AMix-2: Establishing Protein as a Native Modality in Large Language Models

[1]Shanghai Artificial Intelligence Laboratory
[2]Generative Symbolic Intelligence Lab (GenSI), Tsinghua University
[3]Institute for AI Industry Research (AIR), Tsinghua University

Abstract

We present AMix-2, a protein–text foundation model that establishes protein as a native modality in large language models (LLMs), unifying protein understanding and sequence design within a single foundation model. AMix-2 is built upon two key ideas: (1) a unified protein–text formulation that embeds natural language and protein sequence in a shared token space, enabling one model to perform biological reasoning and conditional design instead of separate downstream task-specialized models; and (2) a block-wise diffusion language modeling backbone that combines causal generation across blocks with bidirectional context and iterative refinement within blocks. This scheme better matches the intrinsic nature of proteins than a strict left-to-right factorization. To evaluate protein foundation models under realistic generalization settings, we further introduce ProteinArena, a comprehensive benchmark with time-aware and homology-aware protocols across various understanding and design tasks, and with baselines covering classical bioinformatics tools, protein-specialized models and LLMs. On ProteinArena, AMix-2 outperforms frontier LLMs and demonstrates competitive performance to task-specific protein models. Controlled experiments further show that the diffusion-based paradigm generally surpasses its autoregressive counterpart, highlighting the advantage of flexible generation order for protein sequences. We release both AMix-2 and ProteinArena to facilitate open research in protein foundation models.

Correspondence: zhouhao@air.tsinghua.edu.cn

Project Page: `https://amix-bio.github.io/AMix-2/`

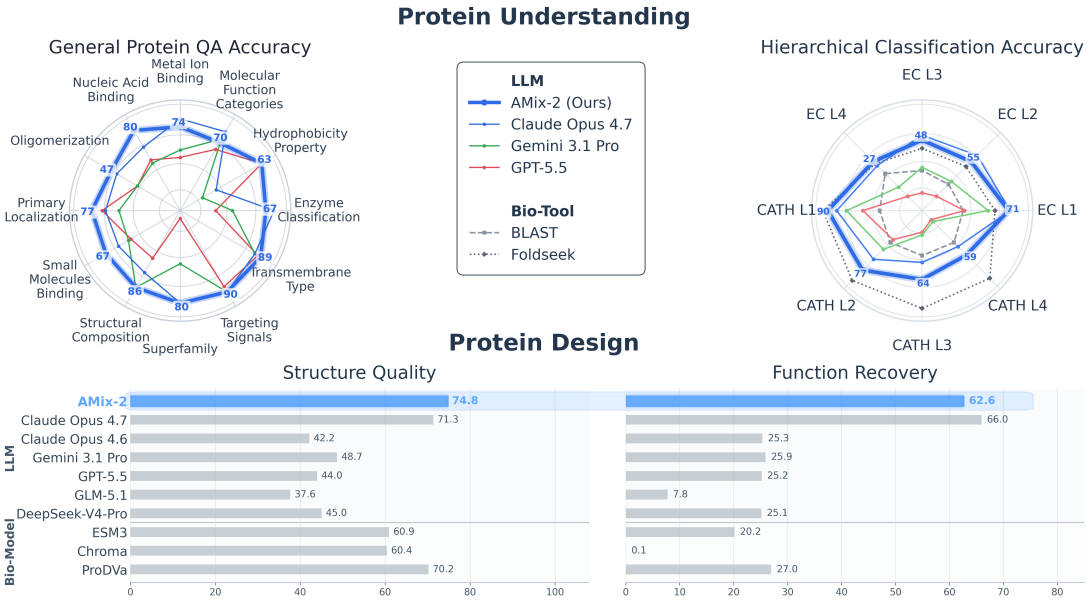

Figure 1 Performance overview of AMix-2. AMix-2 outperforms frontier LLMs and demonstrates competitive performance to task-specific protein models on various protein understanding and design tasks.

Contents

1 Introduction

One of the next major frontiers for large language models (LLMs) is scientific discovery, with life science emerging as a particularly important domain. Among biological entities, proteins are especially central: they carry out most cellular functions and are key objects for both scientific study and bioengineering [1–5]. A model that can directly read, reason about, and even generate proteins would therefore mark a meaningful step toward more capable scientific foundation models.

Current progress towards this goal has followed two complementary paths: (1) Protein-specialized foundation models [6–10] and bioinformatics tools [11–13] have driven major advances in representation learning, structure prediction, sequence generation, and similarity-based search, but remain largely **specialized and isolated**: limited to the protein modality, with interfaces specifically designed for certain fixed-format objectives such as embedding extraction or masked prediction, rather than open-ended language instruction. In practice, protein understanding and sequence design are often handled by different models or by separate task-specific fine-tuning pipelines [14]. (2) Frontier LLM systems [15–20], on the other hand, are **general and externally scaffolded**: they offer flexible natural-language interfaces and can coordinate biological workflows with harnessing. Equipped with retrieval, database access, and other tool use, such systems have shown strong promise for biological problem solving [21–23]. However, in many cases their protein capabilities are mediated primarily through external resources, rather than on protein knowledge learned natively within the model itself. In this work, we use **native protein capability** to refer to the ability to understand and generate proteins directly from protein sequence inputs and instructions, without relying on tools or complex pipelines.

We present AMix-2, a protein–text foundation model that treats proteins as a native modality within a unified language-modeling framework, combining the protein-modeling capacity of specialist models with the instruction-following flexibility of large language models. In Section 2, AMix-2 formulates protein understanding and functional sequence design as protein–text conditional generation in a shared token space: with natural-language instructions specifying the task, protein sequences serving as inputs or outputs, AMix-2 reasons about its functional context. This unified formulation allows a single model to perform diverse protein sequence understanding and design tasks through various instructions. Section 3 describes how AMix-2 uses a block-wise diffusion language modeling backbone. This choice is motivated by a mismatch between protein tasks and standard left-to-right generation. Protein sequences exhibit non-local dependencies [24–26], and many practical tasks involve partial infilling, local editing and iterative refinement rather than purely autoregressive continuation [27]. Block-wise diffusion addresses this by combining causal generation across blocks with bidirectional denoising within each block, supporting both global consistency and fine-grained controllability [28, 29]. Under matched training data, our diffusion backbone outperforms its autoregressive counterpart with large improvements on protein design (Table 1).

To establish a rigorous, shared testbed for assessing native protein capabilities, we introduce ProteinArena in Section 4, a benchmark that follows standard **time-aware** and **homology-aware** evaluation protocols across protein question answering, hierarchical enzyme commission number (EC) and structural fold (CATH) classification, and function-conditioned de novo design [30–33]. These protocols are important for assessing realistic generalization across model families. Despite the fact that for frontier LLMs with broad and largely opaque pretraining corpora, temporal and homology exposure cannot be completely ruled out as the other models, ProteinArena still provides an informative evaluation ground to assess them. By comparing frontier LLMs, protein language models, and specialist bioinformatics tools under the same setting, ProteinArena benchmarks these methods in the broader landscape of protein modeling.

On ProteinArena, AMix-2 generally outperforms non-controlled general-purpose LLMs and shows strong competitiveness with protein-specialized language models and dedicated bioinformatics tools under strict splits (Section 5). It achieves state-of-the-art accuracy in General Protein QA (65.70%), generalizes well to low-homology data regimes in EC and CATH classification, while simultaneously yielding high structural plausibility and function recovery rate in protein design. Together, these results support unified protein–text modeling as an effective route to stronger native protein capability. AMix-2 encourages protein knowledge to be internalized within the model and transferred across tasks, rather than fragmenting it into separate task-specific models and engineering pipelines.

We summarize our contributions as follows:

- We introduce AMix-2, a protein–text foundation model with a block-wise diffusion language modeling backbone that supports both global constraints and local refinement.
- We introduce ProteinArena, a benchmark with strict evaluation protocols across a wide range of protein understanding and design tasks, providing a shared testbed for different model families.
- We show that AMix-2 outperforms frontier LLMs, protein language models, and specialist bioinformatics tools on ProteinArena, and that our dLLMs backbone demonstrates its superiority in protein modeling.

2 Unified Protein-Text Modeling

Our goal is to treat proteins as a native modality within a single foundation model. Rather than building separate architectures or task-specific interfaces for protein understanding and sequence design, we cast both as instruction-following tasks over mixed text and protein sequences. This section presents the unified modeling interface and the pipeline used to construct large-scale training data under this formulation.

2.1 Task Formulation

AMix-2 establishes a single discrete vocabulary $\mathcal{V}$ spanning both natural language and protein amino acids, enabling heterogeneous modalities to be processed within a shared representation space. Concretely, text is tokenized with a standard subword tokenizer and protein sequences are tokenized at the residue level. Each example is serialized as a single sequence that interleaves text tokens x_{text} and protein tokens x_{prot}, so that both modalities are processed by one model under a unified instruction-following objective.

The two main task families differ only in the direction of generation. For **protein understanding**, the model reads a text instruction and a protein sequence, then generates a text answer:

$$p\big(x^{\text{ans}}_{\text{text}} \mid x^{\text{inst}}_{\text{text}},\, x_{\text{prot}}\big)\,. \tag{1}$$

As for **sequence design**, the model reads a text instruction and generates a protein sequence:

$$p\big(x^{\text{ans}}_{\text{prot}} \mid x^{\text{inst}}_{\text{text}}\big)\,. \tag{2}$$

Protein understanding refers to tasks in which the model reads a sequence and answers a textual query, such as a functional description, class label, localization or other property prediction regarding specific aspects of the protein. Sequence design refers to tasks in which the model takes a textual specification of desired functional or biochemical properties, and is required to translate the specified functional requirements to a novel protein sequence. By expressing both tasks as instruction-following in a shared token space, AMix-2 unifies protein understanding and sequence generation within a single modeling interface. Unlike previous protein generative models trained solely on protein data with fixed conditional inputs such as caption labels, which typically assumes a fixed conditioning direction (e.g. text $\rightarrow$ protein) and relies on modality-specific fusion mechanisms [34], or embedding-only models such as ESM2 [6] designed for representation tasks, this formulation is symmetric: the model can condition on and generate either text or protein tokens, enabling a more general **any-to-any** modeling interface across modalities and tasks.

2.2 Data Construction

To support the unified protein–text formulation, we construct a multimodal corpus that expresses diverse biological knowledge from curated biological databases including UniProtKB [30], UniRef50 [35], InterPro [36] and CARE [31]. Training proceeds in two stages: **continual pre-training** for protein knowledge injection and **post-training** for downstream task alignment.

