# PROGEA: UNIFYING PROTEIN SEQUENCE, STRUCTURE, AND LANGUAGE WITH DIFFUSION LANGUAGE MODELS

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

We envision a general-purpose protein foundation model as a unified, interactive model of sequence, structure, and function, capable of understanding, generation, and cross-modal translation while retaining the instruction-following and reasoning capabilities of modern LLMs. Toward this vision, we introduce **PROGEA**, the first native, unified, multimodal diffusion language model (dLLM) for proteins. PROGEA is built on two key design choices: (1) a unified discrete token space that jointly represents sequence, structure, and natural language; (2) a dLLM backbone that replaces strict left-to-right factorization with bidirectional or semi-bidirectional token interactions, enabling global residue communication and long-range contact modeling that are naturally suited to structure modeling. This formulation unifies six core capabilities: folding, inverse folding, sequence understanding, structure understanding, sequence design, and structure design, while retaining the native language capabilities of general-purpose LLMs. On established benchmarks, PROGEA achieves broadly competitive performance. In folding, Progea-MDLM improves backbone TM-score over existing tri-modal model by 0.30/0.16 on CAMEO22/PDB split. For sequence understanding, Progea-BDLM closely matches the best evaluated frontier LLM, Claude Opus 4.7 (62.37%). This result provides strong evidence for effective sequence–structure alignment, extending the sequence–language representation space established in prior work into 3D physical space.

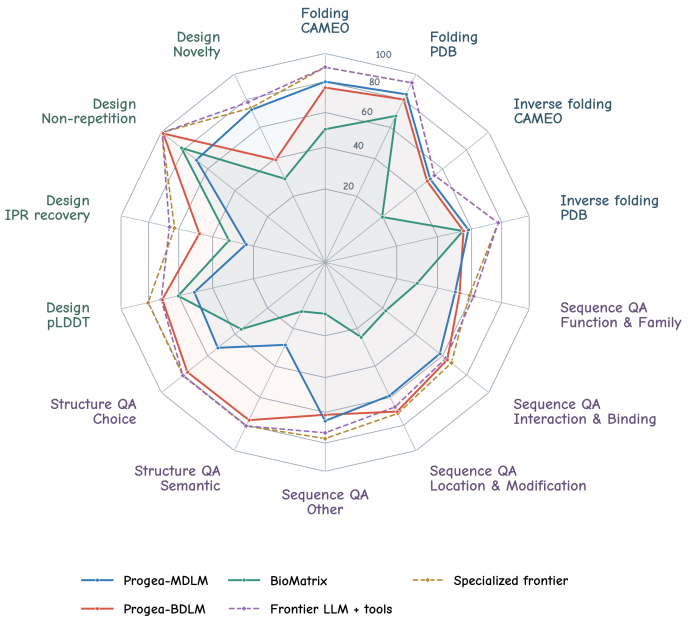



Figure 1: Performance Rader.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

# 1 INTRODUCTION

Proteins constitute the molecular machinery of life, carrying out the catalytic, structural, signaling, and regulatory processes that sustain cells and organisms. A comprehensive understanding of proteins draws on three complementary modalities: (1) *amino acid sequences* describe molecular identity and evolutionary constraints (Rives et al., 2021). (2) *3D structures* represent the physical and geometric constraints that govern molecular interactions, thereby revealing the mechanistic basis of protein function (Dill & MacCallum, 2012). (3) *function descriptions* capture not only specific protein functions but also the broader body of human knowledge accumulated through biological research (gen, 2021; uni, 2023).

Importantly, these modalities are not independent views of proteins. We distill protein modeling into six meta-capabilities spanning the directed transformations among sequence, structure, and function: **folding**, **inverse folding**, **sequence understanding**, **structure understanding**, **sequence design**, and **structure design**. Rather than isolated tasks, these capabilities serve as composable primitives whose combinations give rise to diverse and increasingly complex protein research workflows.

Substantial progress has been made through specialized protein models; in Sec. 2 and Appendix. A, we review these developments and revisit several long-overlooked connections to modern language modeling. Nevertheless, protein modeling remains fragmented across specialized systems and largely disconnected from general-purpose LLMs. We identify two key requirements for next-generation general-purpose protein foundation models:

- **Unified protein framework.** A single model should enable knowledge sharing and mutual reinforcement across tasks, without relying on cross-model interfaces or task-specific adaptation.
- **Native language capabilities.** The model should retain the instruction-following, reasoning, and interactive capabilities native to general-purpose LLMs.

Early attempts explored unifying heterogeneous protein modalities within autoregressive frameworks (Ma et al., 2025; Chen et al., 2024), roughly in parallel with the emergence of unified vision models (Team, 2025; Wang et al., 2024a). Yet these approaches remained limited on structure-intensive tasks such as protein folding, leaving open *whether the autoregressive paradigm itself provides the right foundation for unified protein modeling*.

We introduce **PROGEA**, to our knowledge the first native, unified, tri-modal diffusion language model (dLLM) for proteins. PROGEA builds on two key design choices: (1) a shared discrete token space that jointly represents protein sequence, structure, and natural language; and (2) a non-causal diffusion backbone that relaxes strict left-to-right factorization, enabling global residue interactions modeling within an LLM architecture. Together, these designs unify the six directed capabilities across the sequence–structure–function space within a single generative model. Across established benchmarks, PROGEA achieves broadly competitive performance, with particularly substantial improvements over existing unified tri-modal models on folding.

We highlight our contributions as follows:

- **Framework.** We develop PROGEA, a native unified protein dLLM that brings sequence, structure, and natural language into a shared discrete token space and models them with a non-causal diffusion backbone. Guided by the intrinsic compatibility between specialized protein modeling and modern language-model primitives, PROGEA unifies six core capabilities within a single generative framework while retaining native language interaction, and further supports their interleaved composition.
- **Capability.** At the 4B scale, PROGEA achieves competitive performance across all six core capabilities and substantially strengthens unified tri-modal folding, reaching $\mathrm{TM}_{\mathrm{bb}}$ 0.84 with a single unified MDLM checkpoint—comparable to prior tri-modal models evaluated with a 200$\times$ larger inference budget. On GeneralQA, PROGEA further performs on par with frontier models such as Claude Opus 4.7 (Anthropic, 2026) and AMix2 (Qiu et al., 2026).

# 2 Related Work

## 2.1 Diffusion Language Models

Autoregressive (AR) language models generate sequences through a left-to-right factorization, which imposes a fixed generation order. Diffusion language models (dLLMs) provide an alternative by modeling text generation as an iterative denoising process, allowing tokens to be predicted with bidirectional context. Recent advances have enabled dLLMs to rapidly close the performance gap with AR LLMs (Bie et al., 2026; Team et al., 2026), establishing diffusion-based language modeling as a promising alternative to the dominant paradigm.

Foundational studies (Sahoo et al., 2024; Shi et al., 2025; Ou et al., 2026) established the formulation of objective as *a time-independent, weighted sum of standard cross-entropy losses*. Recent studies have further explored block-based formulations that introduce semi-autoregressive generation (Arriola et al., 2025), along with scalable training and sampling strategies (Zhou et al., 2026; Bie et al., 2025), and novel architectures (Zhu et al., 2026; Team et al., 2026).

## 2.2 Unified Tri-modal Models

Early tri-modal unified models primarily focus on representation-level alignment, mapping sequence, structure, and text into a shared embedding space through contrastive learning (Su et al., 2025; Zhang et al., 2024). ESM3 (Hayes et al., 2025) moves beyond alignment toward generative unification. BioMatrix explores native tri-modal generation with a single AR LLM backbone (Pei et al., 2026), while our work concurrently investigates the analogous unified formulation with dLLM framework.

