc classification task or a weak conditioning signal, our method tightly couples function and sequence as co-evolving modalities, enabling conditioned reasoning during the generative process [8].

Our model operates within a broader research effort in our lab to build OpenProt, a unified, large-scale multimodal protein foundation model. OpenProt is designed to simultaneously integrate sequence, structure, and function into a shared latent space, enabling the joint generation of structure and sequence with functional knowledge. While the work presented in this thesis focuses specifically on the function-sequence axis, it is architected to interface cleanly with ongoing efforts in structure modeling, including a parallel structure track that reasons over atomic coordinates and distance maps. The long-term vision is a flexible, general-purpose model that can perform any-to-any translation across the protein modalities.

In this context, the specific contribution of this thesis is the design, implementation, and evaluation of a function-conditioned generative model for protein sequences. Our model is trained to jointly model residue-level function labels and amino acid sequences, using a variant of the transformer architecture adapted for multimodal input [9]. We describe the data preprocessing pipeline used to extract residue-level GO annotations from UniRef100 [1], the encoding decisions involved in integrating functional representations, the loss functions used to encourage alignment between generated sequences and target functions, and the iterative sampling process to produce a new protein. Through both quantitative metrics and other pre-trained model, we evaluate the structural integrity, sequence diversity, and functional adherence of our generated proteins.

By embedding function directly into the generation process, we enable a more powerful and controllable form of protein design. This work advances the frontier of generative protein modeling by introducing methods that explicitly condition on biological intent,

laying the groundwork for future systems that can co-design sequences, structures, and functions in a unified framework.

The remainder of this thesis is organized as follows. Chapter 1 provides background on protein modeling, including existing approaches to structure prediction, sequence generation, and function annotation. Chapter 2 describes the methods involved in this work, including dataset construction and preprocessing pipeline, model architecture, sampling process, and evaluation metrics. Chapter 3 presents evaluation results, including comparisons to existing methods, key findings, and limitations. Chapter 4 discusses future directions, including integration into the broader OpenProt framework. We conclude with reflections on the role of multimodal models in the future of computational protein design.

1.2 Related Works

Recent advances in deep learning have dramatically reshaped our ability to model, predict, and design proteins. Particularly influential are protein language models (PLMs), which leverage large-scale unlabeled data to learn contextual representations of amino acid sequences [10, 11, 12]. However, despite successes in generative modeling, structure prediction, and function classification, few models to date have effectively co-designed protein sequences with fine-grained functional intent. This section reviews the most relevant work in protein modeling, spanning language models, structure-aware generation, functional annotation, and biological databases.

1.2.1 Protein Language Models and ESM3

Protein language models treat amino acid sequences analogously to natural language, learning distributed representations of residues through self-supervised training objectives. Among these, ESM3 stands out as a state-of-the-art multimodal model, capable of integrating multiple input tracks—sequence, structure, MSA, and functional informa-

tion—for tasks including masked token prediction, function classification, and sequence generation [7].

A central innovation of ESM3 is its support for function-conditioned sequence generation. Unlike earlier PLMs, ESM3 enables residue-level functional annotations, allowing the model to modulate sequence design based on localized functional information. Functional annotations are integrated via TF-IDF-style embeddings of free-text keywords derived from curated descriptions, providing the model with some notion of functional context across residues.

However, this approach introduces important limitations that motivate this study. The TF-IDF-based keyword embeddings, while computationally tractable, are semantically coarse and often ambiguous. Many keywords lack specificity (e.g., "catalytic," "binding") or suffer from redundancy and inconsistency. Furthermore, because these labels are not grounded in structured ontologies like Gene Ontology (GO), they do not reflect well-defined biological hierarchies or relationships. As a result, ESM-3's function conditioning mechanism struggles to precisely localize and differentiate functional intent.

In contrast, our model builds on this by employing GO-term annotations at the residue level, derived through alignment with the InterPro database [13]. This allows the model to condition generation on biologically curated, hierarchically structured functional terms—providing both greater interpretability and functional specificity.

1.2.2 Structure Prediction and Structure-Conditioned Design

The three-dimensional structure of a protein is intimately tied to its function, and accurate structure modeling has become a cornerstone of modern protein design. AlphaFold2 transformed the field with a deep attention-based architecture capable of predicting near-experimental-resolution protein structures directly from sequence [5]. Though non-generative, AlphaFold's ability to assess the foldability of designed proteins has made it a key evaluative tool in design pipelines.

On the generative side, RFdiffusion reimagines structure generation as a diffusion process, progressively denoising 3D coordinates toward desired conformations [14]. RFdiffusion can generate proteins de novo or conditionally fold sequences into target geometries. However, it does not incorporate explicit functional conditioning, and downstream function often relies on heuristics or post hoc evaluation.

Other structure-aware models like RoseTTAFold and DPLM/DPLM2 aim to learn shared sequence-structure embeddings [15, 16]. These models support structure-based transfer learning, stability prediction, and structure-conditioned generation. Yet across these models, the integration of biological function as a first-class modeling objective remains limited.

This work addresses this gap by focusing on function as a conditioning signal during generation, rather than as a classifier. Moreover, it contributes to a larger research effort within our lab to build OpenProt, a multimodal protein model that integrates sequence, structure, and function into a unified generative framework. While the work in this thesis focuses on the function-sequence axis, future iterations of OpenProt will incorporate structure tracks for end-to-end co-design across all three modalities.

1.2.3 Functional Prediction Models

A related but distinct body of work has emerged around function prediction: inferring functional roles or annotations from sequence or structure. These models aim to map structures or sequences to standardized function labels—usually GO terms—and provide a diagnostic view of protein biology.

One prominent example is DeepFRI [17], which uses graph convolutional networks on protein structure graphs to predict GO terms. DeepFRI learns localized residue-function relationships by modeling spatial proximity, achieving strong performance on novel fold families. They can also operate on sequence inputs, by inferring its structural translation.

Other models, such as NetGO [18], use transformer-based or LSTM-based architec-

tures trained on protein-function pairs, often combining alignment-based features (e.g., BLAST hits) with deep learning embeddings. DeepGOPlus [19], for instance, integrates CNNs with sequence homology features to improve multi-label GO term prediction.

These models demonstrate that function can be inferred with high accuracy, especially when structural or evolutionary context is available. However, they are discriminative rather than generative—they do not enable sequence design—and therefore operate more as evaluative tools than design agents. Importantly, their success underscores the tractability of function annotation, validating the use of GO terms as meaningful conditioning signals in generative frameworks like the one proposed here [20, 21, 22].

1.2.4 Annotation Resources: InterPro and Gene Ontology

High-quality annotations are essential for both prediction and generative tasks. This work leverages two widely used and complementary biological resources:

The Gene Ontology (GO) [23] provides a hierarchical, multi-label vocabulary for protein function, structured across three axes: molecular function, biological process, and cellular component. GO terms are standardized, curated, and interpretable, making them ideal conditioning labels for controlled protein design.

InterPro aggregates predictive models from domain databases like Pfam [24], SMART, and PROSITE [25] to provide domain- and residue-level annotations for protein families. InterPro’s mappings allow for residue-specific alignment of GO terms, bridging the gap between global protein function and local residue-level roles.

By aligning InterPro’s functional domains with GO’s semantic structure, this work constructs a training set that enables residue-level function conditioning, improving both interpretability and biological plausibility in the generative model.

Chapter 2

Methods

This section outlines the methodology used in this study, divided into three main components: data acquisition and preprocessing, model design and training, and function conditioning strategies for function-aware protein co-design. The goal is to enable localized, function-conditioned generation of protein sequences by leveraging curated ontological annotations, evaluated through a downstream prediction model.

2.1 Data

2.1.1 Uniref

The UniRef100 database, maintained by the UniProt Consortium, serves as a foundational dataset in this study [1]. UniRef100 clusters identical ('100' percent similarity) sequences and sub-fragments across all known organisms into single representative entries, creating a non-redundant dataset that offers high coverage of protein sequence space. Each entry in UniRef100 includes metadata such as UniProtKB accessions, protein descriptions, and taxonomy identifiers. Because of its comprehensiveness and standardization, UniRef100 is a common choice for training deep learning models on protein data.

The rationale for using UniRef100 over more heavily clustered sets like UniRef90 or

UniRef50 is to preserve the fine-grained diversity and specificity of protein sequences—important when the goal is to capture subtle functional differences at the residue level. Furthermore, the UniRef100 database at the time of this study has the largest amount of unique sequences, containing around 435 million sequences, of which we were able to annotate 265 million with GO terms.

2.1.2 InterPro

To annotate protein sequences in UniRef100 with functional domain-level information, we use InterProScan [13]. InterProScan is a software package developed by the European Bioinformatics Institute (EBI) that scans protein sequences against predictive models from multiple member databases, including Pfam [24], PROSITE [25], SMART, TIGRFAMs, PRINTS, and others. These member databases each provide complementary types of information, from structural domains to functional motifs.