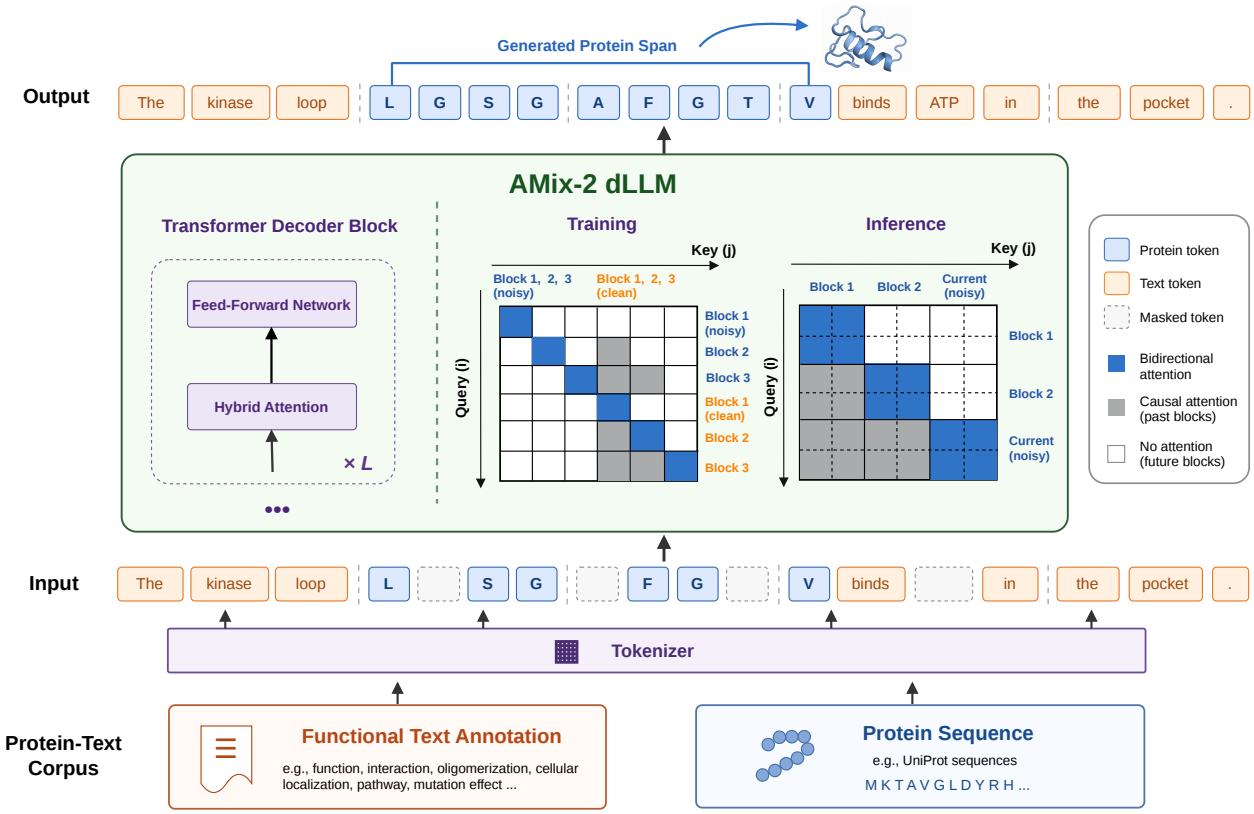

Figure 2 Block-wise diffusion LLM architecture of AMix-2, jointly reasoning over the unified protein–text token space. Protein sequences and functional text annotations are tokenized into a single mixed sequence, where a subset of tokens are masked and predicted according to a noise schedule. The model is implemented as a decoder-only Transformer with hybrid attention: during training, tokens within each diffusion block attend bidirectionally to one another while also attending causally to previous blocks; during inference, previously denoised blocks are treated as fixed context and stored in KV Cache, and the current noisy block is iteratively refined.

Continual Pre-Training We pair the protein sequences of UniRef50 with textual descriptions derived from UniProtKB entries, together with general-domain text to preserve broad language competence. The goal of this stage is to introduce protein sequences into the model's pretraining distribution, so that amino-acid patterns and sequence-level regularities are learned jointly with natural-language semantics rather than treated as out-of-distribution character strings. This stage equips AMix-2 with broad aligned knowledge over proteins and text, providing the foundation for downstream protein understanding and design.

Post-Training We build an instruction-following dataset spanning both protein understanding and sequence design. Protein question answering examples are constructed from Swiss-Prot, a manually reviewed subset of UniProtKB [37]. Function-conditioned *de novo* design cases are derived from InterPro functional keywords and motif descriptions. Hierarchical enzyme commission (EC) and CATH structural fold classification tasks are built from CARE and curated Swiss-Prot annotations, respectively. Each example is formatted as an instruction, the relevant sequence or functional context, and a target output. We additionally attach rationales derived from protein records where appropriate to further boost the reasoning capabilities of AMix-2. These rationales undergo iterative filtering for *factual accuracy*, *logical integrity* and *context consistency*.

Together, the two-stage pipeline transforms heterogeneous biological resources into a unified protein–text training corpus, enabling an all-in-one model of **comprehension**, **classification**, and **sequence design** through instructions alone. Notably, sequences with high similarity to test proteins, or with release dates after the temporal cutoff, have been removed from the labeled training sets for data decontamination.

3 Block-wise Diffusion LLM Architecture

Given the unified protein–text interface, we instantiate AMIX-2 as a **block-wise diffusion large language model** (dLLM). The key motivation is that protein–text joint modeling is not always well served by token-level autoregressive (AR) decoding: while text may be sequential, protein residues naturally exhibit long-range, non-local dependencies, making the standard AR next-token prediction objective less suitable for practical tasks such as motif scaffolding, local editing, and iterative refinement under global constraints. To better accommodate this mixed-modality dependency, AMIX-2 preserves **causal ordering across semantic blocks** for global coherence, while using **bidirectional masked diffusion within each block** to refine tokens jointly. This yields a unified architecture that retains LLM-style controllability over mixed protein–text sequences while better supporting dependency patterns and editing requirements common in protein tasks.

3.1 Mask-based Discrete Diffusion

AMIX-2 models the generative process using mask-absorbing discrete diffusion [27, 38, 39] over the shared discrete vocabulary $\mathcal{V}$ of size K, covering both text sub-words and protein residues. Augmenting $\mathcal{V}$ with a special `[MASK]` state, each clean token is represented by a one-hot vector $\mathbf{x}_0 \in \{e_1, \dots, e_K\} \subset \mathbb{R}^{K+1}$, and let $\mathbf{m} = e_{K+1}$ denote the absorbing `[MASK]` token.

Forward process The forward state at step $t \in [0, 1]$ is governed by the underlying transition matrix

$$Q_t \;=\; \alpha_t\, I \;+\; (1-\alpha_t)\,\mathbf{1}\mathbf{m}^\top \;\in\; \mathbb{R}^{(K+1)\times(K+1)}, \tag{3}$$

where $\alpha_t \in [0, 1]$ is a monotonically decreasing noise schedule. Under this kernel, each token remains unchanged with probability α_t and is absorbed into `[MASK]` with probability $1 - \alpha_t$; once masked, a token stays masked for all subsequent times. The resulting marginal forward distribution is

$$q(\tilde{x}_t \mid x_0) = \text{Categorical}(\tilde{x}_t; \alpha_t x_0 + (1-\alpha_t)\mathbf{m}) \tag{4}$$

Reverse process A neural network p_θ is trained to predict the clean sequence $\mathbf{x}_0$ from the corrupted sequence $\tilde{\mathbf{x}}_t$, parameterizing the reverse denoising process by approximating the reverse posterior transition

$$q(\tilde{x}_s \mid \tilde{x}_t, x_0) = \begin{cases} \text{Categorical}(\tilde{x}_s; x_0) & \text{if } \tilde{x}_t \neq \mathbf{m} \\ \text{Categorical}\left(\tilde{x}_s; \dfrac{\alpha_s - \alpha_t}{1-\alpha_t} x_0 + \dfrac{1-\alpha_s}{1-\alpha_t}\mathbf{m}\right) & \text{if } \tilde{x}_t = \mathbf{m} \end{cases} \tag{5}$$

This mask-absorbing formulation is particularly well suited to protein modeling because it naturally supports partial observation, global conditioning, and iterative refinement.

3.2 Block-wise Factorization

A pure masked-diffusion model denoises the all tokens in parallel. Although this provides maximal flexibility in generation order, it makes variable-length generation less straightforward [40, 41] and leaves the model to resolve unnecessary uncertainty by treating all positions as simultaneously unconstrained. In contrast, AMIX-2 imposes a **block-level** autoregressive factorization on top of the diffusion process [28]. Blocks are generated causally, while tokens within each block are denoised bidirectionally, preserving informative prefix conditioning without sacrificing the flexibility of intra-block arbitrary-order generation.

The interleaved sequence $\mathbf{x}$ is partitioned into L contiguous blocks of block size D,

$$\mathbf{x} = (B_1, B_2, \dots, B_L). \tag{6}$$

AMIX-2 factorizes the joint distribution at the block level:

$$p(\mathbf{x}) = \prod_{k=1}^{L} p(B_k \mid B_{<k}), \tag{7}$$

where $B_{<k} = (B_1, \ldots, B_{k-1})$ denotes the preceding clean or fully denoised blocks. The corrupted block at noise level t is obtained by independently masking each token:

$$q\Big(\widetilde{B}_{k,t} \mid B_k\Big) \;=\; \prod_{i \in B_k} q(\tilde{x}_{i,t} \mid x_{i,0})\,. \tag{8}$$

The key difference from a standard autoregressive model is that each conditional distribution $p(B_k \mid B_{<k})$ is not generated from left to right. Instead, all D tokens of B_k are recovered jointly via masked diffusion conditioned on $B_{<k}$, enabling bidirectional interactions within the block. This hybrid factorization preserves contextual conditioning across blocks while avoiding a rigid within-block left-to-right ordering that is often poorly matched to protein structure and function [42].

3.3 Parallel Training Objective and Hybrid Attention

Training Objective During training, we sample a block index k and a diffusion time t, corrupt the current block to obtain $\widetilde{B}_{k,t}$, and train the model to recover the clean tokens in the masked positions. The resulting objective is

$$\mathcal{L}_{\text{block}}(\theta) = \mathbb{E}_{\mathbf{x},k,t,\widetilde{B}_{k,t}} \left[w(t) \sum_{i \in M_{k,t}} -\log p_\theta\Big(x_{i,0} \mid B_{<k}, \widetilde{B}_{k,t}\Big) \right], \tag{9}$$

where $M_{k,t}$ denotes the masked positions in the target block, and $w(t)$ weights different noise levels. The ELBO can be reduced to a weighted cross-entropy over the masked positions [40, 43, 44].

Compared with the standard AR objective $\mathcal{L}_{\text{AR}} = -\mathbb{E}\sum_i \log p_\theta(x_i \mid \mathbf{x}_{<i})$, the block diffusion objective provides **denser supervision**: at each training step, the model is trained to reconstruct multiple masked positions simultaneously, so gradient signals are aggregated from all $|M_{k,t}|$ targets rather than a single next-token prediction. This is particularly beneficial for protein modeling, where residue identities are often correlated, and the model is required to capture the co-evolutionary constraints across multiple sites [24, 45].

Hybrid Causal-Bidirectional Attention The block factorization in Equation (7) induces a structured attention mask that is the architectural centerpiece of AMix-2, as illustrated in Figure 2. At inference time, blocks are generated sequentially, and the current block is denoised conditioned on previously generated clean blocks. For efficient training, we implement the same dependency structure with a parallel masking scheme. Specifically, we pack all corrupted blocks together with their clean counterparts into a single sequence.

$$S \;=\; \big(\widetilde{B}_{1,t},\; \widetilde{B}_{2,t},\; \ldots,\; \widetilde{B}_{k,t},\; \ldots, B_1,\; B_2,\; , \ldots, B_k,\; \ldots\big). \tag{10}$$

For a query position $u \in \widetilde{B}_{k,t}$, the attention mask is defined so that u may attend to all tokens in the same corrupted block and to all clean prefix blocks:

$$A_{uv} = 1 \iff \big(\underbrace{v \in \widetilde{B}_{k,t}}_{\text{bidirectional}}\big) \text{ or } \big(\underbrace{v \in B_{<k}}_{\text{causal}}\big). \tag{11}$$

Thus, each noisy block attends only to itself and the appropriate clean prefix that would be available at inference time, while all blockwise denoising losses can still be computed and optimized in parallel during a single forward pass. This makes the model well suited to protein–text tasks that require autoregressive conditioning on informative upstream content while refining protein tokens within a coherent local region.