# 3 Methods

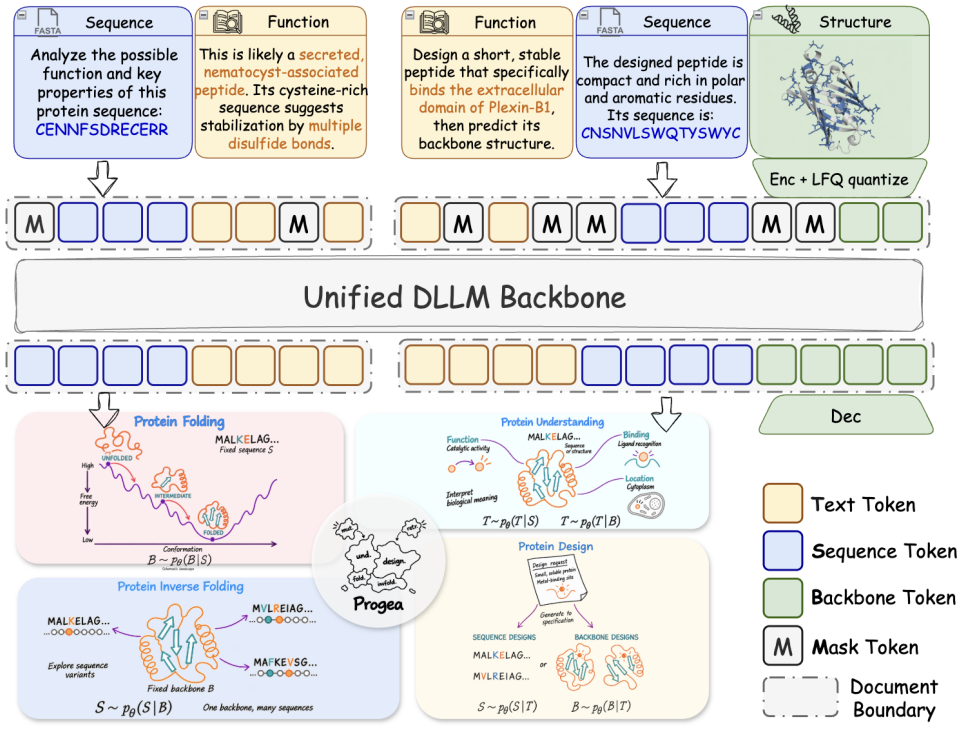



Figure 2: Overview of Progea.

### 3.1 SHARED EMBEDDING

We adopt token-space early-fusion. Specifically, we extend the Qwen3 vocabulary (Yang et al., 2025) with dedicated sequence and quantized structure tokens, allowing all modalities to be processed by a single dLLM from the first layer without separate modality encoders or cross-attention modules. A key challenge is how to discretize continuous protein structures into an LLM-compatible token space. We favor a tokenizer with three properties:

- **Generative.** The tokenizer should preserve sufficient geometric information for high-fidelity backbone reconstruction, rather than serving only retrieval search (Van Kempen et al., 2024). This is naturally supported by autoencoding objectives (Kingma & Welling, 2022; van den Oord et al., 2018), optionally regularized by auxiliary semantic supervision (Hayes et al., 2025). Recent work further shows that diffusion- or flow-matching-based decoders can substantially improve reconstruction fidelity, both in visual tokenization (Shaulov et al., 2026; Chen et al., 2025) and protein structure tokenization (Dilip et al., 2026b; Giri et al., 2026), by modeling a conditional distribution over reconstructions rather than relying on a deterministic decoder.
- **Local.** Each token should correspond to a local structural context, preserving residue-level positional correspondence and allowing contiguous tokens to represent local substructures and motifs. This contrasts with global tokenization schemes (Dilip et al., 2026a), which may weaken residue-level grounding.
- **Codebook-free.** Codebook-free quantization, such as LFQ (Yu et al., 2024), maps tokens to vertices of an $n$-dimensional hypercube, providing $2^n$ discrete codes with a natural Hamming geometry that can be readily absorbed into the extended embedding space, as shown in 4.2.1.

Codebook-free quantization is a convenient choice rather than a strict requirement; more generally, the tokenizer geometry should be faithfully represented in the language-model embedding space, which can also emerge through large-scale training (Wang et al., 2024a; Team, 2025). Following these principles, we use the DPLM2 structure tokenizer (Wang et al., 2024c).

### 3.2 UNIFIED ARCHITECTURE

Given the unified token space, we use a single dLLM backbone to jointly model protein sequence, backbone structure, and natural language, as illustrated in 2. All modalities share the same Transformer and can flexibly serve as conditions or generation targets, enabling both intra- and cross-modal generation without modality-specific heads.

We instantiate two variants with the same architecture but different generation factorizations: PROGEA-MDLM uses fully non-causal masked diffusion, while PROGEA-BDLM applies bidirectional diffusion within blocks and causal factorization across blocks. This provides a controlled comparison of how bidirectionality affects protein modeling.

$$\begin{aligned}
\mathcal{L}_{\text{MDLM}}(\theta) &= -\mathbb{E}_{t,x_0\sim p_{\text{data}}}\mathbb{E}_{x_t\sim q(x_t|x_0)}\left[\frac{1}{t}\sum_{i=1}^{L}\mathbf{1}\left[x_t^i=\mathbf{M}\right]\log p_\theta\left(x_0^i \mid x_t\right)\right] \\
\mathcal{L}_{\text{BDLM}}(\theta) &= -\mathbb{E}_{t,x_0\sim p_{\text{data}}}\mathbb{E}_{x_t\sim q(x_t|x_0)}\left[\frac{1}{t}\sum_{b=1}^{B}\sum_{i\in\mathcal{I}_b}\mathbf{1}[x_t^i=\mathbf{M}]\log p_\theta\left(x_0^i \mid x_0^{[<b]}, x_t^{[b]}\right)\right],
\end{aligned} \tag{1}$$

Here, $\mathbf{M}$ denotes the mask token, $L$ is the sequence length, and $B$ is the number of blocks with $\mathcal{I}_b$ denoting the token indices in block $b$. For BDLM, $x_0^{[<b]}$ denotes the clean tokens in preceding blocks, while $x_t^{[b]}$ denotes the corrupted tokens in the current block; thus diffusion is bidirectional within each block but causal across blocks.

### 3.3 MULTI-STAGE TRAINING

Protein sequence bridges language and backbone structure in our training strategy. We first align sequence with language during continual pre-training (CPT), then connect both to structure through

multimodal supervised fine-tuning (SFT). Given the scarcity of structural data, structure is introduced only during SFT. Starting from Qwen3-4B, we proceed through AR-CPT for protein knowledge acquisition, A2D-CPT for adaptation to diffusion modeling (Zhou et al., 2026), and SFT for all six directed cross-modal transformations (Table 1).

Table 1: Training configuration. T/S/B: text/sequence/backbone; ST: sequence–text pairs; X2X: all six cross-modal directions. Lengths are in tokens.

| | **Stage 1** | **Stage 2** | **Stage 3** |
|---|---|---|---|
| Training Paradigm | AR-CPT | A2D-CPT | SFT |
| Context Length | 4096 | 4096 | 4096 |
| Block Size | 1 | $8 \rightarrow 16 \rightarrow 32$ | 32 (bdlm) / 4096 (mdlm) |
| Assistant Mask | $\times$ | $\times$ | ✓ |
| Document Mask | $\times$ | $\times$ | ✓ |
| Task Type | T + S + ST | T + S + ST | X2X |

# 4 Experiments

## 4.1 Quantitative Performance

### 4.1.1 Evaluation Settings

**Benchmarks**. We prioritize evaluations that reflect the current frontier of protein modeling. For example, ProteinArena (Qiu et al., 2026) refines PDFBench (Kuang et al., 2025), providing a more recent and rigorous benchmark for sequence understanding and design. The benchmarks are summarized as follows:

- **Folding Tasks**: CAMEO2022 (Jing et al., 2023), PDB-date (Campbell et al., 2024)
- **Inverse Folding Tasks**: CAME20O22 (Jing et al., 2023), PDB-date (Campbell et al., 2024)
- **Protein Analysis Tasks**: ProteinArena (Qiu et al., 2026), OPI-struct Xiao et al. (2026)
- **Protein Design Tasks**: ProteinArena (Qiu et al., 2026)

**Baselines**. Our baseline selection follows two principles. *Capability*: we include specialized protein models and frontier LLMs as task-level performance references. To maintain comparable input requirements, we exclude AlphaFold-family configurations that require MSA inputs. *Paradigm*: we include native multimodal models spanning text–sequence, sequence–structure, and text–sequence–structure modeling, covering autoregressive, masked diffusion, and block diffusion generation paradigms. Together, these comparisons assess both Progea's task performance and the implications of its modality coverage and generation paradigm.

### 4.1.2 Folding and Inverse folding

As shown in Table 2, Progea substantially pushes forward the folding capability of unified protein models, while remaining competitive on inverse folding. We highlight two observations from these results.

**Bidirectional context benefits structure modeling.** On the PDB date split, Progea-MDLM substantially outperforms the re-evaluated BioMatrix checkpoint, improving folding TM-score from 0.68 to 0.84 and reducing RMSD from 10.70 to 5.55. It even approaches BioMatrix@200's reported performance (TM-score 0.84, RMSD 5.38). Unlike autoregressive generation, protein geometry is not governed by a left-to-right dependency order: spatially adjacent residues may be far apart in sequence. Causal attention prevents a residue from directly accessing such partners when they occur later in the sequence, whereas bidirectional attention permits this context. The consistent MDLM-over-BDLM-over-AR advantage supports this interpretation, which we further investigate in Sect. 4.2.2.

**Global geometric consistency remains a challenge.** Progea-MDLM reaches a PDB $\mathrm{TM}_{\mathrm{bb}}$ of 0.84 without explicit triangular modules, but still trails ESMFold (0.93). This gap suggests room to

improve how the model coordinates interactions among residues beyond sequence–structure translation. We aim to improve this capability through self-training on model-generated data, preserving the simplicity of modern LLM architectures rather than introducing computationally intensive triangular modules.