InterPro itself is an integrative database that consolidates these domain annotations, mapping them to standard identifiers and connecting them to known biological functions. One of InterPro’s most valuable features is its cross-referencing to the Gene Ontology (GO) database via the InterPro2GO mapping. This mapping provides a systematic link between conserved protein domains and known biological roles.

By running InterProScan on UniRef100 entries, we can obtain rich annotations that localize function to specific residues or domains. For each matched domain, InterProScan returns its start and end residue indices, an InterPro accession (e.g., IPR000001), and any mapped GO function terms, if available. Not all InterPro entries map to a GO term, but about 60% of sequences in UniRef were annotated with GO terms when ran against InterPro scan. These domain-residue annotations are crucial for training the function conditioning module described later.

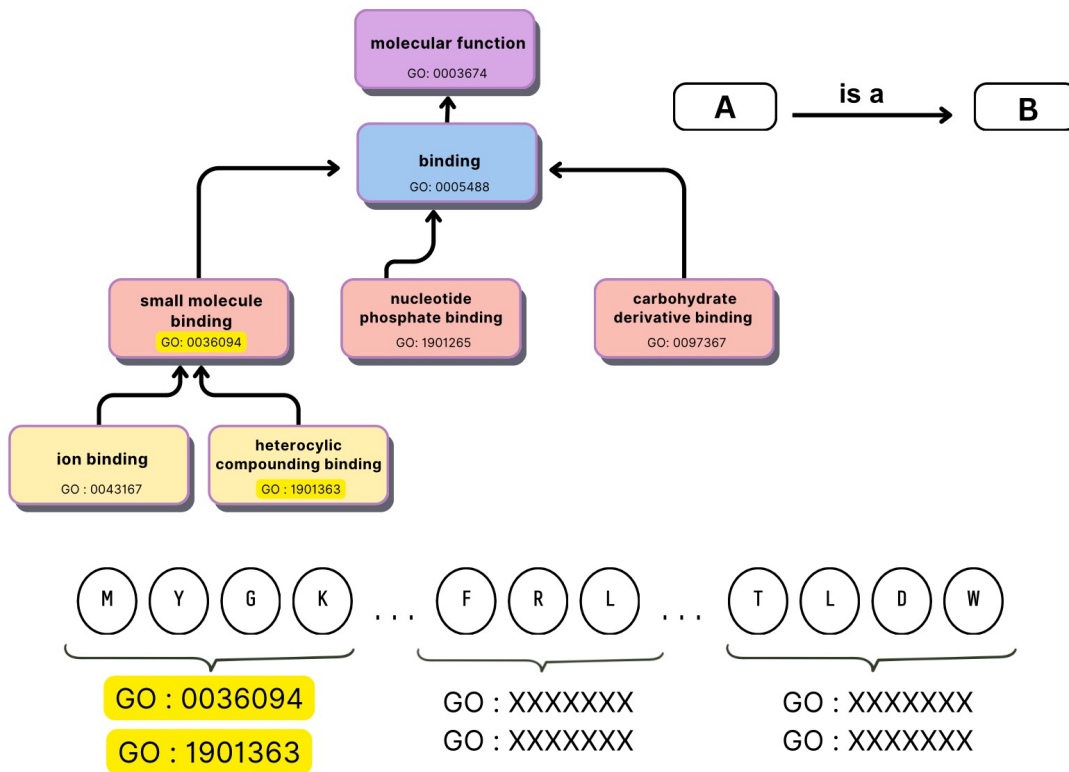


Figure 2.1: Example of GO term inheritance and GO function annotations from Inter-ProScan on random peptide chain.

2.1.3 Gene Ontology

The Gene Ontology (GO) project provides a controlled vocabulary to describe protein function across species [23]. It consists of three main branches:

- Molecular Function (MF): The biochemical activity of a protein (e.g., ATP binding).
- Biological Process (BP): Larger biological goals accomplished by molecular functions (e.g., DNA replication).
- Cellular Component (CC): The subcellular location or macromolecular complex where the protein is active (e.g., ribosome, cytoplasm).

GO is structured as a directed acyclic graph (DAG). Each term can have multiple parents and children, with relationships such as `is_a` (inheritance), `part_of`, and `regulates`. This structure allows for flexible and nuanced annotation, supporting both fine-grained and high-level descriptions of protein function.

GO Category	Number of Terms	Average Depth	Root
Molecular Function (MF)	~11,000	~6	GO:0003674
Biological Process (BP)	~30,000	~8	GO:0008150
Cellular Component (CC)	~3,000	~4	GO:0005575

Table 2.1: Gene Ontology (GO) categories with their statistics and root nodes.

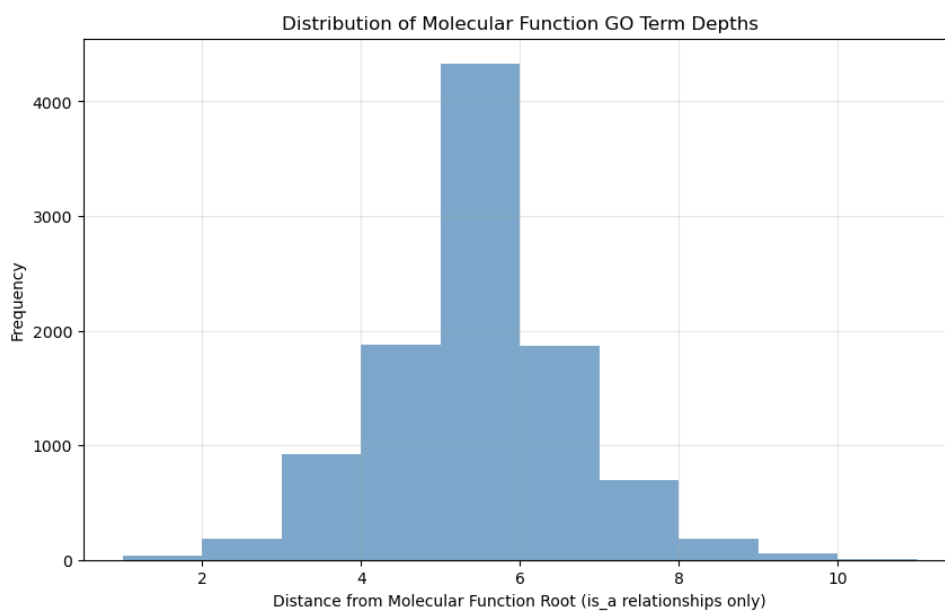


Figure 2.2: Distribution of MF GO terms in the GO ontology and their distance from MF root term.

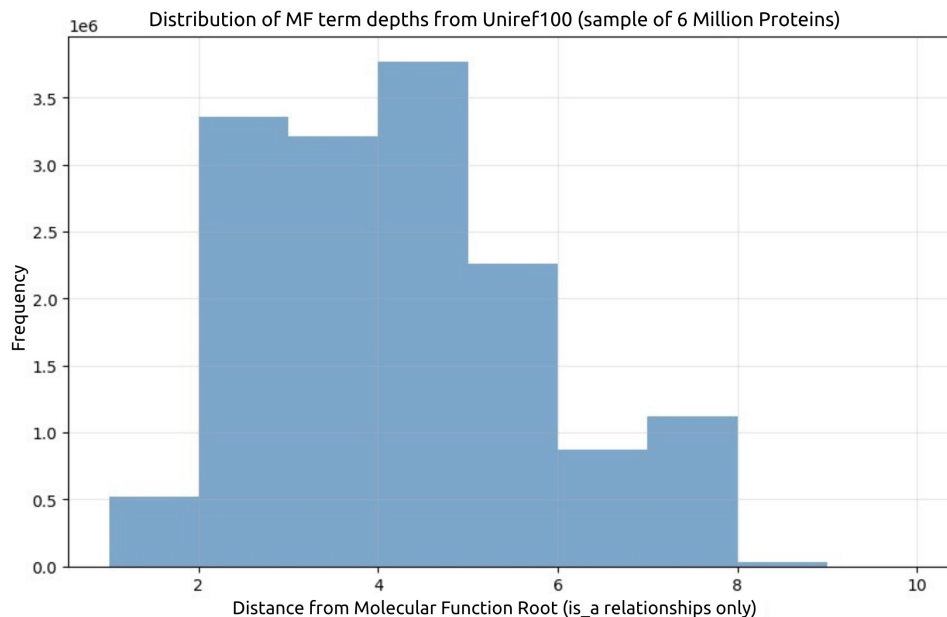


Figure 2.3: Distribution of MF GO term annotations and their distance from MF root term for a sample of 6 Million UniRef100 sequence.

2.2 Model and Sequence Generation

2.2.1 Model Architecture

The model follows a multimodal transformer architecture designed for conditional protein generation, inspired by recent advances in protein language models [6, 7]. It begins by processing input proteins through two parallel encoding tracks: one for the amino acid sequence and one for functional annotations.

First, masking noise is applied over the input representations at random and with a configurable distribution and level. The model’s track is to then attempt to recover it downstream. The sequence track encodes the input sequence using one-hot vectors for the vocabulary of amino acids with a mask over unknown residues, and produces residue-level embeddings. Simultaneously, the function track encodes residue-level Gene Ontology (GO) annotations into a learned functional embedding space. These embeddings are then

aggregated to form the input representation for the bulk of the model.