4 ProteinArena: A Time-aware and Homology-aware Benchmark

Reliable evaluation of native protein capability requires more than random train–test splits. Protein datasets are highly redundant, and models can achieve deceptively strong performance by exploiting close homologs rather than learning transferable biochemical regularities [46–48]. In addition, for models trained on broad scientific or web-scale corpora, temporal leakage can confound evaluation if test proteins were already publicly available during training. To address these issues, we introduce ProteinArena, a benchmark designed for fair evaluation of native protein understanding and functional sequence design.

4.1 Benchmark Principles and Split Protocols

ProteinArena is built around two principles: **time awareness** and **homology awareness**, partitioning training and evaluation data based on both temporal cutoffs and sequence identity thresholds.

By retaining only proteins with a Swiss-Prot first public date on or after **January 1, 2025**, ProteinArena removes the risk of temporal contamination for AMix-2, as well as bioinformatics tools and protein language model baselines, which were all trained on sequences prior to this cut-off date. Moreover, our benchmark explicitly controls sequence similarity between training and evaluation sets. Under the primary test setting, only those proteins that exhibit less than 30% sequence identity to any sequence released on or before **December 31, 2024** are included. This low-homology regime is intended to measure genuine generalization rather than near-neighbor retrieval. We additionally report higher-homology ranges as supplementary analysis in Appendix D, but focus our main interest on the more challenging low-homology setting.

As ProteinArena is designed to evaluate the native capabilities of models, assessment is conducted without external retrieval, database lookup, or workflow orchestration, thus keeping the focus of evaluation on model-internal protein competence.

4.2 Protein Understanding

The understanding track evaluates newly released, low-homology proteins using a variety of tasks spanning both open-form question answering and fine-grained classification, measuring fundamental aspects of protein understanding such as function, interaction, localization and structure.

General Protein QA To evaluate open-ended protein understanding in a natural-language setting, we construct a General Protein QA subset from reviewed Swiss-Prot entries released under the benchmark protocol. The objective of this task is to measure whether a model can infer biologically meaningful properties directly from sequence and express them through text, rather than only producing fixed labels. As illustrated in Figure 4, this subset contains 481 samples across 16 representative categories, including cleavage sites, enzyme classification, functional domains, hydrophobicity property, molecular function categories, metal ion binding, nucleic acid binding, oligomerization, post-translational modifications, primary localization, protein family, small molecules binding, structural composition, superfamily, targeting signals, and transmembrane type. Each sample pairs a protein entry with a paraphrased natural-language question, providing the ability to probe whether a model can infer biologically meaningful properties directly from the sequences given. Instruction instances for each category are detailed in Appendix B.1.

Hierarchical Classification To facilitate direct comparison with protein-specialized models and classical bioinformatics tools, ProteinArena further includes hierarchical classification tasks of EC and CATH numbers. These tasks require the evaluated model to predict and assign labels across all four levels of the corresponding ontology, from Level 1 (`x.-.-.-`) to Level 4 (`x.x.x.x`), thereby testing progressively fine-grained functional and structural discrimination capabilities. Compared with coarse binary or family-level prediction, the four levels of EC and CATH classification provide a more stringent measure of biological understanding, evaluating whether a model can move beyond superficial sequence similarity and recover functionally and structurally meaningful information under strict generalization constraints.

4.3 Protein Design

The design track evaluates function-conditioned de novo sequence generation. Following recent functional protein design benchmark [33], we construct prompts from expert-reviewed InterPro functional keywords recorded in Swiss-Prot annotations under the time-aware protocol. Given these functional constraints, the model conducts *de novo* design, i.e., generating a protein sequence from scratch.

This task aims to evaluate the model's instruction-following capability in translating functional constraints into valid protein sequences. Therefore, our evaluation framework fundamentally assesses two complementary criteria: (1) *functional alignment*, i.e., whether the generated sequence realizes the requested function, and (2)

biophysical plausibility, i.e., whether it resembles viable natural proteins in structural quality and sequence-level characteristics. We additionally track uniqueness and novelty to distinguish meaningful design from degenerate generation or simple memorization. Evaluation metrics are further detailed in Section 5.2.

By jointly covering comprehensive understanding and function-conditioned design, ProteinArena provides a unified benchmark for native protein capabilities, enabling direct comparison across frontier LLMs, protein language models, and specialist bioinformatics tools under a common, leakage-aware evaluation protocol.

5 Experiments

5.1 Training Setup

We train two architectural variants of AMix-2 —the block-wise diffusion model (dLLM) and the autoregressive model (AR)—following a standardized two-stage large language model training paradigm. Both models are initialized from the `Qwen3-4B-Base` [20] backbone and undergo initial continual pre-training on large-scale protein–text corpora, followed by post-training on a curated mixture of protein understanding and design instruction data. A fixed evaluation split of 1,000 packed samples is reserved to monitor convergence. Across both architectures, training is conducted in mixed-precision (BF16) using the AdamW optimizer with weight decay of 0.01, $\beta_2 = 0.995$, and gradient norm clipping at 1.0. To enhance stability and generalization, an Exponential Moving Average (EMA) is applied to model weights with a decay rate of 0.9995. Learning rates follow a cosine decay schedule with a peak of 5×10^{-5}.

AMix-2 dLLM Serving as our flagship architecture, this model utilizes a block diffusion language modeling objective with a linear noise schedule and a block size $D = 32$. For large-scale distributed training, we employ a global batch size of 768, pack samples to a length of 4096, and set the warmup ratio to 0.05. Notably, for the continual pre-training phase, we employ a block size warming-up schedule for smoother gradient norms and more stabilized training, while post-training utilizes document-aware packing with sample boundaries aligned to the block size D. This diffusion-based approach allows for bidirectional refinement within each block, facilitating the global and local conditioning required for complex protein tasks.

AMix-2 AR As a comparative baseline, the autoregressive variant is trained using a standard next-token prediction objective. To ensure a rigorous comparison, we maintain the same core hyperparameter configurations and training data as AMix-2 dLLM, with the exception of using a global batch size of 256, a packing length of 8192, and a warmup ratio of 0.1.

5.2 Baselines and Metrics

We compare the dLLM and autoregressive backbone variants of our AMix-2 model against a suite of baselines: (1) **Frontier LLMs** including Claude Opus 4.7 [15], Claude Opus 4.6 [15], Gemini 3.1 Pro [16], GPT-5.5 [17], DeepSeek-V4-Pro [18], GLM-5.1 [19] and Qwen3.5-27B [20]. (2) For protein understanding, we compare against **protein-specific representation models**, including ESM2 [6], ESM3 [10], ProTrek [9] and SaProt [8], alongside **retrieval-based bioinformatics tools** such as BLAST [11] and Foldseek [13]. (3) For sequence design, we consider baselines that are compatible with our conditioning setup, including both **sequence-based** such as ProteinDT [49] and **structure-based** such as Chroma [50], together with the multimodal ESM3, as detailed in Appendix C.1. Notably, structure-based models such as RFdiffusion [4] are not included because they do not support the programmable functional-specification, and therefore are not directly comparable.

Crucially, while frontier LLMs typically do not restrict their training data regarding specific protein sequences, all included protein-specific models and bioinformatics tools were verified to have training and reference databases cutoff dates prior to **December 31, 2024**. This ensures that no overlap exists between their supervised training samples and the test in ProteinArena, consistent with the setting used for AMix-2.

Specifically, for the **General Protein QA** task, we employ LLM-as-a-judge [51] using Gemini 3 Flash, evaluating whether the model-generated answers are semantically identical to the ground-truth protein functions. Traditional protein language models and specialist bioinformatics tools are omitted from this task, as their

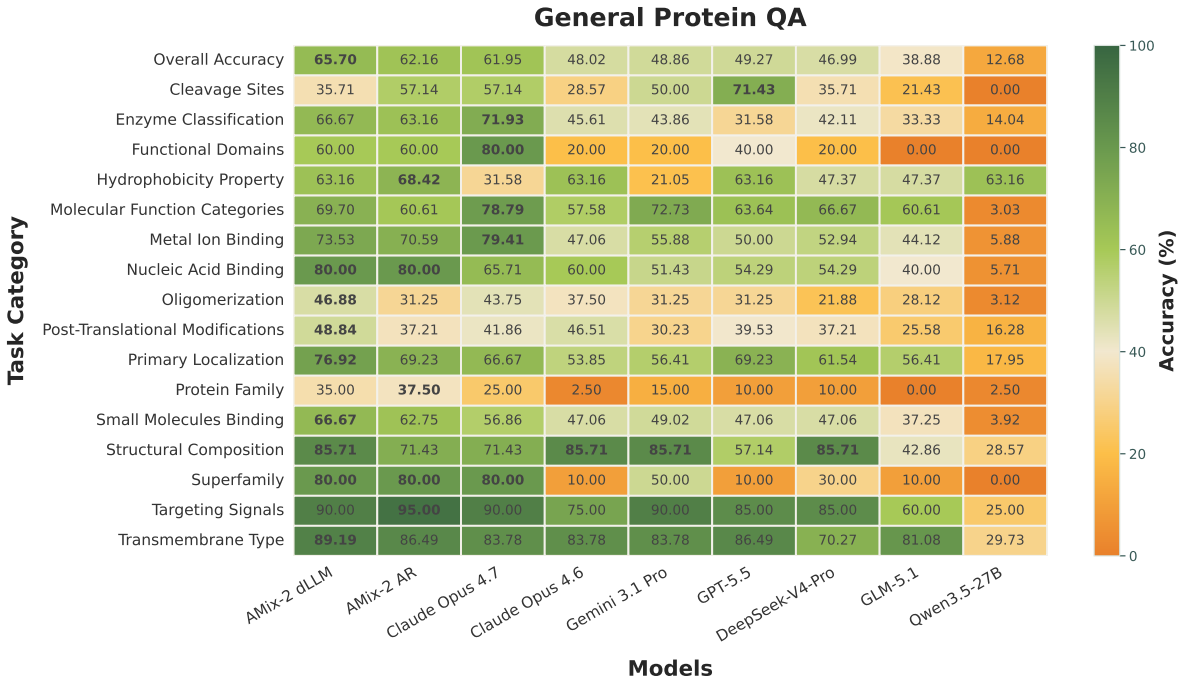

Figure 3 General Protein QA performance across AMix-2 and frontier large language models.

architectures are designed for representation learning or classification rather than free-form answer generation in a conversational QA-based setting. For the **Functional De Novo Design** task, following the evaluation setup of Kuang et al. [33], we assess the generated sequences using a comprehensive suite of structural and functional metrics. Structural plausibility is measured via the ESMFold [6] predicted Local Distance Difference Test (pLDDT). To evaluate functional adherence, we compute Function Recovery (InterPro Recovery, IPR) by comparing the functional labels—obtained by scanning the generated sequences with InterProScan—against the ground-truth annotations. Furthermore, we quantify the sequence-level and structure-level novelty and diversity of the generated proteins utilizing MMseqs2 [12] and Foldseek, respectively. For the hierarchical classification tasks of **EC Prediction** and **CATH Prediction**, sequences are first grouped by identity, and performance is then measured by calculating accuracy metrics according to the precise correctness of predicted labels across the four hierarchical levels, for example "5.6.2.4". Details of metrics are left to Appendix C.2.