Table 2: **Folding and inverse folding.** Gray: group best (including ties). †: DPLM-2 paper. ∗: BioMatrix paper (Pei et al., 2026), $N = 200/50$ for folding/inverse folding; metrics as reported. Other results are re-evaluated; –: unavailable.

| Method | Params | Folding | | | | Inverse Folding | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | **CAMEO 2022** (183) | | **PDB date split** (449) | | **CAMEO 2022** (183) | | **PDB date split** (449) | |
| | | TMbb/ca ↑ | RMSDbb ↓ | TMbb/ca ↑ | RMSDbb ↓ | AAR ↑ | scTMbb ↑ | AAR ↑ | scTMbb ↑ |
| *Frontier Folding/Inverse Folding Models* | | | | | | | | | |
| ESMFold/IF | 3.7B/0.1B | 0.90 / 0.85 | 3.97 | 0.93 / 0.91 | 2.69 | 52.96 | 0.90 | 59.20 | 0.95 |
| SimpleFold | 3B | 0.89 / 0.83 | 4.38 | 0.93 / 0.91 | 2.48 | – | – | – | – |
| PVQD† | - | – / 0.81 | 4.08 | – | – | – | – | – | – |
| *Sequence–Structure Co-generation Models* | | | | | | | | | |
| MultiFlow† | 21.8M | 0.50 / – | 17.84 | 0.53 / – | 15.64 | 32.28 | 0.87 | 37.74 | 0.94 |
| ESM3 | 1.4B | 0.86 / 0.79 | 5.18 | 0.89 / 0.84 | 4.10 | 48.53 | 0.90 | 51.64 | 0.95 |
| DPLM2 | 3B | 0.81 / 0.72 | 7.00 | 0.86 / 0.81 | 4.19 | 51.48 | 0.88 | 61.18 | 0.92 |
| DPLM2 | 0.7B | 0.78 / 0.69 | 7.71 | 0.84 / 0.77 | 5.19 | 48.49 | 0.87 | 55.26 | 0.91 |
| DPLM2.1 | 0.7B | 0.82 / 0.74 | 6.52 | 0.91 / 0.88 | 2.87 | 53.79 | 0.88 | 77.62 | 0.93 |
| *Sequence–Structure–Text LLMs* | | | | | | | | | |
| BioMatrix@N* | 4B | – | – | 0.84 / – | 5.38 | – | – | 75.50 | 0.86 |
| BioMatrix | 4B | 0.50 / 0.37 | 18.02 | 0.68 / 0.58 | 10.70 | 19.76 | 0.57 | 54.67 | 0.75 |
| Progea-MDLM | 4B | 0.80 / 0.71 | 7.73 | 0.84 / 0.77 | 5.55 | 50.48 | 0.88 | 58.03 | 0.92 |
| Progea-BDLM | 4B | 0.76 / 0.66 | 9.07 | 0.80 / 0.73 | 7.19 | 48.32 | 0.87 | 55.00 | 0.91 |

### 4.1.3 PROTEIN UNDERSTANDING

Table 3: **Sequence and structure understanding.** Accuracy (%, ↑). F&F: Function & Family; I&B: Interaction & Binding; L&M: Location & Modification; Other: physicochemical/structural properties. Choice: option accuracy; Semantic: final-answer semantic correctness. Gray: group best; –: unavailable. †: BioMatrix uses sequence+structure input and nearest-option mapping for Choice; its Semantic score uses a different judge prompt.

| Method | Params | GeneralQA | | | | | StructureMCQA | |
|---|---|---|---|---|---|---|---|---|
| | | Overall (481) | F&F (150) | I&B (152) | L&M (153) | Other (26) | Choice (3,918) | Semantic (3,918) |
| *Frontier LLMs* | | | | | | | | |
| Claude Opus 4.7 | – | 62.37 | 61.33 | 62.50 | 66.67 | 42.31 | – | – |
| Claude Opus 4.6 | – | 48.02 | 32.67 | 48.68 | 58.17 | 73.08 | – | – |
| Gemini 3 Flash Thinking | – | 49.48 | 42.00 | 46.05 | 59.48 | 53.85 | – | – |
| GPT-5.5 | – | 48.44 | 32.00 | 46.05 | 66.01 | 53.85 | – | – |
| DeepSeek-V4-Pro | 1.6T | 46.57 | 36.67 | 45.39 | 56.21 | 53.85 | – | – |
| Qwen3.7-Max | – | 43.45 | 31.33 | 43.42 | 54.25 | 50.00 | – | – |
| GPT-OSS-120B | 120B | 43.24 | 32.00 | 38.82 | 56.21 | 57.69 | – | – |
| GLM-5.1 | 0.8T | 38.05 | 26.67 | 37.50 | 49.02 | 42.31 | – | – |
| *Sequence–Text LLMs* | | | | | | | | |
| AMix2-dLLM | 4B | 66.11 | 58.67 | 67.11 | 71.24 | 73.08 | – | – |
| AMix2-AR | 4B | 62.58 | 56.00 | 61.84 | 67.32 | 76.92 | – | – |
| *Sequence–Structure–Text LLMs* | | | | | | | | |
| BioMatrix | 4B | 24.32 | 29.33 | 21.71 | 24.18 | 11.54 | 35.76† | 12.61† |
| Progea-MDLM | 4B | 56.13 | 50.00 | 57.89 | 58.82 | 65.38 | 52.25 | 28.08 |
| Progea-BDLM | 4B | 62.16 | 52.67 | 63.82 | 69.93 | 61.54 | 76.72 | 76.19 |

We evaluate sequence understanding on GeneralQA and structure understanding on StructureMCQA, the multiple-choice subset of OPI-Struc. Table 3 summarizes both tasks.

**Progea combines frontier-level sequence understanding with backbone-grounded functional reasoning.** Progea-BDLM achieves 62.16% accuracy on GeneralQA, closely matching Claude Opus 4.7 (62.37%), and 76.72% on StructureMCQA with backbone-only inputs. The latter remains below STELLA's reported 80.56% (Xiao et al., 2026), but input modalities and test sets differ: STELLA and BioMatrix's published structure-understanding evaluations use both sequence and structure (Xiao et al., 2026; Pei et al., 2026), whereas backbone geometry alone may leave some functional questions underdetermined.

### 4.1.4 Protein Design

Table 4: **Text-conditioned protein design.** All metrics use a 0–100 scale. SR covers all 870 prompts; other metrics average over evaluated sequences. Gray: group best; –: unavailable; Params: total model size. ‡: likelihood-ranked best-of-10; †: pLDDT on 819 sequences.

| Method | Params | SR ↑ | Repetition | | Foldability | Function | Novelty |
|---|---|---|---|---|---|---|---|
| | | | Rep2 ↓ | Rep5 ↓ | pLDDT ↑ | IPRRecovery ↑ | Hard (Seq) ↑ |
| *Frontier LLMs* | | | | | | | |
| Claude Opus 4.7 Compact | - | 84.14 | 44.65 | 0.65 | 71.33 | 65.95 | 36.54 |
| Claude Opus 4.6 Compact8192 | - | 90.57 | 48.44 | 1.86 | 42.17 | 25.32 | 45.91 |
| Gemini 3.1 Pro Thinking | - | 100.00 | 44.21 | 7.17 | 48.67 | 25.92 | 45.59 |
| GPT-5.5 | - | 99.89 | 53.19 | 3.86 | 43.95 | 25.21 | 44.21 |
| DeepSeek-V4-Pro | 1.6T | 85.06 | 50.32 | 8.14 | 45.03 | 25.14 | 51.25 |
| Qwen3.5-27B | 27B | 100.00 | 76.95 | 63.24 | 33.47 | 0.47 | 77.93 |
| GPT-OSS-120B | 120B | 100.00 | 46.77 | 2.20 | 31.85 | 3.56 | 53.42 |
| GLM-5.1 | 0.8T | 100.00 | 47.34 | 4.32 | 37.64 | 7.75 | 54.46 |
| *Protein Design Models* | | | | | | | |
| CFPGen | 0.7B | 36.67 | 58.91 | 13.65 | 75.78 | 31.99 | 48.43 |
| Chroma | 0.3B | 100.00 | 52.02 | 0.44 | 60.46 | 0.02 | 59.69 |
| ESM3 | 1.4B | 94.94 | 69.46 | 21.64 | 60.56 | 19.20 | 73.70 |
| Pinal‡ | 2B | 100.00 | 42.47 | 3.23 | 80.33 | 47.78 | 38.60 |
| ProDVa† | 1.8B | 99.08 | 25.35 | 6.67 | 70.22 | 27.43 | 23.49 |
| ProteinDT | 0.7B | 100.00 | 52.46 | 2.03 | 38.88 | 8.63 | 61.94 |
| *Sequence–Text LLMs* | | | | | | | |
| AMix2-dLLM | 4B | 69.31 | 45.22 | 1.23 | 74.75 | 62.61 | 30.57 |
| AMix2-AR | 4B | 93.22 | 44.90 | 1.61 | 74.89 | 39.92 | 7.57 |
| *Sequence–Structure–Text LLMs* | | | | | | | |
| BioMatrix | 4B | 98.62 | 50.93 | 18.03 | 60.32 | 31.40 | 28.69 |
| Progea-MDLM | 4B | 56.55 | 72.50 | 30.89 | 50.42 | 23.27 | 72.42 |
| Progea-BDLM | 4B | 62.18 | 43.79 | 1.63 | 70.65 | 47.50 | 39.29 |