To capture dependencies across residue positions in the aggregated representation, we also produce a pairwise encoding that is passed into the transformer blocks. The combination of the original embeddings and the pairwise encodings is passed through a series of transformer blocks, each composed of layer normalization, geometric multimodal attention heads, and feed-forward layers with dropout [26, 27]. The attention mechanism enables the model to jointly reason over sequence and function information.

At generation time, an optional function condition vector of GO information can be injected to guide sampling toward a specific functional profile. The final output is an iteratively sampled protein sequence that is consistent with the target function annotation, enabling controllable and biologically informed protein design [2].

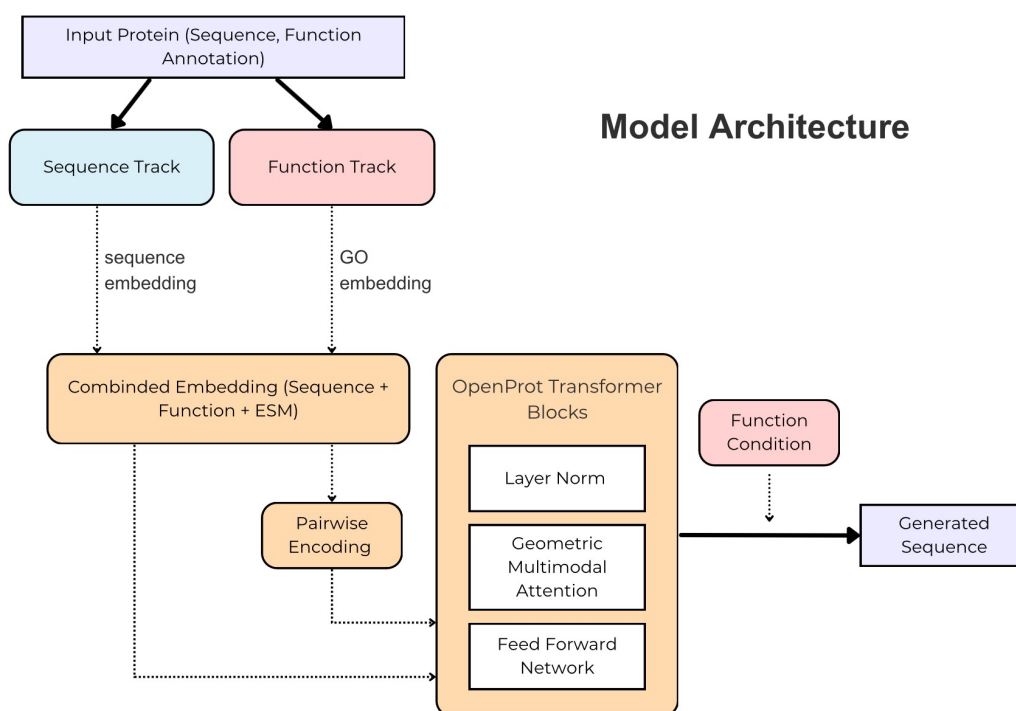


Figure 2.4: Model architecture, consisting of two parallel tracks for sequence and function, followed by transformer blocks with multi-model geometric attention heads.

2.2.2 Noising and Denoising

To train the model on masked token prediction, we apply a custom corruption scheme to the input sequence. This corruption combines elements of dropout, masked language modeling (MLM), and optional token reweighting, applied only to protein sequences. Our approach shares conceptual similarities with denoising techniques used in recent protein generative models [4, 16].

Corruption is performed through a dedicated function that outputs both a corrupted input and a corresponding supervision mask. The process involves dropout-style masking, where each token in the protein sequence may be replaced by a special mask token with a predefined probability. Supervision is only applied to these masked positions. Additionally, we implement Masked Language Modeling(MLM)-style substitution where tokens may be randomly selected for corruption with a certain probability [28]. These selected tokens are replaced by uniformly sampled amino acids with another probability parameter. These positions are also supervised and weighted more heavily via a configurable weight parameter.

We also incorporate optional reweighting strategies for supervision targets, scaling them by either the inverse of the noise probability or linearly with the noise probability, depending on the configuration. This is applied only to dropout-masked positions.

In addition to sequence corruption, we apply dropout to the functional annotations, implementing both keyword dropout and annotation dropout for each residue with probability p . The model’s objective is to reconstruct the original, uncorrupted sequence given the partially masked input and the functional annotation. Let x denote the original amino acid sequence and \tilde{x} . The model learns the conditional likelihood $p_{\theta}(x | \tilde{x}, f)$ where f represents the functional annotation. Since each amino acid prediction is a classification problem across 20 possible classes, we use cross-entropy loss. The objective of training is then to minimize:

$$\mathcal{L}_{\text{denoise}} = - \sum_{i \in \text{masked}} \log p_{\theta}(x_i | \tilde{x}, f) \quad (2.1)$$

This denoising formulation enables the model to internalize functional constraints during the reconstruction process, reinforcing the association between sequence motifs and their functional roles. It also regularizes the model to avoid overfitting to spurious correlations in the data, improving generalization in downstream sequence generation tasks.

2.3 Function Conditioning

We evaluate function conditioning in the model on two different methods—ESM3’s tokenization method for function annotations as a baseline, and then our method of a pure GO term informed encoding.

2.3.1 ESM3 Function Tokenization

The function conditioning mechanism implemented in ESM3 [7] provides one of the only large-scale baselines for function-aware protein sequence generation currently available. ESM3’s conditioning pipeline does not rely on an explicit ontology graph structure, but instead begins by leveraging free-text annotations drawn from InterPro entry descriptions and Gene Ontology (GO) term names. In this setup, functional context is extracted as a flat corpus of textual information, without explicitly encoding hierarchical relationships between biological functions.

To reproduce ESM3’s function conditioning approach, we first annotated each UniRef100 protein sequence using InterProScan [13]. For each sequence, any matched InterPro entries were identified, and their corresponding text descriptions and associated GO term names were collected. These textual annotations were processed to build a vocabulary of unigrams and bigrams, resulting in a feature space of size V (where V denotes the

total number of extracted tokens). This vocabulary excludes a manually-formed list of particularly common and non-informative words, including 'domain', 'protein', etc.

Each annotation segment was then converted into a term frequency–inverse document frequency (TF-IDF) representation [29]. Specifically, for a given token t in annotation d , the TF-IDF weight $w_{t,d}$ was computed as

$$w_{t,d} = \text{tf}(t, d) \times \log \left(\frac{N}{\text{df}(t)} \right),$$

where $\text{tf}(t, d)$ is the frequency of token t in document d , $\text{df}(t)$ is the number of documents containing t , and N is the total number of documents. These TF-IDF vectors were propagated to every residue covered by the corresponding annotation. In cases where multiple annotations overlapped at a single residue, a max-pooling operation was applied element-wise across the TF-IDF vectors to resolve the overlap, ensuring that the strongest functional signal was retained.

Following TF-IDF construction, local sensitivity hashing (LSH) was applied to discretize the continuous vectors into tokenized function embeddings [30]. Eight independent sets of random hyperplanes were sampled from the TF-IDF feature space, each with dimensionality V , and residues were projected onto these hyperplanes to produce binary hash codes. Each 8-bit binary vector was then interpreted as an integer between 0 and 255, yielding a compact vocabulary of LSH tokens. Two additional special tokens were reserved to represent masked and padded residues. Critically, the process was repeated eight times with different random hyperplane initializations, producing an eight-dimensional LSH representation for every residue in the sequence.

While this reproduction of ESM3’s function tokenization serves as an important baseline, we observed that its reliance on TF-IDF representations that introduce noise, as common but biologically insignificant words from InterPro descriptions may dominate the feature space despite offering little discriminative power for precise function targeting. Furthermore, the resulting vocabulary of unigrams and bigrams becomes extremely

large ($V \gg 10^4$), increasing sparsity and the risk of token collisions after LSH compression. The lack of structured knowledge from the Gene Ontology graph (such as parent-child relationships or molecular function specificity) also means that conditioning signals may be semantically diluted. These factors make it difficult to guarantee that generation conditioned via ESM3 tokens will align tightly with a desired biological function.

2.3.2 GO Function Tokenization

To develop a more biologically grounded conditioning mechanism, we implemented a method based exclusively on Gene Ontology (GO) term annotations [23]. Unlike free-text-based approaches, our method uses the structure of the GO graph to encode molecular functions in a semantically meaningful way.

We begin by restricting the vocabulary to a curated subset of approximately 8,000 GO terms belonging to the Molecular Function branch and appearing in our data, ensuring that conditioning focuses directly on protein activity rather than broader biological processes or cellular localization. Each GO term within this subset is assigned a unique index in the vocabulary. To capture hierarchical information, we traverse the GO ontology graph to identify all ancestor terms for each selected GO term, excluding the root node Molecular Function, which is trivially shared among all entries.