5.3 Results and Analysis

General Protein QA We evaluate the performance of our proposed model, AMix-2, on ProteinArena across 16 distinct protein functionality categories, comparing it against state-of-the-art frontier models. To ensure rigorous assessment, all proteins sharing a sequence identity greater than 30% with ProteinArena were strictly excluded from the training of AMix-2. As illustrated in Figure 3, the AMix-2 dLLM variant achieves the highest overall accuracy of 65.70%, followed by AMix-2 AR at 62.16%. Both variants surpass the leading frontier model Claude Opus 4.7 at 61.95%, and substantially outperform other gigantic general-purpose LLM baselines such as GPT-5.5 and Gemini 3.1 Pro. While particular models exhibit higher accuracy at tasks like Cleavage Sites and Functional Domains, both variants of AMix-2 remain highly competitive, outperforming the majority of the frontier cohort. Detailed task-specific analysis yields the following insights:

- **Taxonomic and Evolutionary Classification:** In Protein Family and Superfamily classification, AMix-2 achieves an accuracy of 35% and 80% respectively, whereas leading models like Claude Opus 4.6, Gemini 3.1 Pro and GPT-5.5 struggle significantly. This highlights AMix-2's ability to map sequences to evolutionary clusters even in low-homology regimes.

- **Structural and Localization Properties:** Our AMix-2 models show exceptional ability in predicting structural and localization properties, achieving near-ceiling performance on the tasks of Targeting Signals, Transmembrane Type, and Structural Composition.
- **Interaction and Binding:** AMix-2 leads the benchmark in tasks such as Small Molecules Binding, Oligomerization, and Nucleic Acid Binding, while remaining competitive on Metal Ion Binding, demonstrating its ability to recognize protein interaction and binding types.

Functional De Novo Design Table 1 shows AMix-2 dLLM effectively integrates the strengths of protein-native modeling with the semantic reasoning capabilities of LLM foundations. It achieves superior generation quality, yielding both high pLDDT and functional recovery rate (IPR), significantly exceeding the vast majority of evaluated frontier LLMs in protein quality and surpassing specialized bio-models in satisfying text-specified functional constraints. We summarize our findings from the design results below:

- **Design Failure in General LLMs:** Most off-the-shelf frontier models exhibit substantial weaknesses in protein design, typically suffering from pathological n-gram sequence repetition and structural validity. For instance, Qwen3.5 yields a Rep5 score of 63.24, while most models produce poor structural quality with pLDDT scores below 50. A notable exception is Claude Opus 4.7, showing highly competitive performance for protein sequence modeling compared to its peers.
- **Usability Bottlenecks of Specialist Models:** Protein-specialized models generally produce more structurally plausible sequences, but their practical utility is limited by rigid conditioning mechanisms. Unlike natural-language interface for LLMs, these models require task-specific input adaptations tied to their training formats–ranging from raw InterPro IDs and mapped class labels to exact UniProt text templates, greatly restricting their flexibility (exemplified in Table 9). Notably, CFPGen yields valid samples for only 36.7% of the test cases. To enable a better controlled apples-to-apples comparison, we report results on the subset shared across all baselines in Appendix D.
- **Advantage of the dLLM Architecture:** Our results indicate that dLLM architecture is inherently better suited for protein design than traditional autoregressive counterparts. Compared with our standard AMix-2 AR variant, the dLLM architecture provides a substantial boost in sequence quality and effectively mitigates the degeneration and overfitting behavior observed in AR models for such data-constrained scenarios, yielding higher diversity and substantially improved uniqueness.

Hierarchical Classification The performance of our AMix-2 model on the hierarchical classification of Enzyme Commission (EC) numbers and CATH structural domains is detailed in Table 2, demonstrating AMix-2's capability of capturing structural and functional features in low-homology regimes where traditional alignment tools like BLAST struggle. Notably, AMix-2 also surpasses frontier LLMs by a greater margin in high-homology regimes, as shown in Appendix D.

- **EC Prediction:** On the task of EC Prediction, AMix-2 AR and AMix-2 dLLM achieves leading Level-4 accuracies of 30.30% and 27.27%, while the other competitive specialized model ESM2 hits 28.28% and Claude Opus 4.7 yields 25.25%. AMix-2 significantly outperforms other general-purpose LLMs such as Gemini 3.1 Pro and DeepSeek-V4-Pro. This highlights how AMix-2's hierarchical decomposition training strategy for EC numbers allows it to effectively bridge the gap between protein functions and fine-grained enzyme classifications through linguistic reasoning.
- **CATH Prediction:** Within the general-purpose LLM category, AMix-2 dLLM sets a new baseline for structural fold classification from sequence alone, achieving the best performance across all CATH levels, with AMix-2 AR consistently ranking among the strongest LLM baselines as well. Both variants clearly outperform other frontier LLM baselines, highlighting the benefit of unified protein–language modeling for structural inference. While specialist bio-models and structure-aware tools still remain stronger overall, AMix-2 represents a meaningful step toward high-fidelity fold classification without requiring explicit 3D coordinate inputs, unlike methods such as Foldseek, SaProt, and ProTrek.

Category	Model	Repetition			Quality		Seq Distribution	
		Rep	Rep2	Rep5	pLDDT	IPR (%)	Novelty	Unique (%)
Reference	Natural	2.23	44.05	0.48	75.75	100.00	4.82	99.43
Bio-Model	CFPGen	13.52	60.15	15.77	74.45	33.08	50.26	100.00
	Chroma	2.68	51.92	0.47	60.38	0.05	59.56	100.00
	ESM3	28.39	69.46	21.64	60.56	19.20	73.70	100.00
	Pinal	14.28	52.11	4.68	68.49	36.44	49.76	99.98
	ProDVa	3.91	25.29	6.69	70.22	26.96	22.86	82.53
	ProteinDT	5.09	52.25	2.16	39.53	8.98	61.05	99.90
LLM	Qwen3.5-27B	15.98	76.95	63.24	33.47	0.47	77.93	100.00
	GLM-5.1	1.55	47.34	4.32	37.64	7.75	54.46	100.00
	DeepSeek-V4-Pro	6.92	50.32	8.14	45.03	25.14	51.25	100.00
	GPT-5.5	2.99	53.19	3.86	43.95	25.21	44.21	100.00
	Gemini 3.1 Pro	6.38	45.23	7.61	48.91	25.77	46.17	99.06
	Claude Opus 4.6	2.24	48.44	1.86	42.17	25.32	45.91	97.59
	Claude Opus 4.7	2.43	44.65	0.65	71.33	**65.95**	36.54	99.86
	AMix-2 AR	6.99	48.35	6.30	64.02	39.07	11.83	49.68
	AMix-2 dLLM	3.47	45.22	1.23	**74.75**	62.61	30.57	95.73

Table 1 Comparison of various models on generation quality, novelty, and sequence uniqueness metrics.

- **Domain-Specific Advantages vs. LLM Limitations Analysis:** The stronger performance of protein-specialized methods is expected, as these tasks are closely tied to sequence patterns that encode protein function and structure. Classical bioinformatics tools directly leverage sequence similarity and homology-based transfer, and protein language models benefit from pretraining on large protein datasets that capture domain-specific biochemical representations. General-purpose LLMs, in contrast, are primarily optimized for natural language and typically treat protein sequences as peripheral inputs rather than a native modeling target. By internalizing protein knowledge within a shared foundation model, AMix-2 narrows this gap: it retains the flexible language interface of LLMs while learning protein-aware representations that better support fine-grained functional and structural classification.

6 Related Work

Protein-Specialized Foundation Models. Specialized protein language models (PLMs) have emerged as a powerful foundation for modeling protein sequences. For protein understanding, masked language models (MLMs) such as the ESM family [6, 24, 52] learn informative representations through bidirectional context modeling, providing strong priors for downstream prediction tasks. However, as MLMs are optimized for reconstruction rather than direct generation, they are not naturally suited for *de novo* sequence design. For protein generation, autoregressive (AR) models like ProGen [53–55] support sequence design but often underperform in representation learning. The diffusion-based DPLM family [27, 56–58] has sought to bridge this gap, noting that the strict left-to-right factorization of AR models may hinder the modeling of complex global residue interactions. Despite these advances, existing protein foundation models still largely rely on predefined conditioning schemes, making them suboptimal for flexible instruction-driven reasoning. Consequently, protein understanding and sequence design are often addressed using separate models or task-specific fine-tuning pipelines. This limitation directly motivates AMix-2, which unifies biological reasoning and conditional sequence design within a single foundation model equipped with open-ended natural language interfaces.

Large Language Models (LLMs) for Biology. Recent efforts have sought to overcome the rigid interfaces of specialized protein foundation models by integrating biological modalities into general-purpose LLMs. One line of work focuses on **modality adaptation**, where early approaches primarily projected textual annotations

Category	Model	EC Prediction				CATH Prediction			
		L1	L2	L3	L4	L1	L2	L3	L4
Bio-Tool	BLAST	36.36	31.31	29.29	21.21	44.07	44.07	44.07	44.07
	Foldseek	60.61	49.49	43.43	26.26	94.07	88.98	88.14	86.44
Bio-Model	ESM2	70.71	57.58	49.49	28.28	91.53	88.14	85.59	81.36
	ESM3	59.60	44.44	39.39	18.18	93.22	85.59	78.81	73.73
	ProTrek	62.63	45.45	44.44	24.24	94.92	88.14	87.29	81.36
	SaProt	69.70	50.51	48.48	24.24	92.37	86.44	83.90	78.81
LLM	Qwen3.5-27B	12.12	2.02	0.00	0.00	28.81	16.10	4.24	0.85
	GLM-5.1	27.27	5.05	4.04	3.03	56.78	24.58	6.78	2.54
	DeepSeek-V4-Pro	45.45	19.19	18.18	7.07	66.95	44.07	27.97	20.34
	GPT-5.5	37.37	18.18	14.14	9.09	59.32	40.68	22.88	14.41
	Gemini 3.1 Pro	55.56	34.34	31.31	14.14	73.73	52.54	25.42	16.95
	Claude Opus 4.6	42.42	21.21	18.18	9.09	64.41	38.14	16.95	14.41
	Claude Opus 4.7	72.73	**61.62**	51.52	25.25	82.20	64.41	50.00	49.15
	AMix-2 AR	**76.77**	58.59	**54.55**	**30.30**	86.44	69.49	60.17	56.78
	AMix-2 dLLM	70.71	54.55	48.48	27.27	**89.83**	**77.12**	**64.41**	**59.32**

Table 2 Level-wise EC and CATH prediction accuracy on proteins with $< 30\%$ sequence identity.

as conditioning input into specialized PLMs for controllable generation or retrieval-augmented prediction [49, 59, 60], leaving text encoder the major information bottleneck. More recent works instead move toward the other way round, projecting proteins, nucleic acids, and other biological signals into the token space of established LLMs to enable biological understanding and instruction following [23, 61]. Another direction studies **agentic scientific systems**, where frontier LLMs are augmented with external biological harnesses. Systems such as Claude for Life Sciences [62], GPT-Rosalind [63] and other agentic frameworks [21, 22] have shown great potential in multi-step biological problem-solving via tool calling. While such orchestration is immensely powerful, its potential is naturally bounded by the native biological capabilities of the foundation model. Motivated by this, our work targets the gap between general LLMs and specialized PLMs.

7 Conclusion

In this work, we argue that biology needs models with **native protein capability**: the ability to process proteins directly from sequence, rather than relying mainly on external tools or retrieved knowledge. Our analysis shows that strong general language ability does not automatically translate into strong protein competence: under a shared evaluation protocol, frontier LLMs still lag behind protein-specialized models and bioinformatics tools on sequence-grounded biological understanding and design.

To address this gap, we present AMix-2, a unified protein–text foundation model that treats proteins as a native modality within a diffusion language modeling framework. By casting both protein understanding and sequence design as instruction-following tasks over a shared token space, AMix-2 supports diverse tasks instead of separate task-specific pipelines, while encouraging protein knowledge to be internalized and transferred across tasks. We also introduce ProteinArena, a benchmark for evaluating native protein capability under realistic generalization settings. ProteinArena adopts time-aware and homology-aware protocols across both understanding and design tasks, enabling direct comparison among baselines under a common evaluation setup. On ProteinArena, AMix-2 generally outperforms frontier LLMs and shows strong competitiveness with protein-specialized models and bioinformatics tools.