**Text-to-sequence.** Table 4 shows both the progress and remaining limitations of our unified models in text-conditioned protein design. Without further design-specific refinement, Progea-BDLM improves over the prior unified tri-modal baseline BioMatrix in foldability (pLDDT: 70.65 vs. 60.32), functional recovery(IPRRecovery: 47.50 vs. 31.40), and repetition control, although its lower SR indicates less reliable production of evaluable sequences. However, it still trails AMix2-dLLM in SR (62.18 vs. 69.31), pLDDT (70.65 vs. 74.75), and IPRRecovery (47.50 vs. 62.61), leaving a clear gap in functional design quality.

Within our models, BDLM consistently improves over MDLM in foldability, functional recovery, and repetition control. Without adaptive padding, MDLM tends to fill the inference budget, potentially exacerbating repetitive continuations and reducing sequence quality. Such repetition is a longstanding challenge in diffusion-based protein generation (Wang et al., 2024b) and persists despite the improvements from block diffusion.

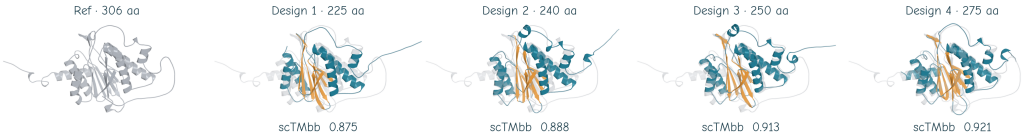



Figure 3: Text-conditioned PrcB backbone designs. Gray: aligned AF reference. scTMbb measures self-consistency through ProteinMPNN and ESMFold.

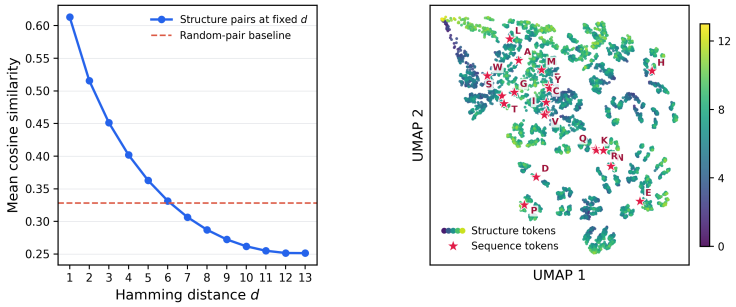
These results indicate progress over prior unified tri-modal modeling, while highlighting the need for further design-specific refinement in adaptive length and format control, repetition reduction, and functional fidelity. All quality comparisons are conditional on each model's evaluated subset.

**Text-to-backbone.** Without a suitable benchmark in our current evaluation setup, we conduct a series of text-to-backbone case studies across multiple functional descriptions, target lengths, and sampling seeds. Figure 3 presents a representative PrcB example: the selected designs preserve the reference-like fold with local variations and achieve scTMbb scores of 0.875–0.921, consistent with retrieval-like behavior. We hypothesize that the dataset's specific, family-level descriptions constrain the design space; structural self-consistency alone does not establish functional fidelity.

## 4.2 Qualitative Analysis

### 4.2.1 Embedding Geometry

**Embedding geometry is preserved.** Unlike vision models that often reuse or initialize from codebook embeddings, PROGEA learns its structure-token embeddings from scratch. As shown in Fig. 4, embedding similarity decreases consistently with Hamming distance in the LFQ code space, indicating that the tokenizer topology is largely preserved. UMAP further reveals organized ring-like manifolds, with several amino-acid tokens near characteristic structural neighborhoods, suggesting stable local sequence–structure correspondences. Meanwhile, structural regions distant from any amino-acid token reflect the context-dependent, one-to-many nature of folding.




(a) Similarity vs. Code distance. (b) Visualization of embedding space.

Figure 4: **Geometric organization of the shared token space.** (a) Embedding similarity decreases monotonically with Hamming distance in the quantized n-bit structure-code space, as described in 3.1. (b) UMAP visualization shows that structure tokens form an organized manifold while sequence tokens occupy the same global embedding space; color intensity indicates the Hamming weight of each structure token, i.e., the number of 1s in its n-bit code.

### 4.2.2 Probing the Folding Mechanism

**Folding involves residue-level translation.** We initially attributed DPLM2's diagonal sequence–structure attention to its aligned positional encoding. Yet similar patterns persist under global positional encoding (Figure 5), suggesting that residue-level translation can emerge from the folding task itself. This motivates a conceptual pathway: sequence $\rightarrow$ contact constraints $\rightarrow$ structure.

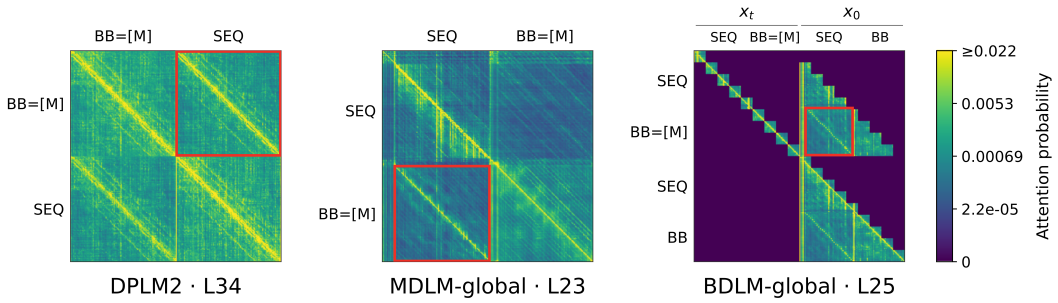



Figure 5: **Sequence–structure attention on 6WEB.** Head-averaged maps for DPLM2, Progea-MDLM, and Progea-BDLM. Red boxes mark structure queries attending to sequence keys. Each layer is selected by maximal conditional attention mass within $\pm 2$ residues of the corresponding sequence position.

**Causal generation requires harder anticipation.** For contacting residues, causal masking prevents the earlier structure token from directly accessing its partner's future structural state. Early next-token decisions must therefore anticipate nonlocal constraints. This is consistent with weak BioMatrix@1 but better BioMatrix@200.

## 5 Limitation and Future Direction

PROGEA currently focuses on protein-only modeling and does not yet cover richer biomolecular systems such as complexes, ligands, or nucleic acids. Moreover, despite strong performance across diverse tasks, a gap remains to the strongest task-specialized models, particularly on structure-intensive problems such as protein folding.

Looking forward, we see two promising directions. (1) Further improving PROGEA's capability ceiling through multi-codebook structure representations, stronger structure decoders (Wang et al., 2024c; Hsieh et al., 2025), and targeted augmentation for short proteins and challenging cases. (2) Exploiting interleaved generation for test-time scaling, where composable capabilities are iteratively generated, evaluated, and refined with additional inference-time computation. With sufficiently strong base capabilities, UniProt-scale sequence–structure–function corpora could be available, providing the substrate for closed-loop self-improvement.

## 6 Conclusion

We presented PROGEA, a native unified tri-modal diffusion language model that brings protein sequence, structure, and natural language into a single generative framework. By casting six core capabilities across the sequence–structure–function space as conditional generation, PROGEA achieves broadly competitive performance at the 4B scale while substantially strengthening the folding capability of unified protein models, demonstrating that strong structure modeling can coexist with broad understanding, generation, and native language interaction within a single model.

Our analyses further support the importance of non-causal modeling for protein structure and show that the geometric organization induced by the structure tokenizer can be naturally absorbed into the language-model embedding space. More broadly, PROGEA turns specialized protein capabilities into composable generative primitives, enabling interleaved workflows and providing a foundation for future test-time scaling and closed-loop self-improvement through self-generated data.

### AI use statement

We used Codex to assist with language editing and rewriting, code assistance, and experiment orchestration, including job submission and monitoring of training and evaluation statistics. AI tools were not used to formulate the central research hypotheses, design the experimental methodology.