For each protein sequence, we retrieve all associated GO term labels from the dataset annotations. Each label is then expanded to include its ancestor terms based on the ontology structure. Thus, each sequence is characterized not only by its direct annotations but also by the broader functional categories it falls under, promoting richer functional signals.

Residue-level conditioning vectors are then constructed as follows: for each residue, we assign a vector of at most M integers, where each integer corresponds to the vocabulary index of either a directly annotated GO term or one of its ancestors. Formally, for each residue r , we define a label set $\{g_1, g_2, \dots, g_M\}$, where $g_i \in \{1, 2, \dots, 10,000\}$ denotes

the index of a GO term present for that residue. Padding is applied if the number of associated terms is less than M .

These discrete GO term indices are subsequently passed through a learned embedding layer, producing a dense vector representation for each residue’s functional context. This embedding is incorporated directly into the model input alongside the amino acid sequence representation, allowing the network to condition generation on both sequence and structured function information.

Compared to free-text tokenization methods, our approach ensures that functional conditioning is precise, interpretable, and grounded in the formal structure of biological knowledge. By explicitly modeling both direct and ancestral GO terms, we enable the model to access varying levels of functional specificity during generation.

2.4 Sampling for Sequence Generation

We implement an iterative sampling approach to generate sequences conditioned on functional annotations, drawing inspiration from recent advances in protein generative modeling [9, 16].

Our sampling procedure exploits the model’s learned distribution over amino acid sequences conditioned on functional annotations. Starting from a fully masked sequence, we iteratively remove noise from the sequence until it is completely unmasked.

The sampling process proceeds as follows:

1. Initialize a sequence of mask tokens with length corresponding to the desired protein.
2. Compute the conditional probability distribution over amino acids at each masked position given the current partial sequence and the functional annotation.
3. Select positions to unmask based on confidence scores.

4. Sample amino acids for the selected positions according to their probability distributions.
5. Update the sequence by replacing the selected mask tokens with their sampled amino acids.
6. Repeat steps 2-5 until all positions are filled or a maximum number of iterations is reached.

A key aspect of our approach is the confidence-based selection of positions to unmask at each iteration. For each position, we compute a score based on the model’s prediction confidence. Given the logits produced by the model for each position, we calculate the probability distribution:

$$p_{\theta}(x_i|\tilde{x}, f) = \text{softmax}(\text{logits}_i/\tau) \tag{2.2}$$

where τ is the temperature parameter that controls the sharpness of the distribution.

To introduce stochasticity into the unmask selection process, we apply Gumbel noise to the scores [31]:

$$\text{scores}_i = \log p_{\theta}(x_i|\tilde{x}, f) + \tau_{\text{topk}} \cdot g_i \tag{2.3}$$

where g_i is drawn from a Gumbel distribution and τ_{topk} is a temperature parameter specifically for the unmask selection process.

The number of positions to unmask at each step follows a predefined schedule that gradually decreases the number of masked tokens:

$$n_{\text{mask}}(t) = \lfloor t \cdot L \rfloor \tag{2.4}$$

where $t \in [0, 1]$ is the progress through the generation process, L is the total sequence length, and $n_{\text{mask}}(t)$ is the number of positions that should remain masked at time t . We

begin by unmasking many positions in the early steps to provide the model with a rough scaffold, then gradually decrease the number of unmasked positions to allow fine-tuning of specific residues as the sequence converges.

For already unmasked positions, we can adjust their scores to either maintain the current amino acids or allow them to be revised. In the case where we permit revisions, we compute adjusted probabilities:

$$\tilde{p}_\theta(x_i|\tilde{x}, f) = \frac{p_\theta(x_i|\tilde{x}, f)}{0.5 \cdot \mathbf{1}[x_i = \tilde{x}_i] + 0.05} \quad (2.5)$$

This adjustment increases the probability of selecting amino acids different from the current ones, promoting exploration of the sequence space.

When sampling amino acids for the selected positions, we use either standard categorical sampling:

$$x_i \sim \text{Categorical}(\text{logits}_i/\tau) \quad (2.6)$$

or Gumbel-max sampling:

$$x_i = \arg \max_a (\log p_\theta(a|\tilde{x}, f) + g_a)/\tau \quad (2.7)$$

where g_a is Gumbel noise specific to each amino acid option.

Temperature annealing can be applied over the course of generation [32]:

$$\tau(t) = \tau_{\text{start}} \cdot t + (1 - t) \cdot \tau_{\text{end}} \quad (2.8)$$

This typically starts with higher values to encourage exploration and ends with lower values to refine the sequence.

The final generated sequences reflect both the amino acid preferences learned from the training data and the specific functional constraints provided by the GO term annota-

tions. By conditioning on functional annotations during generation, our model produces sequences that are not only biologically plausible but also tailored to satisfy the desired functional specifications.

2.5 Evaluation

Our evaluation methodology integrates structural, sequence-based, and functional analyses, each intended to capture complementary aspects of generation fidelity. This multi-pronged approach was motivated by the need to validate not only the foldability of the generated sequences, but also their compositional realism and adherence to specific functional targets derived from Gene Ontology (GO) terms.

2.5.1 Folding Confidence

To evaluate the structural plausibility of the generated sequences, we predicted their three-dimensional structures using ESMFold, a state-of-the-art structure prediction model based on large protein language model (pLM) embeddings. ESMFold leverages evolutionary-scale pretraining on millions of natural protein sequences to directly infer spatial relationships between residues, bypassing the need for multiple sequence alignments or template-based modeling.

For each generated sequence, we applied ESMFold to predict its atomic coordinates, and subsequently assessed structural confidence via the predicted Local Distance Difference Test (pLDDT) score. The pLDDT provides a per-residue estimate of the expected local accuracy, with values closer to 100 indicating greater confidence that the predicted residue positions closely match their true, native configuration. Mean pLDDT scores, computed by averaging across all residues of a sequence, thus served as a global indicator of the sequence’s foldability and structural regularity.

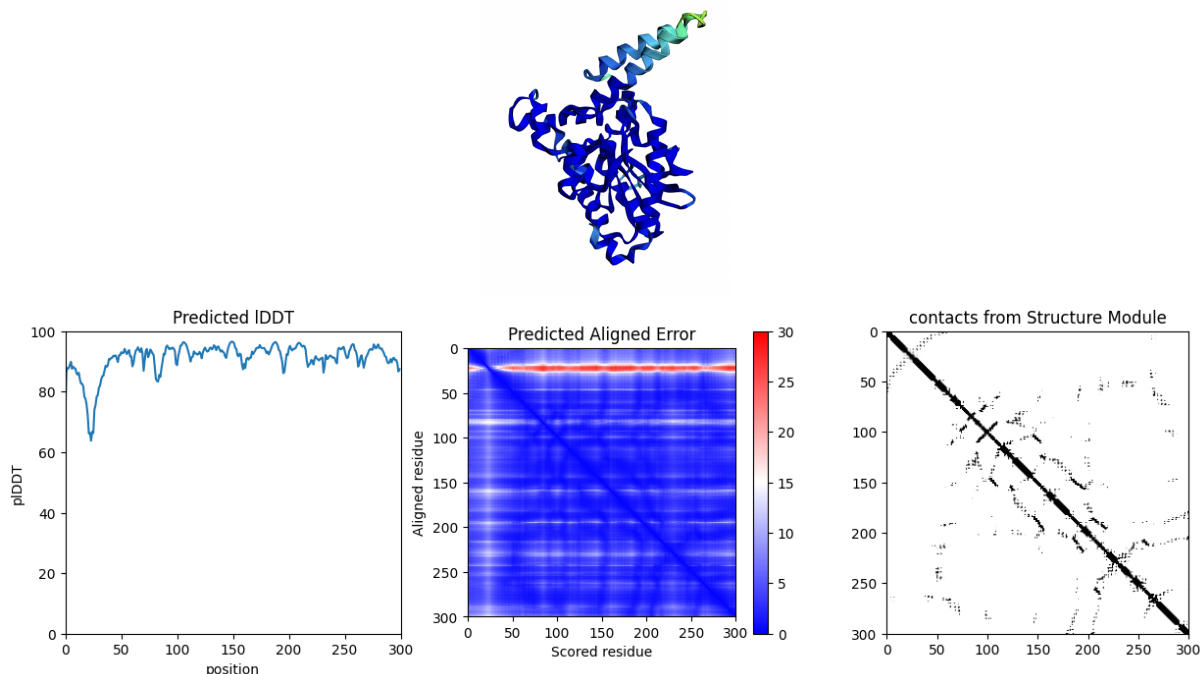


Figure 2.5: Example of ESMFold evaluation on sequence chunk taken from Uniref100. Sequence shown is a component of the acetyl coenzyme A carboxylase (ACC) complex [1].

High mean pLDDT scores suggest that the generated sequence adopts a well-ordered, plausible conformation rather than collapsing into disordered or unrealistic structures. Given the strong dependence of protein function on precise tertiary and quaternary arrangements, structural plausibility was treated as a necessary, though not sufficient, criterion for biological relevance. An illustrative example of an ESMFold-predicted structure for a generated sequence is shown in Figure 2.5, demonstrating the model’s ability to predict topologies without explicit structure supervision during generation.