Taken together, these results suggest that unified protein–text modeling is a promising route toward more general scientific foundation models, reducing the fragmentation of task-specific workflows. We hope that releasing AMix-2 and ProteinArena will support future research on scientific discovery.

Contributions

Project Lead

Lihao Wang[1,2]

Model

Keyue Qiu[1,2,3], Yixin Wu[1,2,5], Zihan Zhou[1,7], Changze Lv[1,5], Lihao Wang[1,2]

Evaluation

Yawen Ouyang[1,2], Jixiang Yu[6], Dongyu Xue[1,2]

Other Contributors

Yuxuan Song[2,3], Xinbo Zhang[1,2], Hao Wang[2,3], Jiangtao Feng[1,2,3], Zhiqiang Gao[1], Lijun Wu[1], Xiaoqing Zheng[5], Ka-Chun Wong[6], Lei Bai[1], Ya-Qin Zhang[3], Wei-Ying Ma[3,6], Dahua Lin[1], Bowen Zhou[1,4]

Co-first Authors

Keyue Qiu[1,2,3], Yixin Wu[1,2,5]

Correspondence

Hao Zhou[1,2,3]

Affiliation

[1]Shanghai Artificial Intelligence Laboratory

[2]Generative Symbolic Intelligence Lab (GenSI), Tsinghua University

[3]Institute for AI Industry Research (AIR), Tsinghua University

[4]Tsinghua University

[5]Fudan University

[6]City University of Hong Kong

[7]Chinese University of Hong Kong, Shenzhen

Acknowledgments

We thank DeepLink Team for their infrastructure support. This work is supported by Shanghai Artificial Intelligence Laboratory.

References

[1] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with alphafold. **nature**, 596(7873):583–589, 2021.

[2] Minkyung Baek, Frank DiMaio, Ivan Anishchenko, Justas Dauparas, Sergey Ovchinnikov, Gyu Rie Lee, Jue Wang, Qian Cong, Lisa N. Kinch, R. Dustin Schaeffer, Claudia Millán, Hahnbeom Park, Carson Adams, Caleb R. Glassman, Andy DeGiovanni, Jose H. Pereira, Andria V. Rodrigues, Alberdina A. van Dijk, Ana C. Ebrecht, Diederik J. Opperman, Theo Sagmeister, Christoph Buhlheller, Tea Pavkov-Keller, Manoj K. Rathinaswamy, Udit Dalwadi, Calvin K. Yip, John E. Burke, K. Christopher Garcia, Nick V. Grishin, Paul D. Adams, Randy J. Read, and David Baker. Accurate prediction of protein structures and interactions using a three-track neural network. **Science**, 373(6557):871–876, 2021. doi: 10.1126/science.abj8754. URL `https://www.science.org/doi/abs/10.1126/science.abj8754`.

[3] 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.

[4] 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.

[5] 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**, pages 1–3, 2024.

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

[7] Ali Madani, Ben Krause, Eric R Greene, Subu Subramanian, Benjamin P Mohr, James M Holton, Jose Luis Olmos, Caiming Xiong, Zachary Z Sun, Richard Socher, et al. Large language models generate functional protein sequences across diverse families. **Nature Biotechnology**, 41(8):1099–1106, 2023.

[8] Jin Su, Chenchen Han, Yuyang Zhou, Junjie Shan, Xibin Zhou, and Fajie Yuan. Saprot: Protein language modeling with structure-aware vocabulary. **bioRxiv**, pages 2023–10, 2023.

[9] Jin Su, Xibin Zhou, Xuting Zhang, and Fajie Yuan. Protrek: Navigating the protein universe through tri-modal contrastive learning. **bioRxiv**, pages 2024–05, 2024.

[10] 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.

[11] Stephen F Altschul, Warren Gish, Webb Miller, Eugene W Myers, and David J Lipman. Basic local alignment search tool. **Journal of molecular biology**, 215(3):403–410, 1990.

[12] Martin Steinegger and Johannes Söding. Mmseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. **Nature biotechnology**, 35(11):1026–1028, 2017.

[13] 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.

[14] Zhidian Zhang, Chenxi Ou, Yehlin Cho, Yo Akiyama, and Sergey Ovchinnikov. Artificial intelligence methods for protein folding and design. **Current Opinion in Structural Biology**, 93:103066, 2025. ISSN 0959-440X. doi: https://doi.org/10.1016/j.sbi.2025.103066. URL `https://www.sciencedirect.com/science/article/pii/S0959440X25000843`.

[15] Anthropic. Introducing claude 4. `https://www.anthropic.com/news/claude-4`, 2026.

[16] Google DeepMind. Gemini 3.1 pro. `https://deepmind.google/models/gemini/pro`, 2026.

[17] Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, et al. Openai gpt-5 system card. **arXiv preprint arXiv:2601.03267**, 2025.

[18] DeepSeek AI. Deepseek-v4: Towards highly efficient million-token context intelligence. `https://huggingface.co/deepseek-ai/DeepSeek-V4-Pro`, 2026.

[19] Aohan Zeng, Xin Lv, Zhenyu Hou, Zhengxiao Du, Qinkai Zheng, Bin Chen, Da Yin, Chendi Ge, Chenghua Huang, Chengxing Xie, et al. Glm-5: from vibe coding to agentic engineering. **arXiv preprint arXiv:2602.15763**, 2026.

[20] 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`.

[21] Kexin Huang, Serena Zhang, Hanchen Wang, Yuanhao Qu, Yingzhou Lu, Yusuf Roohani, Ryan Li, Lin Qiu, Junze Zhang, Yin Di, et al. Biomni: A general-purpose biomedical ai agent. **bioRxiv**, pages 2025–05, 2025.

[22] Ruofan Jin, Mingyang Xu, Fei Meng, Guancheng Wan, Qingran Cai, Yize Jiang, Jin Han, Yuanyuan Chen, Wanqing Lu, Mengyang Wang, Zhiqian Lan, Yuxuan Jiang, Junhong Liu, Dongyao Wang, Le Cong, and Zaixi Zhang. Stella: Towards a biomedical world model with self-evolving multimodal agents. **bioRxiv**, 2025. doi: 10.1101/2025.07.01.662467.

[23] Adibvafa Fallahpour, Andrew Magnuson, Purav Gupta, Shihao Ma, Jack Naimer, Arnav Shah, Haonan Duan, Omar Ibrahim, Hani Goodarzi, Chris J. Maddison, and Bo Wang. Bioreason: Incentivizing multimodal biological reasoning within a dna-llm model, 2025. URL `https://arxiv.org/abs/2505.23579`.

[24] Alexander Rives, Joshua Meier, Tom Sercu, Siddharth Goyal, Zeming Lin, Jason Liu, Demi Guo, Myle Ott, C Lawrence Zitnick, Jerry Ma, et al. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. **Proceedings of the National Academy of Sciences**, 118(15):e2016239118, 2021.

[25] Roshan M Rao, Jason Liu, Robert Verkuil, Joshua Meier, John Canny, Pieter Abbeel, Tom Sercu, and Alexander Rives. Msa transformer. In **International Conference on Machine Learning**, pages 8844–8856. PMLR, 2021.

[26] Roshan Rao, Joshua Meier, Tom Sercu, Sergey Ovchinnikov, and Alexander Rives. Transformer protein language models are unsupervised structure learners. In **International Conference on Learning Representations**, 2021. URL `https://openreview.net/forum?id=fylclEqgvgd`.

[27] 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**, 2024.

[28] 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. In **The Thirteenth International Conference on Learning Representations**, 2025. URL `https://arxiv.org/abs/2503.09573`.

[29] Yuxuan Song, Zheng Zhang, Cheng Luo, Pengyang Gao, Fan Xia, Hao Luo, Zheng Li, Yuehang Yang, Hongli Yu, Xingwei Qu, Yuwei Fu, Jing Su, Ge Zhang, Wenhao Huang, Mingxuan Wang, Lin Yan, Xiaoying Jia, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Yonghui Wu, and Hao Zhou. Seed diffusion: A large-scale diffusion language model with high-speed inference, 2025. URL `https://arxiv.org/abs/2508.02193`.

[30] The UniProt Consortium. Uniprot: the universal protein knowledgebase in 2023. **Nucleic Acids Research**, 51 (D1):D523–D531, 01 2023. ISSN 0305-1048. doi: 10.1093/nar/gkac1052. URL `https://doi.org/10.1093/nar/gkac1052`.

[31] Jason Yang, Ariane Mora, Shengchao Liu, Bruce J Wittmann, Anima Anandkumar, Frances H Arnold, and Yisong Yue. Care: a benchmark suite for the classification and retrieval of enzymes. **Advances in Neural Information Processing Systems**, 37:3094–3121, 2024.

[32] Vaishali P Waman, Nicola Bordin, Andy Lau, Shaun Kandathil, Jude Wells, David Miller, Sameer Velankar, David T Jones, Ian Sillitoe, and Christine Orengo. Cath v4. 4: major expansion of cath by experimental and predicted structural data. **Nucleic Acids Research**, 53(D1):D348–D355, 2025.

[33] Jiahao Kuang, Nuowei Liu, Jie Wang, Changzhi Sun, Tao Ji, and Yuanbin Wu. Pdfbench: A benchmark for de novo protein design from function. **arXiv preprint arXiv:2505.20346**, 2025.

[34] Qizhi Pei, Zhimeng Zhou, Kaiyuan Gao, Jinhua Zhu, Yue Wang, Zun Wang, Tao Qin, Lijun Wu, and Rui Yan. Leveraging biomolecule and natural language through multi-modal learning: A survey, 2025. URL `https://arxiv.org/abs/2403.01528`.

[35] M Steinegger and J Söding. Clustering huge protein sequence sets in linear time. nat commun 9: 2542, 2018.

[36] Matthias Blum, Antonina Andreeva, Laise Cavalcanti Florentino, Sara Rocio Chuguransky, Tiago Grego, Emma Hobbs, Beatriz Lazaro Pinto, Ailsa Orr, Typhaine Paysan-Lafosse, Irina Ponamareva, et al. Interpro: the protein sequence classification resource in 2025. **Nucleic acids research**, 53(D1):D444–D456, 2025.

[37] Brigitte Boeckmann, Amos Bairoch, Rolf Apweiler, Marie-Claude Blatter, Anne Estreicher, Elisabeth Gasteiger, Maria J. Martin, Karine Michoud, Claire O'Donovan, Isabelle Phan, Sandrine Pilbout, and Michel Schneider. The swiss-prot protein knowledgebase and its supplement trembl in 2003. **Nucleic Acids Research**, 31(1):365–370, 01 2003. ISSN 0305-1048. doi: 10.1093/nar/gkg095. URL `https://doi.org/10.1093/nar/gkg095`.

[38] Jacob Austin, Daniel D. Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. Structured denoising diffusion models in discrete state-spaces. In **Advances in Neural Information Processing Systems**, 2021.

[39] Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, Jingyang Ou, Jun Hu, Jun Zhou, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. Large language diffusion models. **arXiv preprint arXiv:2502.09992**, 2025.

[40] Subham Sekhar Sahoo, Marianne Arriola, Aaron Gokaslan, Edgar Mariano Marroquin, Alexander M Rush, Yair Schiff, Justin T Chiu, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. In **The Thirty-eighth Annual Conference on Neural Information Processing Systems**, 2024. URL `https://openreview.net/forum?id=L4uaAR4ArM`.