All AI-assisted text, code, and experimental workflows were reviewed and verified by the authors. The authors take full responsibility for the final content of this work.

### ETHICS STATEMENT

This work is computational and does not involve human participants or animal experiments. While unified protein modeling and design may enable beneficial applications in biological research and protein engineering, generative protein models may also have potential dual-use implications. The outputs of PROGEA are in silico and designs and should be subject to appropriate experimental validation and domain-specific safety assessment before practical use. We encourage responsible use of the proposed model and methods.

### REPRODUCIBILITY STATEMENT

We provide the information needed to reproduce the main experiments across the appendix and supplementary materials. Appendix C describes the composition of the supervised fine-tuning datasets. We further specify benchmark-specific evaluation protocols and inference settings in Appendix D. To facilitate independent verification, we provide anonymous source code, evaluation scripts, and case-level inference outputs as supplementary materials.

## REFERENCES

The gene ontology resource: enriching a gold mine. *Nucleic acids research*, 49(D1):D325–D334, 2021.

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

Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J Ballard, Joshua Bambrick, et al. Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature*, 630(8016):493–500, 2024.

Woody Ahern, Jason Yim, Doug Tischer, Saman Salike, Seth M Woodbury, Donghyo Kim, Indrek Kalvet, Yakov Kipnis, Brian Coventry, Han Raut Altae-Tran, et al. Atom-level enzyme active site scaffolding using rfdiffusion2. *Nature Methods*, 23(1):96–105, 2026.

Sarah Alamdari, Nitya Thakkar, Rianne Van Den Berg, Neil Tenenholtz, Robert Strome, Alan M Moses, Alex X Lu, Nicolò Fusi, Ava P Amini, and Kevin K Yang. Protein generation with evolutionary diffusion: sequence is all you need. *BioRxiv*, pp. 2023–09, 2023.

Anthropic. Claude opus 4.7 system card. Technical report, Anthropic, 2026.

Marianne Arriola, Aaron Gokaslan, Justin T. Chiu, Zhihan Yang, Zhixuan Qi, Jiaqi Han, Subham Sekhar Sahoo, and Volodymyr Kuleshov. Block diffusion: Interpolating between autoregressive and diffusion language models, 2025. URL https://arxiv.org/abs/2503.09573.

Tiwei Bie, Maosong Cao, Kun Chen, Lun Du, Mingliang Gong, Zhuochen Gong, Yanmei Gu, Jiaqi Hu, Zenan Huang, Zhenzhong Lan, et al. Llada2. 0: Scaling up diffusion language models to 100b. *arXiv preprint arXiv:2512.15745*, 2025.

Tiwei Bie, Jiansheng Cai, Maosong Cao, Xiang Cao, Binbin Chen, Fuyuan Chen, Kun Chen, Yuqi Ding, Lun Du, Zhuocheng Gong, et al. Llada2. 2: Enabling agentic diffusion language models via levenshtein editing. Technical report, CSPaper OpenPrint, 2026.

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 arXiv:2402.04997*, 2024.

Yinbo Chen, Rohit Girdhar, Xiaolong Wang, Sai Saketh Rambhatla, and Ishan Misra. Diffusion autoencoders are scalable image tokenizers. *arXiv preprint arXiv:2501.18593*, 2025.

Zhiyuan Chen, Tianhao Chen, Chenggang Xie, Yang Xue, Xiaonan Zhang, Jingbo Zhou, and Xiaomin Fang. Unifying sequences, structures, and descriptions for any-to-any protein generation with the large multimodal model helixprotx. *arXiv preprint arXiv:2407.09274*, 2024.

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

Justas Dauparas, Gyu Rie Lee, Robert Pecoraro, Linna An, Ivan Anishchenko, Cameron Glasscock, and David Baker. Atomic context-conditioned protein sequence design using ligandmpnn. *Nature Methods*, 22(4):717–723, 2025.

Rohit Dilip, Ayush Varshney, and David Van Valen. Adaptive Protein Tokenization, February 2026a. URL http://arxiv.org/abs/2602.06418. arXiv:2602.06418 [cs].

Rohit Dilip, Evan Zhang, Ayush Varshney, and David Van Valen. Flow autoencoders are effective protein tokenizers. In *International Conference on Learning Representations*, volume 2026, pp. 145901–145926, 2026b.

Ken A. Dill and Justin L. MacCallum. The protein-folding problem, 50 years on. *Science*, 338 (6110):1042–1046, 2012. doi: 10.1126/science.1219021. URL https://www.science.org/doi/abs/10.1126/science.1219021.

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

Tomas Geffner, Kieran Didi, Zuobai Zhang, Danny Reidenbach, Zhonglin Cao, Jason Yim, Mario Geiger, Christian Dallago, Emine Kucukbenli, Arash Vahdat, et al. Proteina: Scaling flow-based protein structure generative models. In *International Conference on Learning Representations*, volume 2025, pp. 98803–98851, 2025.

Tomas Geffner, Kieran Didi, Zhonglin Cao, Danny Reidenbach, Zuobai Zhang, Christian Dallago, Emine Kucukbenli, Karsten Kreis, and Arash Vahdat. La-proteina: Atomistic protein generation via partially latent flow matching. In *International Conference on Learning Representations*, volume 2026, pp. 67350–67395, 2026.

Nabin Giri, Steven Farrell, and Kristofer E Bouchard. Yeti: A compact protein structure tokenizer for reconstruction and multi-modal generation. *arXiv preprint arXiv:2605.09981*, 2026.

Chengyue Gong, Xinshi Chen, Yuxuan Zhang, Yuxuan Song, Hao Zhou, and Wenzhi Xiao. Protenix-mini: Efficient structure predictor via compact architecture, few-step diffusion and switchable plm. *arXiv preprint arXiv:2507.11839*, 2025.

Xiao-Yu Guo, Yi-Fan Li, Yuan Liu, Xiaoyong Pan, and Hong-Bin Shen. Ab-initio amino acid sequence design from protein text description with protdat. *Nature Communications*, 16(1):10544, 2025.

Thomas Hayes, Roshan Rao, Halil Akin, Nicholas J Sofroniew, Deniz Oktay, Zeming Lin, Robert Verkuil, Vincent Q Tran, Jonathan Deaton, Marius Wiggert, et al. Simulating 500 million years of evolution with a language model. *Science*, 387(6736):850–858, 2025.

Cheng-Yen Hsieh, Xinyou Wang, Daiheng Zhang, Dongyu Xue, Fei Ye, Shujian Huang, Zaixiang Zheng, and Quanquan Gu. Elucidating the design space of multimodal protein language models. *arXiv preprint arXiv:2504.11454*, 2025.

Bowen Jing, Stephan Eismann, Patricia Suriana, Raphael JL Townshend, and Ron Dror. Learning from protein structure with geometric vector perceptrons. *arXiv preprint arXiv:2009.01411*, 2020.

Bowen Jing, Ezra Erives, Peter Pao-Huang, Gabriele Corso, Bonnie Berger, and Tommi Jaakkola. Eigenfold: Generative protein structure prediction with diffusion models. *arXiv preprint arXiv:2304.02198*, 2023.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2022. URL https://arxiv.org/abs/1312.6114.

Jiahao Kuang, Nuowei Liu, Jie Wang, Changzhi Sun, Tao Ji, and Yuanbin Wu. PDFBench: A Benchmark for De novo Protein Design from Function, September 2025. URL http://arxiv.org/abs/2505.20346. arXiv:2505.20346 [cs].

Shengchao Liu, Yanjing Li, Zhuoxinran Li, Anthony Gitter, Yutao Zhu, Jiarui Lu, Zhao Xu, Weili Nie, Arvind Ramanathan, Chaowei Xiao, et al. A text-guided protein design framework. *Nature Machine Intelligence*, 7(4):580–591, 2025a.

Yufeng Liu, Rui Wu, Xinyu Wang, Sheng Wang, Linghui Chen, Fudong Li, Quan Chen, and Haiyan Liu. Enhancing functional proteins through multimodal inverse folding with abacus-t. *Nature Communications*, 16(1):10177, 2025b.

Zicheng Ma, Chuanliu Fan, Zhicong Wang, Zhenyu Chen, Xiaohan Lin, Yanheng Li, Shihao Feng, Ziqiang Cao, Jun Zhang, and Yi Qin Gao. Prottex: Structure-in-context reasoning and editing of proteins with large language models. *Journal of Chemical Information and Modeling*, 65(13): 6599–6612, 2025.

Ali Madani, Bryan McCann, Nikhil Naik, Nitish Shirish Keskar, Namrata Anand, Raphael R Eguchi, Po-Ssu Huang, and Richard Socher. Progen: Language modeling for protein generation. *BioRxiv*, pp. 2020–03, 2020.