2.5.2 Sequence Entropy

In parallel with structural evaluation, we assessed the compositional diversity of the generated sequences through entropy analysis, building on amino acid distribution analysis techniques similar to those used in previous protein design studies [33, 34]. For each

sequence, we first computed the empirical distribution $\{p_i\}$ over the 20 canonical amino acids along with an additional category representing unknown or masked residues, resulting in a total of 21 possible residue types. Based on this empirical distribution, the Shannon entropy S was calculated according to

$$S = - \sum_{i=1}^{21} p_i \log p_i,$$

where p_i denotes the frequency of residue type i within the sequence.

Rather than reporting the raw entropy directly, we exponentiated the negative entropy to yield an effective diversity score.

$$\text{Effective Diversity} = e^{-S}.$$

This provides a more intuitive interpretation: it estimates the effective number of distinct amino acid categories contributing meaningfully to the sequence. In this formulation, higher effective diversity values correspond to broader sampling across amino acid types, while lower values indicate skewed or degenerate distributions dominated by a few residues.

This measure is critical for evaluating the generative model’s ability to avoid mode collapse, a common failure mode in generative tasks where outputs become repetitive or overly simplistic. A sequence exhibiting low effective diversity would suggest that the model has converged toward trivial motifs (e.g., homopolymeric stretches) rather than exploring the vast combinatorial space of plausible protein sequences. Conversely, higher effective diversity suggests a richer exploration of sequence space, increasing the likelihood of sampling sequences with novel structures and functions.

Importantly, ensuring compositional diversity is linked to biological plausibility. Proteins that perform complex biological functions often exhibit heterogeneous amino acid compositions tailored to the physicochemical demands of their structures and interactions.

Thus, maintaining high sequence diversity is necessary for designing proteins capable of achieving non-trivial biological activities.

2.5.3 DeepFRI

Finally, functional evaluation was performed by applying a pre-trained DeepFRI model to the generated sequences directly [17]. DeepFRI predicts functional annotations from raw amino acid sequences without requiring structural input, leveraging a learned mapping from sequence patterns to Gene Ontology terms. Since sequence generation was conditioned on specific GO term targets, we evaluated recall by comparing the set of GO terms predicted by DeepFRI against the set of target labels used for conditioning. Recall in this setting measures the fraction of conditioned proteins whose functions were successfully retrieved from the model’s predictions, offering a direct assessment of functional alignment. High recall scores indicated that the generative process succeeded not only in producing plausible and diverse sequences, but also in embedding sequence features necessary for the desired molecular functions, biological processes, or cellular components.

DeepFRI is a graph convolutional network-based method originally developed for functional annotation of proteins, capable of predicting Gene Ontology (GO) terms from protein inputs. While DeepFRI was initially designed to incorporate both sequence and structure information through graph-based representations, it also supports predictions from raw amino acid sequences alone, learning patterns that correlate sequence motifs with functional outcomes.

For each generated sequence, DeepFRI outputs a probabilistic prediction across a predefined set of GO terms encompassing molecular functions, biological processes, and cellular components. Because sequence generation in this work was conditioned on specific GO term targets, functional evaluation focused on backwards prediction: that is, the prediction performance of DeepFRI to recover the function annotations that the sequence was conditioned on. Prediction performance metrics include recall (the fraction of target

functions successfully recovered), precision (the fraction of predicted functions that match the target), F1 score (the harmonic mean of recall and precision), coverage (the proportion of sequences for which at least one GO term was predicted), and confidence (the average probability score assigned to correctly predicted terms). Together, these metrics provide a comprehensive assessment of how well the generated protein sequences express the functional properties they were designed to have, with higher values indicating more successful function-guided sequence generation.

Functional evaluation using DeepFRI is especially valuable because it bridges the gap between structural plausibility and biological relevance. A sequence may exhibit high structural confidence (e.g., high pLDDT scores) and high compositional diversity, yet still fail to encode meaningful function if the conditioning signal is not faithfully incorporated. By using a function predictor, we are able to assess not only whether the sequences are plausible proteins, but also whether they are likely to perform the intended molecular tasks. Thus, DeepFRI prediction performance serves as an essential complement to structural and diversity metrics, enabling a more comprehensive evaluation of generative success.

Chapter 3

Discussion

3.1 Results

We trained models with hidden dimensions 512 and 1024 for both ESM3 and GO-based function conditioning. After training for 30000 steps, we evaluate 5 batches of 100 sequences, 500 total, on the its structural integrity, sequence diversity, and functional adherence, and calculate average values in each metric in these areas. Further training sees the model often converging towards trivial motifs, thus we implement this early stop to prevent over-homologous sequences.

3.1.1 Structural Integrity

The structural integrity of generated protein sequences was assessed using ESMFold’s predicted Local Distance Difference Test (pLDDT) scores, with results presented in Figure 3.1. pLDDT scores serve as a proxy for the confidence in the predicted structure, with higher values indicating greater structural plausibility. As shown, all model variants produced sequences with substantially lower pLDDT scores (ranging from 32.5 to 36.8) compared to the baseline of a random sample of 1000 naturally occurring proteins in Uniref100 (81.1), suggesting limited structural stability in the generated proteins.

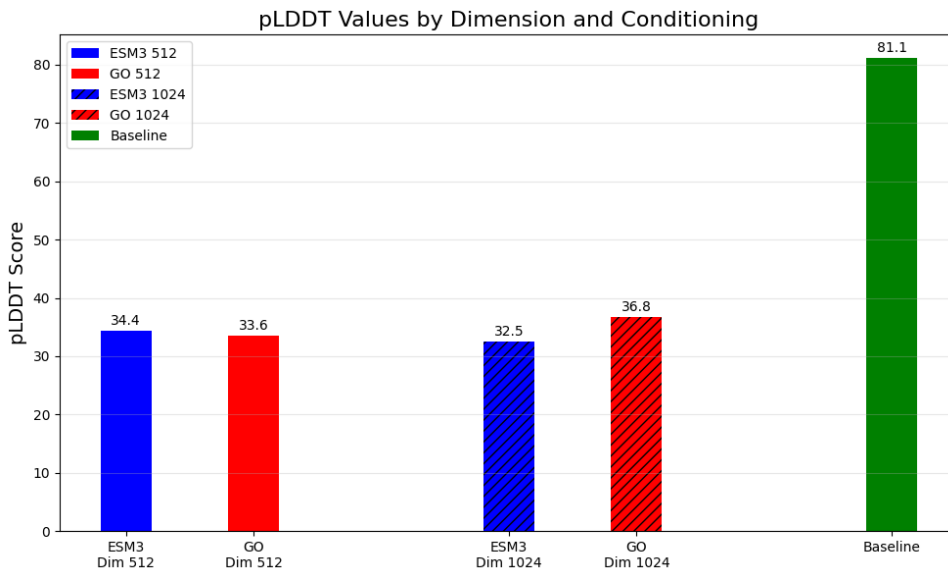


Figure 3.1: pLDDT evaluation against baseline. Baseline metrics calculated from a sample of naturally occurring proteins in the UniRef100 dataset

Table 3.1: pLDDT Values

Model	Hidden Dim	pLDDT
ESM3	512	34.4
GO	512	33.6
ESM3	1024	32.5
GO	1024	36.8
Baseline	-	81.1

Table 3.1 provides the numerical pLDDT values across model configurations. Interestingly, increasing the hidden dimension from 512 to 1024 had no significant effects, and the GO model with 1024 hidden dimensions achieved the highest pLDDT score (36.8) among all tested configurations, though still far below baseline.

3.1.2 Sequence Diversity

Sequence diversity, measured by entropy across the generated amino acid distributions, is visualized in Figure 3.2. Higher entropy values indicate greater sequence diversity, which is desirable for exploring the protein sequence space. The generated sequences maintained competitive entropy levels compared to naturally occurring proteins, which we again used as the baseline.