[41] Jinsong Li, Xiaoyi Dong, Yuhang Zang, Yuhang Cao, Jiaqi Wang, and Dahua Lin. Beyond fixed: Training-free variable-length denoising for diffusion large language models. **arXiv preprint arXiv:2508.00819**, 2025.

[42] 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**, 2024. doi: 10.1101/2023.09.11.556673. URL `https://www.biorxiv.org/content/early/2024/11/04/2023.09.11.556673`.

[43] Jiaxin Shi, Kehang Han, Zhe Wang, Arnaud Doucet, and Michalis K. Titsias. Simplified and generalized masked diffusion for discrete data. In **Advances in Neural Information Processing Systems**, 2024.

[44] 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, 2024.

[45] Debora S. Marks, Lucy J. Colwell, Robert Sheridan, Thomas A. Hopf, Andrea Pagnani, Riccardo Zecchina, and Chris Sander. Protein 3d structure computed from evolutionary sequence variation. **PLOS ONE**, 6(12):1–20, 12 2011. doi: 10.1371/journal.pone.0028766. URL `https://doi.org/10.1371/journal.pone.0028766`.

[46] Roshan Rao, Nicholas Bhattacharya, Neil Thomas, Yan Duan, Peter Chen, John Canny, Pieter Abbeel, and Yun Song. Evaluating protein transfer learning with tape. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, **Advances in Neural Information Processing Systems**, volume 32. Curran Associates, Inc., 2019.

[47] Christian Dallago, Jody Mou, Kadina E Johnston, Bruce Wittmann, Nick Bhattacharya, Samuel Goldman, Ali Madani, and Kevin K Yang. FLIP: Benchmark tasks in fitness landscape inference for proteins. In **Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)**, 2021. URL `https://openreview.net/forum?id=p2dMLEwL8tF`.

[48] Minghao Xu, Zuobai Zhang, Jiarui Lu, Zhaocheng Zhu, Yangtian Zhang, Ma Chang, Runcheng Liu, and Jian Tang. Peer: A comprehensive and multi-task benchmark for protein sequence understanding. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, **Advances in Neural Information Processing Systems**, volume 35, pages 35156–35173. Curran Associates, Inc., 2022.

[49] Shengchao Liu, Yanjing Li, Zhuoxinran Li, Anthony Gitter, Yutao Zhu, Jiarui Lu, Zhao Xu, Weili Nie, Arvind Ramanathan, Chaowei Xiao, Jian Tang, Hongyu Guo, and Anima Anandkumar. A text-guided protein design framework. **Nature Machine Intelligence**, 7(4):580–591, March 2025. ISSN 2522-5839. doi: 10.1038/s42256-025-01011-z. URL http://dx.doi.org/10.1038/s42256-025-01011-z.

[50] John B. Ingraham, Max Baranov, Zak Costello, Karl W. Barber, Wujie Wang, Ahmed Ismail, Vincent Frappier, Dana M. Lord, Christopher Ng-Thow-Hing, Erik R. Van Vlack, Shan Tie, Vincent Xue, Sarah C. Cowles, Alan Leung, João V. Rodrigues, Claudio L. Morales-Perez, Alex M. Ayoub, Robin Green, Katherine Puentes, Frank Oplinger, Nishant V. Panwar, Fritz Obermeyer, Adam R. Root, Andrew L. Beam, Frank J. Poelwijk, and Gevorg Grigoryan. Illuminating protein space with a programmable generative model. **Nature**, 623(7989):1070–1078, 2023. doi: 10.1038/s41586-023-06728-8.

[51] Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, Saizhuo Wang, Kun Zhang, Yuanzhuo Wang, Wen Gao, Lionel Ni, and Jian Guo. A survey on llm-as-a-judge, 2025. URL https://arxiv.org/abs/2411.15594.

[52] Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Allan dos Santos Costa, Maryam Fazel-Zarandi, Tom Sercu, Sal Candido, et al. Language models of protein sequences at the scale of evolution enable accurate structure prediction. **bioRxiv**, 2022.

[53] 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. **arXiv preprint arXiv:2004.03497**, 2020.

[54] Erik Nijkamp, Jeffrey A Ruffolo, Eli N Weinstein, Nikhil Naik, and Ali Madani. Progen2: exploring the boundaries of protein language models. **Cell systems**, 14(11):968–978, 2023.

[55] Aadyot Bhatnagar, Sarthak Jain, Joel Beazer, Samuel C. Curran, Alexander M. Hoffnagle, Kyle Shan Ching, Michael Martyn, Stephen Nayfach, Jeffrey A. Ruffolo, and Ali Madani. Scaling unlocks broader generation and deeper functional understanding of proteins. In **The Thirty-ninth Annual Conference on Neural Information Processing Systems**, 2026. URL https://openreview.net/forum?id=yvGL2HP7pU.

[56] Xinyou Wang, Zaixiang Zheng, Fei Ye, Dongyu Xue, Shujian Huang, and Quanquan Gu. Dplm-2: A multimodal diffusion protein language model. **arXiv preprint arXiv:2410.13782**, 2024.

[57] 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. In **International Conference on Machine Learning**, 2025.

[58] Xinyou Wang, Liang Hong, Jiasheng Ye, Zaixiang Zheng, Yu Li, Shujian Huang, and Quanquan Gu. Towards a generative protein evolution machine with dplm-evo, 2026. URL https://arxiv.org/abs/2605.00182.

[59] Nuowei Liu, Jiahao Kuang, Yanting Liu, Changzhi Sun, Tao Ji, Yuanbin Wu, and Man Lan. Protein design with dynamic protein vocabulary. **arXiv preprint arXiv:2505.18966**, 2025.

[60] Fengyuan Dai, Shiyang You, Yudian Zhu, Yuan Gao, Lihao Fu, Xibin Zhou, Jin Su, Chentong Wang, Yuliang Fan, Xiaoxiao Ma, Xianjun Deng, Letong Yu, Hui Qian, Yan He, Yitao Ke, Chenchen Han, Xing Chang, Liangzhen Zheng, Sheng Wang, Yajie Wang, Anping Zeng, Shunzhi Wang, Tong Si, Jianming Liu, Hongyuan Lu, and Fajie Yuan. Toward de novo protein design from natural language. **bioRxiv**, 2025. doi: 10.1101/2024.08.01.606258. URL https://www.biorxiv.org/content/early/2025/09/16/2024.08.01.606258.1.

[61] Yingce Xia, Peiran Jin, Shufang Xie, Liang He, Chuan Cao, Renqian Luo, Guoqing Liu, Yue Wang, Zequn Liu, Yuan-Jyue Chen, Zekun Guo, Yeqi Bai, Pan Deng, Yaosen Min, Ziheng Lu, Hongxia Hao, Han Yang, Jielan Li, Chang Liu, Jia Zhang, Jianwei Zhu, Ran Bi, Kehan Wu, Wei Zhang, Kaiyuan Gao, Qizhi Pei, Qian Wang, Xixian Liu, Yanting Li, Houtian Zhu, Yeqing Lu, Mingqian Ma, Zun Wang, Tian Xie, Krzysztof Maziarz, Marwin Segler, Zhao Yang, Zilong Chen, Yu Shi, Shuxin Zheng, Lijun Wu, Chen Hu, Peggy Dai, Tie-Yan Liu, Haiguang Liu, and Tao Qin. Nature language model: Deciphering the language of nature for scientific discovery, 2025. URL https://arxiv.org/abs/2502.07527.

[62] Anthropic. Claude for life sciences. https://www.anthropic.com/news/claude-for-life-sciences, October 2025.

[63] OpenAI. Introducing GPT-Rosalind. https://openai.com/index/introducing-gpt-rosalind/, 2026.

[64] Mihaly Varadi, Damian Bertoni, Paulyna Magana, Urmila Paramval, Ivanna Pidruchna, Malarvizhi Radhakrishnan, Maxim Tsenkov, Sreenath Nair, Milot Mirdita, Jingi Yeo, Oleg Kovalevskiy, Kathryn Tunyasuvunakool, Agata Laydon, Augustin Žídek, Hamish Tomlinson, Dhavanthi Hariharan, Josh Abrahamson, Tim Green, John Jumper, Ewan Birney, Martin Steinegger, Demis Hassabis, and Sameer Velankar. Alphafold protein structure database in 2024: providing structure coverage for over 214 million protein sequences. **Nucleic Acids Research**, 52(D1):D368–D375, 01 2024. ISSN 0305-1048. doi: 10.1093/nar/gkad1011. URL https://doi.org/10.1093/nar/gkad1011.

[65] Junbo Yin, Chao Zha, Wenjia He, Chencheng Xu, and Xin Gao. CFP-gen: Combinatorial functional protein generation via diffusion language models. In **Forty-second International Conference on Machine Learning**, 2025. URL https://openreview.net/forum?id=EiM163eZyg.

[66] Sean Welleck, Ilia Kulikov, Stephen Roller, Emily Dinan, Kyunghyun Cho, and Jason Weston. Neural text generation with unlikelihood training, 2019. URL https://arxiv.org/abs/1908.04319.

[67] The UniProt Consortium. Uniprot: the universal protein knowledgebase in 2025. **Nucleic Acids Research**, 53 (D1):D609–D617, 01 2025. ISSN 1362-4962. doi: 10.1093/nar/gkae1010. URL https://doi.org/10.1093/nar/gkae1010.

[68] Philip Jones, David Binns, Hsin-Yu Chang, Matthew Fraser, Weizhong Li, Craig McAnulla, Hamish McWilliam, John Maslen, Alex Mitchell, Gift Nuka, Sebastien Pesseat, Antony F. Quinn, Amaia Sangrador-Vegas, Maxim Scheremetjew, Siew-Yit Yong, Rodrigo Lopez, and Sarah Hunter. Interproscan 5: genome-scale protein function classification. **Bioinformatics**, 30(9):1236–1240, 05 2014. ISSN 1367-4803. doi: 10.1093/bioinformatics/btu031. URL https://doi.org/10.1093/bioinformatics/btu031.

Appendix

A Methodology Notation

In Table 3, we summarize and provide corresponding descriptions for the notations used when introducing our block-wise diffusion large language model backbone AMix-2 dLLM in Section 3.

Symbol	Description
$\mathcal{V}$	Shared discrete vocabulary
K	Vocabulary size (excluding `[MASK]`)
$\mathbf{m}$	Absorbing `[MASK]` token vector (e_{K+1})
x_0	Clean token (one-hot vector)
$x_{i,0}$	Clean token at position i
$\mathbf{x}_0$	Clean sequence
$\tilde{x}_t$	Corrupted token at time step t
$\tilde{x}_{i,t}$	Corrupted token at position i and time t
$\tilde{\mathbf{x}}_t$	Corrupted sequence at time step t
α_t	Monotonically decreasing noise schedule at step t
Q_t	Transition matrix for the forward process
$\mathbf{x}$	Interleaved full sequence
L	Total number of partitioned contiguous blocks
D	Number of tokens per block
B_k	The k-th clean block
$B_{<k}$	Preceding clean or fully denoised blocks $(B_1, \ldots, B_{k-1})$
$\widetilde{B}_{k,t}$	Corrupted k-th block at noise level t
$M_{k,t}$	Set of masked positions in block k at time t
S	Interleaved training sequence of clean and corrupted blocks
A_{uv}	Binary attention mask between positions u and v

Table 3 Summary of notations used in the dLLM architecture.