Jingyang Ou, Shen Nie, Kaiwen Xue, Fengqi Zhu, Jiacheng Sun, Zhenguo Li, and Chongxuan Li. Your absorbing discrete diffusion secretly models the conditional distributions of clean data, 2026. URL https://arxiv.org/abs/2406.03736.

Saro Passaro, Gabriele Corso, Jeremy Wohlwend, Mateo Reveiz, Stephan Thaler, Vignesh Ram Somnath, Noah Getz, Tally Portnoi, Julien Roy, Hannes Stark, et al. Boltz-2: Towards accurate and efficient binding affinity prediction. *BioRxiv*, 2025.

Qizhi Pei, Zhimeng Zhou, Yi Duan, Yiyang Zhao, Wei Li, Han Guo, Liang He, Chengping Li, Chang-Yu Hsieh, Conghui He, et al. Biomatrix: Towards a comprehensive biological foundation model spanning the modality matrix of sequences, structures, and language. *arXiv preprint arXiv:2606.22138*, 2026.

Bo Qiang, Chengyue Gong, Xinshi Chen, Yuxuan Zhang, and Wenzhi Xiao. Protenix-mini+: efficient structure prediction model with scalable pairformer. *arXiv preprint arXiv:2510.12842*, 2025.

Keyue Qiu, Yixin Wu, Lihao Wang, Yawen Ouyang, Jixiang Yu, Zihan Zhou, Changze Lv, Dongyu Xue, Yuxuan Song, Xinbo Zhang, et al. Amix-2: Establishing protein as a native modality in large language models. *arXiv preprint arXiv:2605.30963*, 2026.

Alexander Rives, Joshua Meier, Tom Sercu, Siddharth Goyal, Zeming Lin, Jason Liu, Demi Guo, Myle Ott, C. Lawrence Zitnick, Jerry Ma, and Rob Fergus. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proceedings of the National Academy of Sciences*, 118(15):e2016239118, 2021. doi: 10.1073/pnas.2016239118. URL https://www.pnas.org/doi/abs/10.1073/pnas.2016239118.

Subham Sekhar Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T. Chiu, Alexander Rush, and Volodymyr Kuleshov. Simple and Effective Masked Diffusion Language Models, November 2024. URL http://arxiv.org/abs/2406.07524. arXiv:2406.07524 [cs].

Ariel Shaulov, Itay Hazan, Lior Wolf, and Hila Chefer. Flowmo: Variance-based flow guidance for coherent motion in video generation. *Advances in Neural Information Processing Systems*, 38: 82725–82754, 2026.

Jiaxin Shi, Kehang Han, Zhe Wang, Arnaud Doucet, and Michalis K. Titsias. Simplified and generalized masked diffusion for discrete data, 2025. URL https://arxiv.org/abs/2406.04329.

Jin Su, Yan He, Shiyang You, Shiyu Jiang, Xibin Zhou, Xuting Zhang, Yuxuan Wang, Xining Su, Igor Tolstoy, Xing Chang, et al. A trimodal protein language model enables advanced protein searches. *Nature Biotechnology*, pp. 1–7, 2025.

Chai Discovery team, Jacques Boitreaud, Jack Dent, Matthew McPartlon, Joshua Meier, Vinicius Reis, Alex Rogozhonikov, and Kevin Wu. Chai-1: Decoding the molecular interactions of life. *BioRxiv*, pp. 2024–10, 2024.

Chai Discovery Team, Jacques Boitreaud, Jack Dent, Danny Geisz, Matthew McPartlon, Joshua Meier, Zhuoran Qiao, Alex Rogozhnikov, Nathan Rollins, Paul Wollenhaupt, et al. Zero-shot antibody design in a 24-well plate. *BioRxiv*, pp. 2025–07, 2025.

Chameleon Team. Chameleon: Mixed-modal early-fusion foundation models, 2025. URL https://arxiv.org/abs/2405.09818.

DiffusionGemma Team, Adrien Ali Taïga, James Assiene, Daniele Calandriello, Rahma Chaabouni, João Gante, Tamara von Glehn, Nate Keating, Chris Knutsen, Martin Kukla, et al. Diffusiongemma technical report. *arXiv preprint arXiv:2608.00146*, 2026.

Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning, 2018. URL https://arxiv.org/abs/1711.00937.

Michel Van Kempen, Stephanie S Kim, Charlotte Tumescheit, Milot Mirdita, Jeongjae Lee, Cameron LM Gilchrist, Johannes Söding, and Martin Steinegger. Fast and accurate protein structure search with foldseek. *Nature biotechnology*, 42(2):243–246, 2024.

Xinlong Wang, Xiaosong Zhang, Zhengxiong Luo, Quan Sun, Yufeng Cui, Jinsheng Wang, Fan Zhang, Yueze Wang, Zhen Li, Qiying Yu, Yingli Zhao, Yulong Ao, Xuebin Min, Tao Li, Boya Wu, Bo Zhao, Bowen Zhang, Liangdong Wang, Guang Liu, Zheqi He, Xi Yang, Jingjing Liu, Yonghua Lin, Tiejun Huang, and Zhongyuan Wang. Emu3: Next-token prediction is all you need, 2024a. URL https://arxiv.org/abs/2409.18869.

Xinyou Wang, Zaixiang Zheng, Fei Ye, Dongyu Xue, Shujian Huang, and Quanquan Gu. Diffusion language models are versatile protein learners. *arXiv preprint arXiv:2402.18567*, 2024b.

Xinyou Wang, Zaixiang Zheng, Fei Ye, Dongyu Xue, Shujian Huang, and Quanquan Gu. DPLM-2: A Multimodal Diffusion Protein Language Model, October 2024c. URL http://arxiv.org/abs/2410.13782. arXiv:2410.13782 [cs].

Yuyang Wang, Jiarui Lu, Navdeep Jaitly, Joshua Susskind, and Miguel Angel Bautista. Simplefold: Folding proteins is simpler than you think. In *International Conference on Learning Representations*, volume 2026, pp. 32027–32054, 2026.

Joseph L Watson, David Juergens, Nathaniel R Bennett, Brian L Trippe, Jason Yim, Helen E Eisenach, Woody Ahern, Andrew J Borst, Robert J Ragotte, Lukas F Milles, et al. De novo design of protein structure and function with rfdiffusion. *Nature*, 620(7976):1089–1100, 2023.

Jeremy Wohlwend, Gabriele Corso, Saro Passaro, Noah Getz, Mateo Reveiz, Ken Leidal, Wojtek Swiderski, Liam Atkinson, Tally Portnoi, Itamar Chinn, et al. Boltz-1 democratizing biomolecular interaction modeling. *BioRxiv*, pp. 2024–11, 2025.

Hongwang Xiao, Wenjun Lin, Xi Chen, Hui Wang, Kai Chen, Jiashan Li, Yuancheng Sun, Sicheng Dai, Boya Wu, and Qiwei Ye. Stella: A multimodal llm for protein functional annotation via unified sequence-structure encoding. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 25030–25049, 2026.

Minghao Xu, Xinyu Yuan, Santiago Miret, and Jian Tang. Protst: Multi-modality learning of protein sequences and biomedical texts. In *International conference on machine learning*, pp. 38749–38767. PMLR, 2023.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Zhenyu Yang, Xiaoxi Zeng, Yi Zhao, and Runsheng Chen. Alphafold2 and its applications in the fields of biology and medicine. *Signal transduction and targeted therapy*, 8(1):115, 2023.

Lijun Yu, José Lezama, Nitesh B. Gundavarapu, Luca Versari, Kihyuk Sohn, David Minnen, Yong Cheng, Vighnesh Birodkar, Agrim Gupta, Xiuye Gu, Alexander G. Hauptmann, Boqing Gong, Ming-Hsuan Yang, Irfan Essa, David A. Ross, and Lu Jiang. Language model beats diffusion – tokenizer is key to visual generation, 2024. URL https://arxiv.org/abs/2310.05737.

Li Zhang, Han Guo, Leah Schaffer, Young Su Ko, Digvijay Singh, Hamid Rahmani, Danielle Grotjahn, Elizabeth Villa, Michael Gilson, Wei Wang, et al. Proteinaligner: A multi-modal pretraining framework for protein foundation models. *bioRxiv*, pp. 2024–10, 2024.

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

Yi Zhou, Chan Lu, Yiming Ma, Wei Qu, Fei Ye, Kexin Zhang, Lan Wang, Minrui Gui, and Quanquan Gu. Seedfold: Scaling biomolecular structure prediction. *arXiv preprint arXiv:2512.24354*, 2025.

Zhanhui Zhou, Lingjie Chen, Hanghang Tong, and Dawn Song. dLLM: Simple Diffusion Language Modeling, February 2026. URL http://arxiv.org/abs/2602.22661. arXiv:2602.22661 [cs] version: 1.