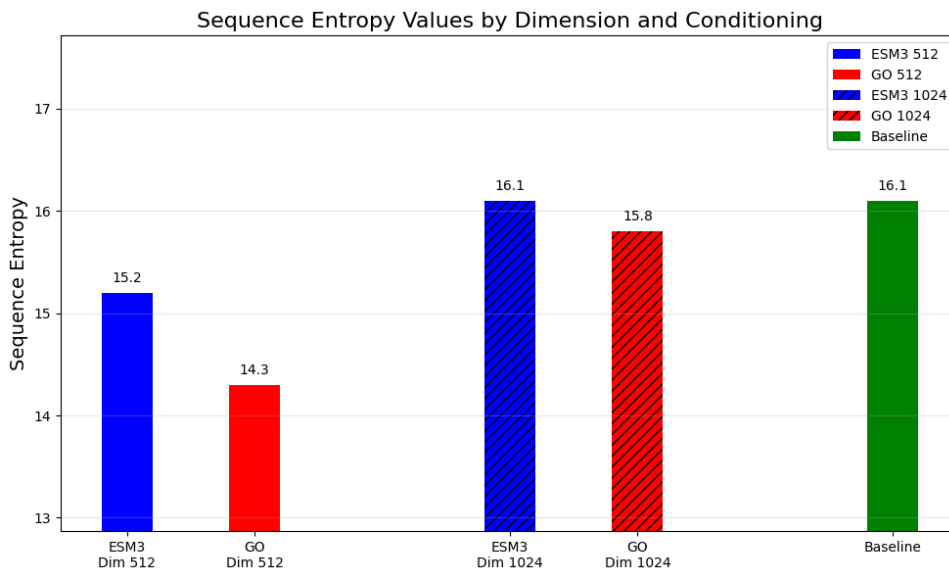


Figure 3.2: Sequence Entropy evaluation results against baseline. Baseline metrics calculated from a sample of naturally occurring proteins in the UniRef100 dataset

Table 3.2: Sequence Entropy Values

Model	Hidden Dim	Sequence Entropy
ESM3	512	15.2
GO	512	14.3
ESM3	1024	16.1
GO	1024	15.8
Baseline	-	16.1

Table 3.2 presents the sequence entropy values across model configurations. The ESM3 model with 1024 hidden dimensions achieved entropy matching the baseline (16.1), while other configurations showed slightly lower but comparable diversity. This suggests that our models successfully capture the natural diversity of the protein sequence space with conditioning on functional information.

3.1.3 Functional Adherence



Figure 3.3: DeepFRI prediction results as a percentage of baseline. Baseline metrics calculated from a sample of naturally occurring proteins with similar annotations in the UniRef100 dataset

Finally, functional adherence was evaluated using DeepFRI to predict GO terms from the generated sequences, with results shown as percentages of baseline performance in Figure 3.3. This assessment measures how well the generated proteins match their intended functions compared to naturally occurring proteins with similar annotations.

Table 3.3: DeepFRI Metrics as Percentages of Baseline

Model	Hidden Dim	Recall	Precision	F1 Score	Coverage	Confidence
ESM3	512	27.1%	33.3%	29.6%	23.8%	56.6%
GO	512	42.7%	58.3%	48.1%	51.6%	48.1%
ESM3	1024	21.9%	50.0%	27.8%	22.9%	58.6%
GO	1024	47.4%	63.9%	53.2%	55.1%	52.7%

Table 3.3 details the functional prediction metrics across model configurations. GO-conditioned models consistently outperformed ESM3-conditioned counterparts across all metrics, with the GO model at 1024 hidden dimensions achieving the best results (47.4% recall, 63.9% precision, and 53.2% F1 score). The relatively high precision compared to recall suggests that when DeepFRI made predictions, they often aligned with intended functions, but many functional aspects were missed altogether.

3.2 Key Findings

Our experimental results reveal several important insights into protein generation with functional conditioning. Entropy metrics remained strong across all model configurations, approaching or matching baseline levels from naturally occurring proteins. This indicates that our models successfully preserve sequence diversity while attempting to satisfy functional constraints. However, we observed that sequence entropy deteriorated with lower temperature settings and overtraining, suggesting a trade-off between diversity and other optimization objectives.

Despite the promising sequence diversity, pLDDT scores remained consistently low across all models (32.5-36.8) compared to naturally occurring proteins (81.1), indicating significant structural instability in the generated sequences. Increasing the hidden dimension from 512 to 1024 yielded minimal improvements, with the GO-conditioned

1024-dimensional model showing the highest score (36.8). This suggests that neither the conditioning method nor model capacity directly addresses the core challenge of generating structurally plausible proteins without explicit structural supervision.

In terms of functional adherence, DeepFRI predictions revealed a clear advantage for GO-conditioned models over ESM3-conditioned variants, with the GO model at 1024 hidden dimensions achieving the best performance (47.4% recall, 63.9% precision). However, overall functional adherence remained limited due to several factors: (1) added noise from the DeepFRI prediction pipeline, (2) the inherent complexity of predicting function without explicit structural information, and (3) low confidence predictions resulting in many proteins receiving no functional labels at all. While precision was relatively high (indicating accuracy when predictions were made), the limited coverage and recall significantly impacted overall F1 scores.

The coverage metric (percentage of proteins receiving any functional predictions) deserves special attention in this evaluation context. Unlike traditional prediction tasks, GO term prediction is complicated by the hierarchical, tree-like structure of the GO ontology, where terms often overlap semantically. Low coverage (22.9-55.1%) indicates that many generated proteins fail to trigger confident predictions for any GO terms, suggesting fundamental issues with functional expression rather than merely imprecise predictions. The notably higher coverage in GO-conditioned models (51.6-55.1%) versus ESM3-conditioned models (22.9-23.8%) further supports the effectiveness of direct functional conditioning.

The superior performance of GO conditioning over ESM3 conditioning (approximately 20 percentage point improvement in recall and 14 percentage point improvement in precision at the 1024 hidden dimension) demonstrates that direct incorporation of functional information through GO terms better guides protein generation than indirect language-based embeddings. This supports our hypothesis that explicit functional conditioning provides more direct supervision for the generative process, even if absolute performance remains below desired levels.

These findings highlight both progress in functionally conditioned protein generation and persistent challenges that require further methodological advances. The gap between generated and naturally occurring proteins remains substantial, particularly in structural integrity, suggesting that sequence-only approaches may be fundamentally limited without incorporating explicit structural constraints or inductive biases toward foldability.

3.3 Limitations

This study highlights several critical limitations in modeling protein sequence generation from function, particularly when structure is absent from the learning and supervision process.

A central limitation lies in the exclusive use of sequence and functional annotations, without incorporating structural information. While amino acid sequence does encode some functional signals, function is most directly emergent from three-dimensional conformation and dynamic interactions. Proteins with near-identical sequences may fold differently and exhibit divergent functional properties, while proteins with low sequence similarity can converge structurally and share common functions. This disconnect is reflected in the low structural fidelity of generated sequences, as measured by pLDDT scores, and the limited recall and confidence observed in downstream function prediction using DeepFRI. The reliance on sequence alone constrains the model’s ability to capture nuanced biophysical patterns necessary for true functional mimicry.

Additionally, while the model does incorporate some information about GO term relationships by expanding the conditioning label set to include each term’s ancestors, this approach treats all ancestors equally and does not explicitly encode distances or inheritance strengths within the GO hierarchy. As a result, important semantic and functional gradations between GO terms are flattened. For instance, closely related sibling terms or deep ancestral terms contribute identically, even if their relevance to the conditioned

function differs significantly. This lack of fine-grained hierarchical encoding may weaken the model’s ability to generalize across related functions or to enforce consistency in multi-label settings.

Evaluation presents its own challenges. The function prediction model used—DeepFRI—is trained on AlphaFold-predicted structures of natural sequences and may not generalize well to synthetic or low-confidence structures. Its poor performance on even natural proteins in the same dataset indicates potential limitations in its predictive granularity and bias towards structural features that the generator does not yet recover. Consequently, the limited function recall observed may reflect both failures of generation and blind spots in evaluation.

Finally, there are broader limitations to the biological assumptions embedded in the model. Functional expression in proteins is highly context-dependent, influenced by localization, interactions, post-translational modifications, and environmental conditions—all of which are omitted in the current modeling scope. These simplifications restrict both the interpretability of results and the applicability of generated sequences in practical or therapeutic settings.

Chapter 4

Future Work

There are several directions in which this work could be meaningfully extended to better capture and control protein functionality.

4.1 Structure Track Implementation

Function is fundamentally constrained and expressed through a protein’s three-dimensional structure. While this project uses functional labels and sequence-level information alone, a promising avenue is the introduction of an explicit structural track, enabling the model to attend over and condition on protein structure during generation.

One viable approach involves aligning sequence data from UniRef with experimentally resolved or predicted structures from the Protein Data Bank (PDB) [35]. By constructing a multimodal training pipeline, embeddings of atomic or residue-level 3D coordinates (or coarser representations like distance maps or torsion angles) could be encoded alongside sequence and function [36, 37]. This would allow the model to internalize structural constraints and improve the plausibility of generated sequences—potentially improving metrics such as pLDDT and functional compatibility. An effort of this is already being made in parallel to this work at the CSAIL Berger Lab.

4.2 Feedback-Driven Protein Generation

Another major direction is incorporating feedback mechanisms into the generation process. Rather than relying on a single forward pass from conditioning to sequence, models could iteratively resample, correct, and refine outputs based on predicted functional and structural evaluations. For example, after an initial sequence is generated, it could be assessed using pretrained structure or function predictors, and then reconditioned with these evaluations to guide resampling [38, 39].

This “chain-of-thought” generation could enable more controllable design: generation that is aware of and responsive to its downstream impact. Such frameworks also open the door to reinforcement learning or other optimization techniques where the reward is defined by downstream functional utility.

4.3 Encoding Functional Semantics

While GO term labels were expanded to include their ancestral nodes in the ontology, the present work does not explicitly encode distances or semantic relationships between GO terms. Future models might benefit from embedding the GO graph itself using ontological graph embedding techniques (e.g., Node2Vec, GraphSAGE, or OPA2Vec) [40, 41, 42] to capture richer relational information.