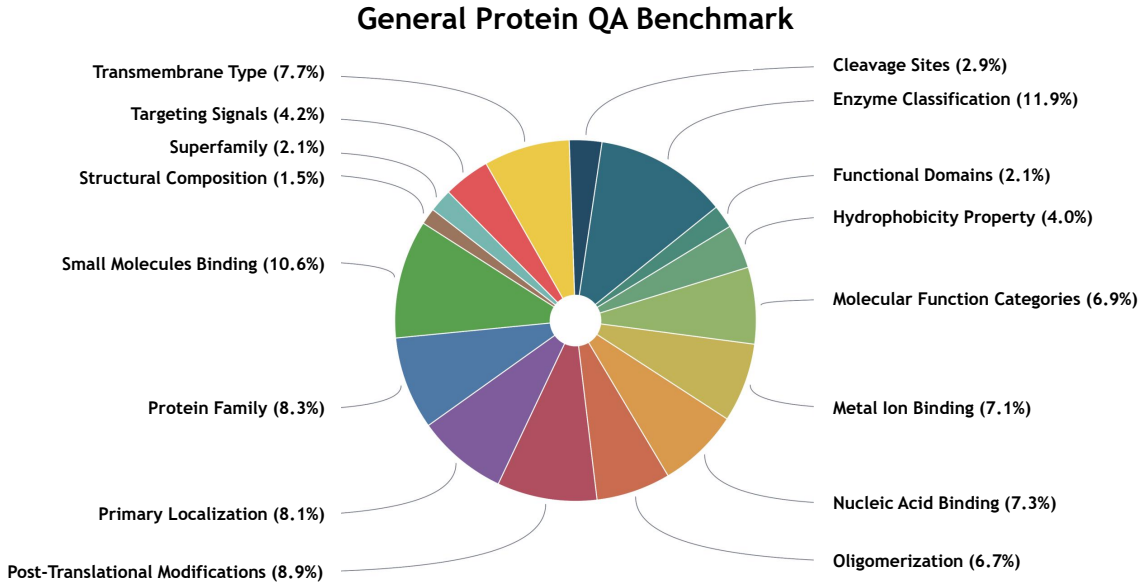

Figure 4 Task distribution of the General Protein QA track of ProteinArena.

B Data Instances

B.1 General Protein QA Tasks

We present representative questions for each of the 16 protein understanding categories in Table 4, which are further paraphrased before being included for evaluation in ProteinArena. These tasks are categorized into 5 core dimensions of protein knowledge: Function, Interaction and Binding, Location and Modification, Physicochemical Property, and Structure. This multi-dimensional taxonomy ensures a holistic evaluation of the model's capacity to internalize complex biological priors and generalize across the vast landscape of protein research.

Task Category	Aspect	Representative Question
Enzyme Classification	Function	Assign the top-level enzyme class (oxidoreductase, transferase, hydrolase, lyase, isomerase, ligase, or translocase) or declare non-enzymatic if no enzymatic function is indicated.
Functional Domains	Function	What functional domains are identified in this protein?
Molecular Function Categories	Function	Infer the single most specific molecular function category, e.g. enzyme, transporter, ion channel, receptor, nucleic acid-binding, regulator, structural molecule, motor protein, or other.
Protein Family	Function	Which protein family does this sequence belong to?
Superfamily	Function	Assign the most plausible evolutionary superfamily membership.
Metal Ion Binding	Interaction and Binding	Is the amino acid sequence likely to coordinate metal ions? If yes, name the most probable metal type when determinable. (e.g. Zinc ions, Calcium ions, Iron ions, No metal binding)
Nucleic Acid Binding	Interaction and Binding	Does this protein bind nucleic acids and what type? (e.g. DNA-binding, RNA-binding, both DNA and RNA binding, no nucleic acid binding)
Oligomerization	Interaction and Binding	Infer the most probable oligomerization state (e.g., monomer, homodimer, heterodimer, higher-order oligomer) for the protein.
Small Molecules Binding	Interaction and Binding	Judge whether this protein binds to small-molecule, and if yes, name the single most probable cofactor type.
Cleavage Sites	Location and Modification	What types of cleavage motifs are present? (e.g. signal peptide cleavage, propeptide cleavage, protease sites, no cleavage sites)
Post-Translational Modifications	Location and Modification	What types of post-translational modifications are likely? (e.g. phosphorylation, glycosylation, ubiquitination, methylation, acetylation, multiple modifications, no PTMs predicted)
Primary Localization	Location and Modification	Where is this protein primarily localized in the cell? (e.g. cytosol, nucleus, membrane, secreted, mitochondrion, other)
Targeting Signals	Location and Modification	What targeting signals are present in this protein? (e.g. signal peptide, mitochondrial targeting sequence, nuclear localization signal, no targeting signals predicted)
Hydrophobicity Property	Physicochemical Property	What is the hydrophobic character of this protein? (e.g. highly hydrophobic, amphipathic, mostly hydrophilic, mixed regions)
Structural Composition	Structure	What broad structural fold class does this protein belong to? (e.g. all-alpha, all-beta, alpha/beta, alpha+beta, membrane protein, other)
Transmembrane Type	Structure	Is this a transmembrane protein and what type? (e.g. single-pass, multi-pass, no transmembrane regions)

Table 4 Task categories and representative questions for General Protein QA.

B.2 Instruction-Tuning Data

We provide data cases for the tasks of General Protein QA, Functional De Novo Design, EC Prediction, and CATH Prediction in Tables 5 to 8 to better depict the objectives of each task and their differences.

Field	Content
Instruction	Determine the principal subcellular compartment in which this protein is found. The protein is `MNI FNQLKQDIIVASRQLYNNQEIANTATIEIPKDSFNGDLSSNIAMIIAAKESIAPREVALKFKEVLITLPYIASIEIAGPGFINF TIKADSWQASIKDILQHEEKFFEIDIDKSRNINIEYVSANPTGPMHIGHARGAVYGDVLARILQKVSYSVTKEYYVNDAGSQIN DLVSTVLLRYKEALGEQITIPAGLYPGEYLIPLGQILAKEYGNKLLTMNYAERFKIIKSFAVEKMLDLNRKDLADLGIKHDIFF SEQSLHDKGEIEETVKLLERMGLIYEGTLPAPKGKIHEEWDNRVQKLFKSTKYGDSQDRPIEKADGSWSYFASDLAYAKDKIER GANHLIYVLGADHSGYVKRIEAIVKALGKEQVKVDVKICQLVNFVENGVPVKMSKRLGSFASVQDVNHEVGKDIIRFMMLTRQN DKPLDFDLVKVKEQSRENPIFYVQYAHVRTISILSKAKELMPESYNNFESGKYDLSLLSSEEEIEIIKLLASWTKTLEASAKYF EPHRIAFYLINLASKFHSMWNFGKENSDYRFVIESNKELTLARLALASAIQKVIASGLEVIGVEPMNKM`.
Answer	Cytoplasm (cytosol)
Rationale	The protein sequence lacks an N-terminal signal peptide or transmembrane helices, indicating it is not secreted or membrane-bound. The protein belongs to the class-I aminoacyl-tRNA synthetase family, enzymes that function in the cytosol to charge tRNAs during protein synthesis. The conserved "HIGH" motif (`PTGPMHIGH`) typical of class-I tRNA synthetases is present, confirming its identity as an Arg-tRNA ligase. Such synthetases are typically cytoplasmic in bacteria and eukaryotes, and the absence of organelle-targeting signals (e.g., mitochondrial presequences) further supports a cytosolic location. Therefore, the most plausible primary localization for this ArgRS is the cytoplasm (cytosol).

Table 5 Data instance for General Protein QA.

Field	Content
Instruction	Create a biologically plausible complete protein sequence that would exhibit the characteristic properties and functions of the KOW protein domain. Consider the typical structural features, length, and functional characteristics associated.
Answer	The `KNDTVVVLSGDDKGKQGAVLELIPAKKAAIV` is a conserved KOW domain characterized by a β-barrel fold and clusters of basic residues for RNA binding; it could be constructed into the full-length `MGKI RKNDTVVVLSGDDKGKQGAVLELIPAKKAAIVKGVNIKTKHRKPSNKNTSGEIITFEAPILLSKLALVAKKATKDKPAIPTRVG FKIENKKKIRIAKKTGKAI`.
Rationale	The design requirements specify a protein that contains a KOW domain within a compact basic RNA-binding protein of roughly 70–100 aa. KOW domains are common in ribosomal proteins. The `KNDTVVVLSGDDKGKQGAVLELIPAKKAAIV` segment satisfies these requirements. Its key features include: • A conserved G-rich loop (`G`) that forms part of the β-barrel core. • Basic Lysine (`K`) residues distributed to create a positive RNA-binding surface. • A pattern of alternating hydrophobic (`V L`) and polar residues that stabilizes the barrel. To build the complete protein, combine: 1. An N-terminal basic tail rich in (`K R`) that may enhance RNA association. 2. The central KOW-like `KNDTVVVLSGDDKGKQGAVLELIPAKKAAIV` segment providing the β-barrel structure. 3. A C-terminal helix with conserved (`A L`)-rich residues to complete the ribosomal protein fold. Connecting these elements yields a plausible full protein sequence, denoted as `MGKIRKNDTVVVLSGDDK GKQGAVLELIPAKKAAIVKGVNIKTKHRKPSNKNTSGEIITFEAPILLSKLALVAKKATKDKPAIPTRVGFKIENKKKIRIAKK TGKAI`, which meets the length and functional criteria for a KOW-containing ribosomal protein.

Table 6 Data instance for Functional De Novo Design.

For protein understanding tasks, question-answer pairs are extracted based on relevant protein context from curated Swiss-Prot annotations within UniProtKB [30] and CARE [31]. These questions are concatenated with their corresponding protein sequences to form comprehensive training instructions. To bolster the reasoning capabilities of AMix-2 and facilitate the acquisition of generalizable biochemical principles, we incorporate step-by-step rationales derived from and verified against the protein records. For the design task, instructions are formulated based on functional constraints from InterPro [36], with answers and rationales synthesized according to the motif fragments and specific protein characteristics under strict fidelity validation. Amino acids are tokenized independently from natural language, establishing protein sequences as a native modality within the model.

Field	Content
Instruction	Determine the most appropriate four-level EC number for the protein whose amino-acid sequence is provided. The protein is `MTTSDLPAFWTVIPAAGVGSRMRADRPKQYLDLAGRTVIERTLDCFLEHPMLRGLVVCLAEDD PYWPGLDCAASRHVQRAAGGVERADSVLSGLLRLLELGARADDWVLVHDAARPNLTRGDLDRLLEELAEDPVGGLLAVPARDTL KRSDRDGRVSETIDRSVVWLAYTPQMFRLGALHRALADALVAGVAITDEASAMEWAGYAPKLVEGRADNLKITTPEDLLRLQRS FPH`.
Answer	2.7.7.60
Rationale	The protein is 234 residues long, which matches the typical size (220–240 aa) of enzymes belonging to the IspD family (2-C-methyl-D-erythritol 4-phosphate cytidylyltransferase). IspD enzymes are cytidylyltransferases that catalyze the transfer of a nucleotidyl group to a phosphorylated substrate, a reaction classified under transferases that transfer phosphorus-containing groups. Sequence analysis reveals motifs characteristic of nucleotidyltransferases, such as the conserved `HDAA` segment and a downstream `HRAL` motif, both of which are commonly found in IspD proteins and are required for binding CTP and the phosphorylated substrate. The overall fold predicted from the length and conserved residues is consistent with the cytidylyltransferase (CTP-binding) family, placing the enzyme in EC class 2 (transferases), subclass 7 (transferring phosphorus-containing groups), sub-subclass 7 (nucleotidyltransferases). The specific activity described for IspD—conversion of 2-C-methyl-D-erythritol 4-phosphate to its CDP-activated form—is uniquely assigned the serial number 60 in this sub-subclass. Therefore, the most likely EC number for this sequence is 2.7.7.60.