Yuchen Zhu, Jing Shi, Chongjian Ge, Hao Tan, Yiran Xu, Wanrong Zhu, Jason Kuen, Koustava Goswami, Rajiv Jain, Yongxin Chen, Molei Tao, and Jiuxiang Gu. FLARE: Diffusion for Hybrid Language Model, 2026. URL https://arxiv.org/abs/2606.01774. Version Number: 1.

## A More Related Work

We review recent progress in specialized protein models and the key inductive biases behind their success, while highlighting their intrinsic connections to the modeling capabilities of modern LLMs.

**Folding.** Protein folding maps amino acid sequences to 3D structures. Modern folding systems rely heavily on triangular residue interactions to capture higher-order geometric consistency and long-range contacts (Yang et al., 2023; Abramson et al., 2024), with recent advances driven by scaling (Zhou et al., 2025), distillation (Gong et al., 2025; Qiang et al., 2025), and richer conditioning (Wohlwend et al., 2025; Passaro et al., 2025; team et al., 2024; Team et al., 2025). From a graph perspective, the effectiveness of triangular modeling can be understood through two complementary components: *translation self-loops*, which map residue context into local structure, and *interaction edges*, which propagate information across residues to enforce global consistency. Although LLMs lack explicit interaction-edge modules, recent results show that strong folding can emerge from largely data-driven Transformer architectures (Wang et al., 2026).

**Inverse folding.** Modern inverse folding models (Dauparas et al., 2022; 2025; Liu et al., 2025b) typically encode protein backbones into geometry-aware representations before light-weight sequence decoding. This closely mirrors the encoder of a protein structure tokenizer, which likewise maps 3D backbone geometry into a compact latent representation, suggesting a natural bridge from inverse folding to language-model-based generation.

**Sequence/Structure understanding.** Protein understanding is shifting from fixed-label prediction toward open-ended functional semantics. Sequence models increasingly align proteins with textual

description (Elnaggar et al., 2021; Xu et al., 2023), while structure-aware models inject geometric context (Jing et al., 2020; Zhang et al., 2022). This makes natural language a natural interface between molecular representations and biological knowledge.

**Sequence/Structure design.** Protein design is likewise moving toward richer conditioning: sequence models progress from functional tags (Madani et al., 2020) to language descriptions (Liu et al., 2025a; Guo et al., 2025; Qiu et al., 2026) and motif constraints (Alamdari et al., 2023; Wang et al., 2024b), while structure generators incorporate increasingly detailed physicochemical and geometric constraints (Watson et al., 2023; Ahern et al., 2026; Geffner et al., 2025; 2026). This trend naturally aligns protein design with instruction-conditioned generation in modern LLMs, where heterogeneous design intents can be expressed and composed through a shared interface.

# B Diffusion Preliminaries

Table 5 summarizes the notation used throughout this appendix.

Table 5: Summary of notation.

| Symbol | Description |
|---|---|
| $\mathcal{V},\ V$ | Token vocabulary and its size (excluding the mask) |
| $x_0,\ x_t$ | Clean and corrupted sequences of length $L$ |
| $x_0^i,\ x_t^i$ | Corresponding one-hot tokens at position $i$ |
| $\mathbf{M}$ | Absorbing mask token, $e_{V+1}$ |
| $s,\ t$ | Diffusion times, $0 \le s < t \le 1$ |
| $\alpha_t$ | Token-retention probability, $\alpha_t = 1 - t$ |
| $Q_t$ | Forward transition matrix from time 0 to $t$ |
| $q(\cdot \mid \cdot)$ | Forward or posterior probability mass function |
| $\mathrm{Cat}(z;\pi)$ | Categorical probability mass at $z$ with vector $\pi$ |
| $\pi_\theta^i(x_t)$ | Predicted clean-token probability vector |
| $p_\theta(\cdot \mid \cdot)$ | Model probability mass function |
| $p_{\text{data}}$ | Clean-data distribution |
| $\mathbf{1}[\cdot]$ | Indicator function; $\mathbf{1}$ alone is the all-ones vector |
| $b,\ K$ | Block size and number of blocks, $K = \lceil L/b \rceil$ |
| $x_0^{[k]},\ x_0^{[<k]}$ | Current block and preceding blocks |
| $c$ | Observed conditioning context |

## B.1 Forward Transition Process

Let $x_0 = (x_0^1, \ldots, x_0^L)$ be a clean sequence sampled from $p_{\text{data}}(x_0)$. Each token is represented by a one-hot column vector in $\{e_1, \ldots, e_V\} \subset \mathbb{R}^{V+1}$, and $\mathbf{M} = e_{V+1}$ denotes the absorbing mask state. We write $\mathrm{Cat}(z;\pi)$ for the categorical probability mass evaluated at $z$ with probability vector $\pi$; thus $q(\cdot \mid \cdot)$ denotes a probability mass function, not a probability vector.

The forward process is defined by the cumulative transition matrix

$$Q_t = \alpha_t I + (1 - \alpha_t)\mathbf{1}\mathbf{M}^\top \in \mathbb{R}^{(V+1)\times(V+1)}, \tag{2}$$

where $t \in [0,1]$, $\alpha_t$ is a monotonically decreasing retention schedule with $\alpha_0 = 1$ and $\alpha_1 = 0$, and $\mathbf{1}$ is the all-ones column vector. Rows index the initial state and columns index the resulting state. A token retains its identity with probability $\alpha_t$ and becomes masked with probability $1 - \alpha_t$; the mask state is absorbing.

Applying this transition independently at each position gives

$$\begin{aligned} q(x_t \mid x_0) &= \prod_{i=1}^{L} q(x_t^i \mid x_0^i), \\ q(x_t^i \mid x_0^i) &= \mathrm{Cat}\big(x_t^i; Q_t^\top x_0^i\big) \\ &= \mathrm{Cat}\big(x_t^i; \alpha_t x_0^i + (1 - \alpha_t)\mathbf{M}\big)\,. \end{aligned} \tag{3}$$

For the linear schedule $\alpha_t = 1 - t$, each position is masked with probability $t$, and $x_1 = (\mathbf{M}, \ldots, \mathbf{M})$.

### B.2 Reverse Denoising Process

For $0 \le s < t \le 1$, the forward posterior is

$$q(x_s^i \mid x_t^i, x_0^i) = \begin{cases} \mathrm{Cat}\big(x_s^i; x_0^i\big), & x_t^i \neq \mathbf{M}, \\ \mathrm{Cat}\left(x_s^i; \dfrac{\alpha_s - \alpha_t}{1 - \alpha_t} x_0^i + \dfrac{1 - \alpha_s}{1 - \alpha_t}\mathbf{M}\right), & x_t^i = \mathbf{M}. \end{cases} \tag{4}$$

Let $\pi_\theta^i(x_t) \in \mathbb{R}^{V+1}$ denote the predicted clean-token probability vector at position $i$, with zero mass on $\mathbf{M}$. Its corresponding probability mass function is

$$p_\theta(x_0^i \mid x_t) = \mathrm{Cat}\big(x_0^i; \pi_\theta^i(x_t)\big). \tag{5}$$

Substituting this vector for the unknown clean token in the masked branch gives, under $\alpha_t = 1 - t$,

$$p_\theta(x_s^i \mid x_t) = \begin{cases} \mathrm{Cat}\big(x_s^i; x_t^i\big), & x_t^i \neq \mathbf{M}, \\ \mathrm{Cat}\left(x_s^i; \dfrac{t - s}{t}\pi_\theta^i(x_t) + \dfrac{s}{t}\mathbf{M}\right), & x_t^i = \mathbf{M}. \end{cases} \tag{6}$$

This ancestral update preserves visible tokens and progressively reveals masked tokens as time decreases from 1 to 0.

### B.3 Masked Diffusion Language Modeling Objective

Under the substitution parameterization of the absorbing-state, the negative evidence lower bound in continuous-time reduces to a weighted reconstruction loss over masked positions (Sahoo et al., 2024). For a general retention schedule, the positive weight on negative log-probability is $-\alpha_t'/(1 - \alpha_t)$. Substituting $\alpha_t = 1 - t$ yields

$$\mathcal{L}_{\text{MDLM}}(\theta) = -\mathbb{E}_{t\sim\mathcal{U}(0,1), x_0\sim p_{\text{data}}(x_0)} \mathbb{E}_{x_t\sim q(x_t|x_0)} \left[\frac{1}{t}\sum_{i=1}^{L} \mathbf{1}\big[x_t^i = \mathbf{M}\big] \log p_\theta\big(x_0^i \mid x_t\big)\right]. \tag{7}$$

Only masked positions contribute to the loss. The predictor uses the same-position clean token as its target, with bidirectional access to visible tokens. Time is implicit in the corruption; the notation does not require explicit time conditioning. Numerical truncation near $t = 0$ and token normalization are specified with the training configuration.