Function could also be encoded more precisely by incorporating domain-specific representations, such as EC numbers or Pfam annotations. These could either be used as auxiliary inputs or as part of a hierarchical conditioning mechanism, allowing the model to tune generation more finely to the biological task.

4.4 Contextualizing Functional Origin

Another limitation is the lack of consideration for organismal context. In natural systems, protein function is not only dependent on structure, but also on the cellular and environmental context in which it operates. Future work could explore adding contextual embeddings indicating species, tissue, or even subcellular localization—if available—to make conditioning more biologically meaningful.

Incorporating evolutionary context such as orthology groups or phylogenetic trees might also help the model distinguish between convergent and divergent functional strategies across species.

4.5 Incorporating Active Site Embeddings

Much of a protein’s function is governed by specific, localized regions—active sites—that carry out catalysis or binding. Tools such as InterProScan provide annotations of these functional subregions. Future models could extract these annotated active sites and embed them separately, allowing models to better reason about localized functional requirements during generation [43].

This could take the form of dual-stream architectures, where one encoder processes global sequence embeddings while another processes sparse active site representations. The fusion of these streams could lead to more accurate control over highly specific molecular functions.

4.6 Toward Experimentally Viable Proteins

While this work assesses generation using metrics like DeepFRI recall and pLDDT confidence, these are proxies. In the long term, a crucial direction is the establishment of a computational evaluation pipeline robust enough to predict real-world viability. Such

a pipeline would assess structure, function, immunogenicity, and stability, and return a high-level score that can act as a selection criterion for wet-lab testing.

Eventually, a generated protein that satisfies these thresholds could be flagged for downstream experimental synthesis and testing. In this vision, generation is not a one-off process but part of an iterative design loop. Computational methods will form a triage layer before moving to the high-cost, high-impact world of experimental biology. Achieving this vision will require further validation of evaluation metrics against experimental outcomes and tighter collaboration between computational and wet-lab communities.

This page is intentionally left blank

Appendix A

Code Availability & Compute Resources

The code for reproducing these results are available at:

<https://github.com/adrinatang77/MEng-Function-Conditioning>

Interproscan is available for download at:

<https://www.ebi.ac.uk/interpro/download/InterProScan/>

The Gene Ontology database is available for download at:

<https://geneontology.org/docs/download-ontology/>

The UniRef database is available for download at:

<https://www.uniprot.org/uniref/>

All models were trained on A6000 8x GPU Machines at the CSAIL Berger Lab.

Appendix B

Model Hyperparameters

Table B.1: Final hyperparameters for training and generation. These hyperparameters are shared between the different conditioning methods and model dimensions examined.

Parameter	Value
Model Architecture	
Number of Heads	16
Number of Blocks	30
Dropout	0.2
Activation Function	GELU
Optimizer	
Type	AdamW
Learning Rate	1e-4
Warmup Steps	1,000
Start Decay	10,000
Sequence Track	
Loss Weight	3.0
Reweight Epsilon	0.01
Function Track	
Term Drop Probability	0.15
Annotation Drop Probability	0.15
Sampling Parameters	
Temperature Start	1.0
Temperature End	0.1
Strategy	dplm
Logits	gumbel
Sample Length	300
Steps	300

Appendix C

Additional Results

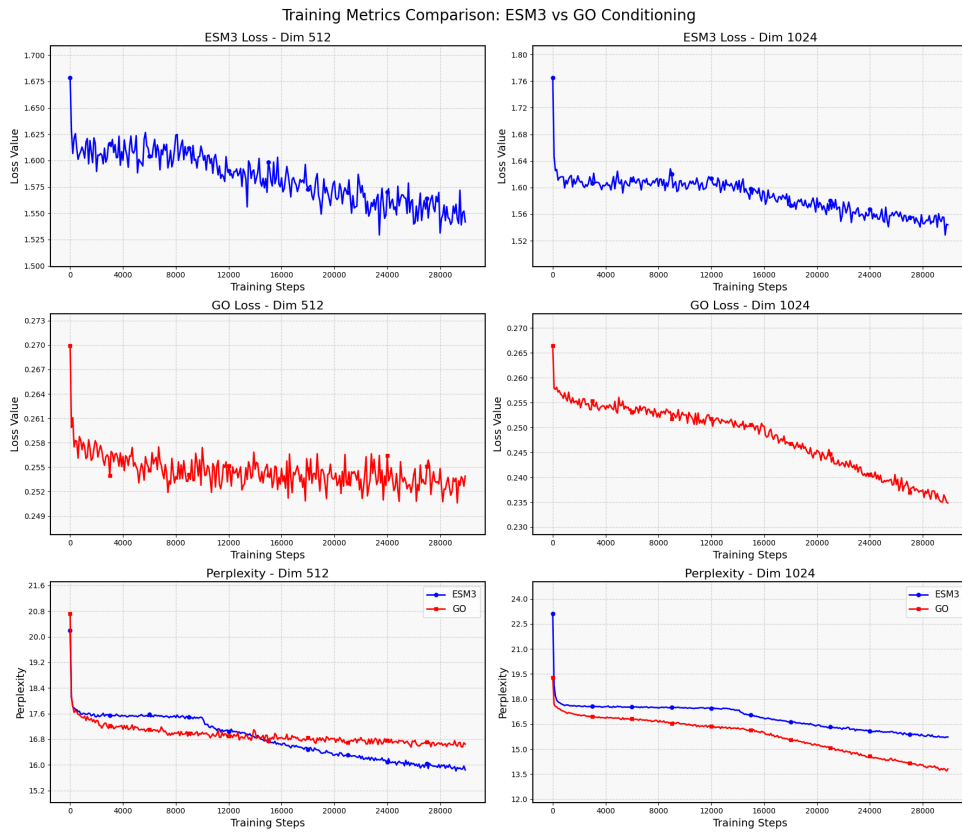


Figure C.1: Loss and perplexity of models during training

Table C.1: Raw Comprehensive Performance Metrics

Metric	ESM3 (512)	GO (512)	ESM3 (1024)	GO (1024)	Baseline
Seq. Entropy	15.2	14.3	16.1	15.8	16.1
pLDDT	34.4	33.6	32.5	36.8	81.1
Recall	0.052	0.082	0.042	0.091	0.192
Precision	0.12	0.21	0.18	0.23	0.36
F1	0.073	0.118	0.068	0.130	0.245
Coverage	0.082	0.178	0.079	0.190	0.345
Confidence	0.392	0.333	0.406	0.365	0.693

This page is intentionally left blank

Bibliography

- [1] UniProt Consortium, Alexandra Bateman, et al. UniProt: the Universal Protein Knowledgebase in 2025. *Nucleic Acids Research*, 53(D1):D609–D617, 2025. doi: 10.1093/nar/gkae1010.
- [2] P.S. Huang, S.E. Boyken, and D. Baker. The coming of age of de novo protein design. *Nature*, 537:320–327, 2016. doi: 10.1038/nature19946.
- [3] Andreas Bjerregaard, Peter Mørch Groth, Søren Hauberg, Anders Krogh, and Wouter Boomsma. Foundation models of protein sequences: A brief overview. *Current Opinion in Systems Biology*, 37:100453, 2024. doi: 10.1016/j.coisb.2024.100453.
- [4] Y. Liu, S. Wang, J. Dong, et al. De novo protein design with a denoising diffusion network independent of pretrained structure prediction models. *Nat Methods*, 21: 2107–2116, 2024. doi: 10.1038/s41592-024-02437-w.
- [5] J. Jumper, R. Evans, A. Pritzel, et al. Highly accurate protein structure prediction with alphafold. *Nature*, 596:583–589, 2021. doi: 10.1038/s41586-021-03819-2.
- [6] Z. Lin, H. Akin, R. Rao, B. Hie, Z. Zhu, W. Lu, N. Smetanin, R. Verkuil, O. Kabeli, Y. Shmueli, A. Dos Santos Costa, M. Fazel-Zarandi, T. Sercu, S. Candido, and A. Rives. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023. doi: 10.1126/science.ade2574.
- [7] Thomas Hayes, Roshan Rao, Halil Akin, Nicholas J. Sofroniew, Deniz Oktay, Zeming Lin, Robert Verkuil, Vincent Q. Tran, Jonathan Deaton, Marius Wiggert, Rohil Badkundri, Irhum Shafkat, Jun Gong, Alexander Derry, Raul S. Molina, Neil Thomas, Yousuf Khan, Chetan Mishra, Carolyn Kim, Liam J. Bartie, Matthew Nemeth, Patrick D. Hsu, Tom Sercu, Salvatore Candido, and Alexander Rives. Simulating 500 million years of evolution with a language model. *bioRxiv*, page 2024.07.01.600583, 2024. doi: 10.1101/2024.07.01.600583.
- [8] E. Nijkamp, J. Ruffolo, E.N. Weinstein, et al. Progen: Language modeling for protein generation. *arXiv*, 2020. doi: 10.48550/arXiv.2004.03497.
- [9] Jarrid Rector-Brooks et al. Steering masked discrete diffusion models via discrete denoising posterior prediction. *arXiv preprint*, 2024. doi: 10.48550/arXiv.2410.08134.