### B.4 Block-Wise Diffusion

Partition the sequence into $K = \lceil L/b \rceil$ contiguous blocks of maximum size $b$, denoted by $x_0^{[1]}, \ldots, x_0^{[K]}$. Block diffusion factorizes the sequence distribution as

$$p_\theta(x_0) = \prod_{k=1}^{K} p_\theta\left(x_0^{[k]} \mid x_0^{[<k]}\right). \tag{8}$$

Each conditional is modeled by the same masking and denoising construction, with the preceding blocks supplied as fixed context. Tokens within the current block interact bidirectionally, while future blocks remain inaccessible. Training uses clean preceding blocks; inference uses completed generated blocks. This combines autoregressive ordering across blocks with diffusion within blocks (Arriola et al., 2025).

### B.5 Conditional Generation Across Modalities

For a task with observed context $c$, replace $p_\theta(x_0^i \mid x_t)$ with $p_\theta(x_0^i \mid x_t, c)$ and draw $(x_0, c)$ from the paired data distribution. The context contains the supplied modality and task instruction, while $x_0$ denotes the target modality. For folding, the context contains an amino-acid sequence and the target contains backbone tokens; inverse folding exchanges these roles. Text understanding and text-conditioned sequence generation follow the same construction. In conditional answer-only training, the context stays uncorrupted and only target positions enter the masking process and loss. Full-sequence supervision is a separate training configuration.

# C SFT DATASET DETAILS

This appendix presents the task templates and corpus used for supervised fine-tuning (SFT).

## C.1 TASK TEMPLATES

Table 6 shows examples of the six directed tasks over sequences ($S$), backbones ($B$), and text ($T$).

Table 6: SFT examples and data sources. Italicized placeholders replace biological token spans; {*motif*} denotes a segment within the generated sequence. Modality and motif special tokens are omitted for clarity.

| Task | Source | Prompt | CoT | Answer |
|---|---|---|---|---|
| S2B | DPLM2; UC40 | Generate a plausible backbone structure for the sequence below: {*sequence*} | × | {*backbone*} |
| B2S | DPLM2; UC40 | {*backbone*} Use the protein structure above to construct a plausible sequence. | × | {*sequence*} |
| S2T | Swiss-Prot | Examine the protein sequence and indicate which targeting signals (e.g., signal peptide, mitochondrial targeting peptide, nuclear localization signal) are present. The protein is {*sequence*} | ✓ | The protein contains an N-terminal signal peptide; no mitochondrial targeting sequence or nuclear localization signal is predicted. |
| T2S | Swiss-Prot | Formulate an original protein sequence featuring the Photosystem II PsbT protein family. Design a complete, biologically relevant protein that exhibits the core features and actions of this protein family. Embrace the normal structural conformations, molecular dimensions, and biological activities of the Photosystem II PsbT (protein family). | ✓ | The {*motif*} is a single N-terminal hydrophobic helix followed by a short basic C-terminal tail; it could be constructed into the {*sequence*}. |
| B2T | OPI-Struc | {*backbone*} Could you share a detailed analysis of the function attributed to the protein? | × | Involved in oxygen transport from the lung to the various peripheral tissues. |
| T2B | Prot2Text | Generate a protein backbone of approximately 354 residues for a Tryptophanyl-tRNA synthetase with the following function: Catalyzes the attachment of tryptophan to tRNA(Trp). | × | {*backbone*} |

All examples use a shared user–assistant format, with role delimiters omitted. CoT marks whether the source instance includes reasoning; Answer shows the final response after that reasoning span.

## C.2 DATASET STATISTICS

Figure 6 summarizes the 18.5B-token SFT corpus before further dataset sampling during training. Corpus proportions therefore differ from the effective training mixture.

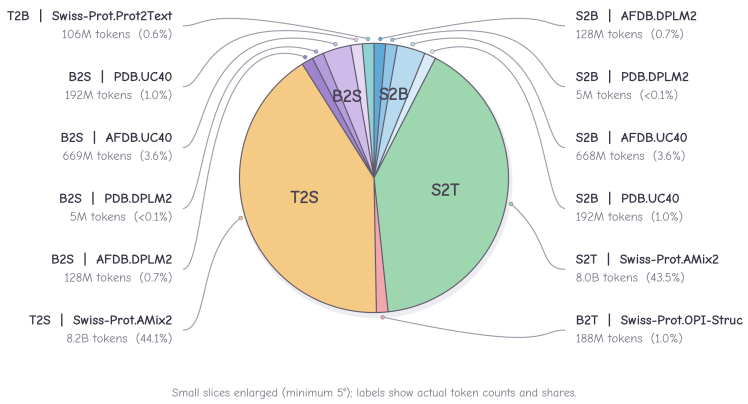



Figure 6: SFT corpus by task and source (*database.collection*). Labels show actual token counts and shares (B: billion; M: million). Small slices are enlarged to at least $5^\circ$ for readability, so areas are not proportional to token counts.

## D Benchmark Evaluation Protocol

**Folding.** We evaluate 632 monomeric targets: 183 from CAMEO 2022 and 449 from the PDB date split. Predictions and native structures are mapped to the reference sequence, with native extensions cropped and missing atoms masked. We report TMca, TMbb, and RMSDbb, averaged per protein within each split. TMca uses C$\alpha$ coordinates, whereas TMbb uses backbone N/C$\alpha$/C coordinates; both are computed with TM-align and normalized by the native residue count after filtering to paired-valid residues. RMSDbb is the root mean squared distance over paired-valid backbone atoms after a single Kabsch alignment, rather than TM-align's local RMSD. Higher TM scores and lower RMSDbb indicate better predictions. Paired table entries show mean TMbb/TMca.

**Inverse folding.** We use the same 183 CAMEO 2022 and 449 PDB targets and report amino-acid recovery (AAR) and self-consistency TMbb (scTMbb). For a generated sequence $\hat{s}$ and reference $s$ of equal length $L$, AAR is $100L^{-1}\sum_{j=1}^{L}\mathbf{1}[\hat{s}_j = s_j]$, without post-hoc sequence realignment. Each generated sequence is folded once with ESMFold v1; scTMbb compares this predicted structure with the native backbone using the same atom masks and native-normalized TMbb protocol as folding. Both metrics are averaged per protein, with higher values indicating better performance. Unavailable folding scores are excluded from scTMbb aggregation while retaining AAR and recording metric coverage.

**Sequence understanding.** GeneralQA contains 481 questions: Function & Family (150), Interaction & Binding (152), Location & Modification (153), and Other (26; 19 physicochemical-property and seven structural-composition questions). A Gemini-based judge compares each generated response with the reference answer for biological equivalence, accepting equivalent concise descriptions and rejecting substantive inconsistencies or ambiguity. Each question receives a binary correct/incorrect decision. Accuracy is $100N_{\text{correct}}/N$ within each category; overall accuracy pools all 481 questions rather than averaging category percentages. Judge confidence does not weight the score.

**Structure understanding.** StructureMCQA comprises 3,918 four-option questions from OPI-Struc. We extract the predicted option from the final answer and report exact-choice accuracy, $100N_{\text{correct}}/3918$; unrecognized or conflicting explicitly marked choices count as incorrect. The BioMatrix entry uses a separately marked nearest-option protocol because its outputs often do not follow the requested format: pure option letters are mapped directly, while free-text responses are mapped to the semantically closest option by a Gemini-based judge without access to the gold answer, then compared with the reference locally. This differs from direct-choice extraction; BioMatrix also receives sequence and structure inputs, whereas Progea receives backbone inputs.

**Sequence design.** We evaluate 870 InterPro-based prompts: Family (569), Superfamily (57), Domain (239), and Site (5), preserving repeated benchmark records. Success rate (SR) is the percentage of all 870 prompts yielding an accepted sequence for evaluation, not functional correctness. Quality metrics are averaged over finite per-sequence scores and displayed on a 0–100 scale. Repetition is $\mathrm{Rep}_n = 1 - |U_n|/(L - n + 1)$ for $n \in \{2, 5\}$, where $U_n$ is the set of distinct overlapping $n$-grams in a sequence of length $L$; it is zero when $L \leq n$, and lower is better. Foldability is measured by ESMFold v1 pLDDT, averaged over atoms in the predicted structure; the PDFBench implementation folds only the first 1,024 residues of longer sequences. Functional recovery is $\mathrm{IPRRecovery} = |R \cap G|/|R|$, where $R$ and $G$ are the reference and generated sequences' InterPro annotation sets; an empty reference set yields an unavailable score. Hard sequence novelty is one minus the maximum MMseqs2 alignment identity among retrieved UniProt hits, or one when a successful search returns no hits. Higher pLDDT, IPRRecovery, and novelty are better. The table uses the first stored candidate per prompt, with the marked likelihood-ranked best-of-ten exception for Pinal; failed parses remain in the SR denominator, and metric-specific coverage is retained.