- [10] Andrew Campbell, Jason Yim, Regina Barzilay, Tom Rainforth, and Tommi Jaakkola. Generative flows on discrete state-spaces: Enabling multimodal flows with applications to protein co-design. *arXiv preprint*, 2024. doi: 10.48550/arXiv.2402.04997.
- [11] M. Van Kempen, S.S. Kim, C. Tumescheit, et al. Fast and accurate protein structure search with foldseek. *Nat Biotechnol*, 41:357–360, 2023. doi: 10.1038/s41587-023-01773-0.
- [12] Michael Heinzinger, Riccardo Colangelo, Abdellah I. Ben Abded, Wolfgang Ritter, Ann K. Green, Johannes Hingerl, and Julien Gagneur. Bilingual language model for protein sequence and structure (ProstT5). *NAR Genomics and Bioinformatics*, 6(4): lqae150, 2024. doi: 10.1093/nargab/lqae150.
- [13] Matthias Blum et al. Interpro: the protein sequence classification resource in 2025. *Nucleic Acids Research*, 2024. doi: 10.1093/nar/gkae1082.
- [14] J.L. Watson, D. Juergens, N.R. Bennett, et al. De novo design of protein structure and function with rdiffusion. *Nature*, 620:1089–1100, 2023. doi: 10.1038/s41586-023-06415-8.
- [15] Minkyung Baek et al. Accurate prediction of protein structures and interactions using a three-track neural network. *Science*, 373:871–876, 2021. doi: 10.1126/science.abj8754.
- [16] Xinyou Wang, Zaixiang Zheng, Fei Ye, Dongyu Xue, Shujian Huang, and Quanquan Gu. Dplm-2: A multimodal diffusion protein language model. *arXiv preprint*, 2024. doi: 10.48550/arXiv.2410.13782.
- [17] V. Gligorijevic, P.D. Renfrew, T. Kosciolk, et al. Structure-based protein function prediction using graph convolutional networks. *Nat Commun*, 12:3168, 2021. doi: 10.1038/s41467-021-23303-9.
- [18] R. You, Z. Zhang, Y. Xiong, et al. Golabeler: improving sequence-based large-scale protein function prediction by learning to rank. *Bioinformatics*, 34:2465–2473, 2018. doi: 10.1093/bioinformatics/bty130.
- [19] M. Kulmanov and R. Hoehndorf. Deepgoplus: improved protein function prediction from sequence. *Bioinformatics*, 36:422–429, 2020. doi: 10.1093/bioinformatics/btz595.
- [20] Predrag Radivojac, Wyatt T Clark, Tal Ronnen Oron, Alexandra M Schnoes, Tobias Wittkop, Artem Sokolov, Kiley Graim, Christopher Funk, Karin Verspoor, Asa Ben-Hur, et al. A large-scale evaluation of computational protein function prediction. *Nature Methods*, 10(3):221–227, 2013.
- [21] Neng Zhou, Yuxiang Jiang, Tyler R Bergquist, Andrew J Lee, Balint Z Kacsóh, Amanda W Crocker, Kimberley A Lewis, Georgia Georghiou, Ha Nguyen, Md Nay-eem Hamid, et al. The cafa challenge reports improved protein function prediction

- and new functional annotations for hundreds of genes through experimental screens. *Genome Biology*, 20(1):244, 2019.
- [22] Yuxiang Jiang, Tal Ronnen Oron, Wyatt T Clark, Asma R Bankapur, David D’Andrea, Rosalba Lepore, Christopher S Funk, Indika Kahanda, Karin M Verpoor, Asa Ben-Hur, et al. An expanded evaluation of protein function prediction methods shows an improvement in accuracy. *Genome Biology*, 17(1):184, 2016.
- [23] M. Ashburner, C.A. Ball, J.A. Blake, et al. Gene ontology: tool for the unification of biology. *Nat Genet*, 25:25–29, 2000. doi: 10.1038/75556.
- [24] J. Mistry, S. Chuguransky, L. Williams, et al. Pfam: The protein families database in 2021. *Nucleic Acids Res.*, 49:D412–D419, 2021. doi: 10.1093/nar/gkaa913.
- [25] C.J.A. Sigrist, E. de Castro, L. Cerutti, et al. New and continuing developments at prosite. *Nucleic Acids Res.*, 41:D344–D347, 2013. doi: 10.1093/nar/gks1067.
- [26] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, 2017. doi: 10.5555/3295222.3295349.
- [27] Bowen Jing, Sebastian Eismann, Pranam Suriana, Raphael J Townshend, and Ron O Dror. Learning from protein structure with geometric vector perceptrons. In *International Conference on Learning Representations (ICLR)*, 2021.
- [28] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. doi: 10.18653/v1/N19-1423.
- [29] Karen Spärck Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28(1):11–21, 1972.
- [30] Aristides Gionis, Piotr Indyk, and Rajeev Motwani. Similarity search in high dimensions via hashing. In *Proceedings of the 25th International Conference on Very Large Data Bases (VLDB)*, pages 518–529. Morgan Kaufmann Publishers Inc., 1999.
- [31] Wouter Kool, Herke van Hoof, and Max Welling. Ancestral gumbel-top-k sampling for sampling without replacement. *Journal of Machine Learning Research*, 21(47): 1–36, 2020. URL <http://jmlr.org/papers/v21/19-985.html>.
- [32] Tianshu Wang, Xiaocheng Jin, Xiaoli Lu, Xiaoping Min, Shengxiang Ge, and Shaowei Li. Empirical validation of proteinmpnn’s efficiency in enhancing protein fitness. *Frontiers in Genetics*, 14:1347667, 2024. doi: 10.3389/fgene.2023.1347667.
- [33] E. Rivas. Evolutionary models for insertions and deletions in a probabilistic modeling framework. *BMC Bioinformatics*, 6:63, 2005. doi: 10.1186/1471-2105-6-63.

- [34] N. Ferruz, S. Schmidt, and B. Höcker. Protgpt2 is a deep unsupervised language model for protein design. *Nat Commun*, 13:4348, 2022. doi: 10.1038/s41467-022-32007-7.
- [35] Helen M. Berman, Kim Henrick, and Haruki Nakamura. Announcing the worldwide Protein Data Bank. *Nucleic Acids Research*, 28(1):235–242, 2000. doi: 10.1093/nar/28.1.235.
- [36] Kevin E. Wu, Kevin K. Yang, Rianne van den Berg, Sarah Alamdari, James Y. Zou, Alex X. Lu, and Ava P. Amini. Protein structure generation via folding diffusion. *Nature Communications*, 15(1):1059, 2024. doi: 10.1038/s41467-024-44897-x.
- [37] Duolin Wang, Mahdi Pourmirzaei, Usman L. Abbas, Shuai Zeng, Nima Manshour, Farzan Esmaili, Bijan Poudel, Yike Jiang, Qingnan Shao, Jie Chen, and Dayu Xu. S-PLM: Structure-Aware Protein Language Model via Contrastive Learning Between Sequence and Structure. *Advanced Science (Weinheim)*, 12(5):e2404212, 2025. doi: 10.1002/advs.202404212.
- [38] Yi Wang, Hui Tang, Lichao Huang, Lulu Pan, Lixiang Yang, Huanming Yang, Feng Mu, and Meng Yang. Self-play reinforcement learning guides protein engineering. *Nature Machine Intelligence*, 5:845–860, 2023. doi: 10.1038/s42256-023-00733-0.
- [39] Brian L. Hie and Kevin K. Yang. Adaptive machine learning for protein engineering. *Current Opinion in Structural Biology*, 72:145–152, 2022. ISSN 0959-440X. doi: 10.1016/j.sbi.2021.11.002. URL <https://doi.org/10.1016/j.sbi.2021.11.002>.
- [40] A. Grover and J. Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 855–864, 2016. doi: 10.1145/2939672.2939754.
- [41] W.L. Hamilton, R. Ying, and J. Leskovec. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems*, pages 1024–1034, 2017. doi: 10.48550/arXiv.1706.02216.
- [42] F.Z. Smaili, X. Gao, and R. Hoehndorf. Opa2vec: combining formal and informal content of biomedical ontologies to improve similarity-based prediction. *Bioinformatics*, 34(21):3504–3510, 2018. doi: 10.1093/bioinformatics/bty933.
- [43] Xiaorui Wang, Xiaodan Yin, Dejun Jiang, Huifeng Zhao, Zhenxing Wu, Odin Zhang, Jike Wang, Yuquan Li, Yafeng Deng, Huanxiang Liu, Pei Luo, Yuqiang Han, Tingjun Hou, Xiaojun Yao, and Chang-Yu Hsieh. Multi-modal deep learning enables efficient and accurate annotation of enzymatic active sites. *Nature Communications*, 15(1):1929, 2024. doi: 10.1038/s41467-024-51511-6.