Table 7 Data instance for EC Prediction.

Field	Content
Instruction	Determine the most probable CATH hierarchical classification (x.x.x.x) for the provided protein sequence. The protein is `MLNTELLTTGVKGLDELLGGGVAKGVILQVYGPFATGKTTFAMQVGLLNEGKVAYVDTEGGFS PERLAQMAESRNLDVEKALEKFVIFEPMDLNEQRQVIARLKNIVNEKFSLVVVDSFTAHYRAEGSREYGELSKQLQVLQWIARR KNVAVIVVNQVYYDSNSGILKPIAEHTLGYKTKDILRFERLRVGVRIAVLERHRFRPEGGMVYFKITDKGLEDVKNED`.
Answer	3.40.50.300
Rationale	The sequence is 225 residues long, a size typical for small nucleotide-binding enzymes. A close inspection shows the classic Walker A P-loop motif: `GPFATGKTT`, where the conserved `GxxxxGKT` pattern is present. Further downstream a Walker B acidic motif `DE` can be identified, confirming the presence of a P-loop NTP-binding domain. Such domains are built from alternating β-strands and α-helices, placing them in CATH class 3 (Alpha-Beta). The overall arrangement matches the 3-Layer (aba) Sandwich architecture, corresponding to CATH level 2 code 3.40. The connectivity of the β-sheet and surrounding helices follows the Rossmann-like topology, giving level 3 code 3.40.50. Finally, the combination of the Walker motifs and the Rossmann topology is characteristic of the P-loop containing nucleotide triphosphate hydrolases superfamily, which is assigned CATH level 4 code 300. Therefore, the most likely CATH number for this protein is 3.40.50.300.

Table 8 Data instance for CATH Prediction.

C Experimental Setup

C.1 Inference Setting

Bio Tools Bio-tool baselines are employed as retrieval methods, and evaluated on the enzyme commission (EC) prediction and CATH structural fold classification tasks. The reference proteins are indexed into a database, with each query sequence searched against that database, and the annotation of the top-1 hit is transferred as the prediction.

BLAST-DIAMOND is used as a representative sequence-based retrieval tool, following the CARE [31] pre-processing utilities. The database is built from the UniProt enzyme reference FASTA and thus precludes non-enzyme candidates.

Foldseek [13] is used for structure-based search against the prepared AlphaFoldDB-V4 database [64].

Bio Models Bio-model baselines are evaluated on EC, CATH, and when supported by the model interface, InterPro-guided functional design.

ESM2 [6] is a large-scale transformer-based protein language model trained with a masked language modeling objective to internalize evolutionary patterns from hundreds of millions of diverse sequences. In this study, we employ the `esm2-t33-650M-UR50D` checkpoint to extract high-dimensional protein representations, which are subsequently fed into a task-specific trainable head for downstream predictions.

ESM3 [10] is a large-scale generative masked protein language model that jointly reasons over sequence, structure, and function tracks represented as discrete tokens, and fills masked positions through iterative sampling. It can be used for both understanding and design tasks, and is run with the `ESM3-sm-open-v1` checkpoint. For protein design, ESM3 is function-annotation-conditioned rather than instruction-conditioned. The primary protocol keeps only InterPro IDs supported by the ESM3 tokenizer and builds the precisely spanned `FunctionAnnotation` constraints. ESM3 supports a larger subset of InterPro labels compared to CFP-Gen, covering 826/870 samples.

ProTrek [9] is a tri-modal protein language model that enables joint contrastive learning across protein sequence, structure, and function tracks. By integrating a pre-trained ESM encoder for amino acid sequences, a BERT-based text encoder for functional descriptions, and a structural encoder for Foldseek-derived 3Di tokens, ProTrek maps multiple modalities into a shared representation space. In this work, we focus on its sequence and structure modalities by concatenating the representations using the `ProTrek-650M-UniRef50` checkpoint.

SaProt [8] is a structure-aware protein language model that utilizes a novel "structure-aware" (SA) vocabulary, which integrates residue types with structural information. By encoding 3D structures into discrete tokens using Foldseek, SaProt represents proteins as sequences of SA tokens, combining primary sequence data with tertiary geometric conformations. We utilize the `SaProt-650M-AF2` checkpoint to extract representations.

ProteinDT [49] is a text-guided protein design framework with three stages: `ProteinCLAP` aligns protein and text representations through contrastive learning, a facilitator maps text representations toward the protein latent space, and a decoder generates protein sequences from the resulting representation. The model is text-conditioned, but its upstream text encoder is trained on SwissProtCLAP-style UniProt free text rather than imperative design instructions. Therefore, the primary protocol uses prompts constructed from matched UniProt comment text. The prompt builder concatenates available UniProt XML text comments, including function, catalytic activity, pathway, subunit, similarity, and cofactor descriptions. The selected decoder variant is T5, a Transformer-based autoregressive decoder, using the released `ProtBERT-BFD/SciBERT`

`https://github.com/facebookresearch/esm`
`https://github.com/evolutionaryscale/esm`
`https://github.com/westlake-repl/ProTrek`
`https://github.com/westlake-repl/SaProt`
`https://github.com/chao1224/ProteinDT`

text-protein checkpoint and T5 decoder checkpoint. Inference uses default setting and one explicit seed per design.

Pinal [60] is a natural-language-to-protein design baseline that first converts a text description into structural tokens with T2struc and then uses SaProt-T to generate amino-acid sequences conditioned on the predicted structure and text. Pinal accepts the compact InterPro-derived caption text description, namely a semicolon-separated list of `<InterPro-Name> <InterPro-Type>` phrases. The loaded model components are the released `T2struc-1.2B` and `SaProt-T 760M` weights.

Chroma [50] is a programmable generative model for proteins that uses diffusion modeling with equivariant graph neural networks and conditional random fields to sample all-atom structures; its sequence and side-chain design networks are jointly trained with the structure model. Chroma is not conditioned on the raw benchmark instruction in the primary protocol. Instead, captions are passed to `ProCapConditioner`, which provides natural-language conditioning as a differentiable conditioner during sampling. The selected replicated variant uses ProCap-aligned UniProt-style captions; direct InterPro name/type captions and instruction-only captions are used only for ablations. We follow the default setting and use the released Chroma backbone, design, and ProCap checkpoints with the `GPT-Neo-125M` caption model.

CFP-Gen [65] is a combinatorial functional protein generation baseline built on the diffusion language model DPLM [27]. It introduces annotation-guided feature modulation for composable functional labels and residue-controlled functional encoding for residue-wise control, supporting GO, InterPro, EC, sequence-motif, and backbone constraints. CFP-Gen is class-conditional rather than natural-language-conditioned. The current protocol conditions only on directly supported InterPro identifiers from the denovo benchmark; no InterPro identifier is rescued through Gene Ontology mappings. Supported InterPro IDs are mapped to CFP-Gen class IDs, and targets without a supported InterPro ID are dropped. For each seed, the generated config uses the `CFP-Gen 650M` checkpoint following default configuration. CFP-Gen fails to support most of the newly incorporated InterPro labels, and thus only 319 samples out of 870 have been evaluated.

ProDVa [59] is a dynamic-vocabulary augmented protein design baseline that combines a text encoder for functional descriptions, a protein language model for sequence generation, and a fragment encoder/retriever that injects protein fragments as a task-specific dynamic vocabulary. The functional-design protocol uses CAMEO-style keyword-list prompts and ProDVa's retrieval-backed dynamic vocabulary. The active configuration uses the ProDVa-CAMEO checkpoint, CAMEO protein-fragment phrases, and PubMedBERT embeddings. Inference follows default configuration.

Frontier LLMs Frontier LLMs are evaluated across all the protein understanding tasks and the functional-design task. We use default temperature setting, with `top_p` also omitted unless explicitly required by the API or reasoning mode (e.g., `top_p=0.95` for Claude). We set `max_tokens=8192` by default, and relax it to `32768` only when the task output may exceed the default limit. For functional design, the model is additionally required to return exactly one uppercase amino-acid sequence with length at most 1024 residues.

AMix-2 For protein understanding tasks, AMix-2 dLLM uses `temperature=0.7`, `top_p=0.9` and `max_tokens=2048`, while AMix-2 AR uses `temperature=0.7`, `top_p=0.5` and `max_tokens=4096`. For protein design, both models use `temperature=0.7` and `top_p=0.6`, which we find more suitable for generation over the much smaller amino-acid vocabulary (20 amino acid tokens). Across tasks, we find that temperatures around 0.6–0.8 work well in practice; top-p around 0.6 is generally preferable for protein design, while 0.6–0.9 works fine for protein understanding. Note that the meaning of temperature is similar between AR and dLLM, as both use it to control the sharpness of the sampling distribution, but for top-p it differs: for AR, it follows standard nucleus sampling based on the cumulative next-token probability mass, whereas for dLLM, it controls truncation of marginal probabilities across parallel tokens at each iterative denoising step, making it less susceptible to compounding exposure bias.

`https://github.com/westlake-repl/Denovo-Pinal`
`https://github.com/generatebio/chroma`
`https://github.com/yinjunbo/cfpgen`
`https://github.com/sornkL/ProDVa`

Model	Conditioning Type	Representative Prompt
CFP-Gen	Class-label conditioning (supported labels only)	`IPR008567`, `IPR013785`.
ESM3	Class-label conditioning as function annotation (supported labels only)	`FunctionAnnotation(label=IPR008567, start=S1, end=E1)`, `FunctionAnnotation(label=IPR013785, start=S2, end=E2)`.
Chroma	Text prompt from UniProt functional comments via `ProCapConditioner` caption	Catalyzes the condensation of 3,5-dioxohexanoate and acetyl-CoA, forming acetoacetate and acetoacetyl-CoA. May be involved in fatty acid biosynthesis rescue via triacetic acid lactone. Belongs to the BKACE family.
ProteinDT	Text prompt from UniProt comments	Catalyzes the condensation of 3,5-dioxohexanoate and acetyl-CoA, forming acetoacetate and acetoacetyl-CoA. May be involved in fatty acid biosynthesis rescue via triacetic acid lactone. 3,5-dioxohexanoate + acetyl-CoA = acetoacetyl-CoA + acetoacetate. Belongs to the BKACE family.
Pinal	Text prompt from InterPro labels	Beta-keto acid cleavage enzyme family; Aldolase-type TIM barrel homologous superfamily; Beta-keto acid cleavage enzyme family
ProDVa	Text prompt constructed from InterPro labels	Generate a protein sequence for a novel protein that integrates the following function keywords: Beta-keto acid cleavage enzyme, Aldolase-type TIM barrel. The designed protein sequence is

Table 9 Representative conditioning inputs built for the bio-model functional-design baselines. The examples correspond to the same denovo target with InterPro annotations `IPR008567` (BKACE) and `IPR013785` (Aldolase TIM).

C.2 Evaluation Metrics

Functional Design In addition to the accuracy adopted for protein understanding, we detail the metrics calculation for functional design as follows, where each method is evaluated in terms of sequence repetition, foldability, functional recovery, and novelty.

For sequence repetition, we report `Repeat` [33], which measures the fraction of residues contained in tandemly repeated regions, together with `Rep2` and `Rep5` following Rep-N [66]. For n-gram repetition, we compute

$$\mathrm{Rep}_n(x) = 1 - \frac{|\mathrm{unique}(G_n(x))|}{|G_n(x)|},$$

where $G_n(x)$ is the multiset of all overlapping n-grams in sequence x. We report `Rep2` and `Rep5` by setting $n = 2$ and $n = 5$, respectively, and averaging over generated sequences.

For structural plausibility, we use `ESMFold-v1` [6] as the folding model to calculate pLDDT.

For sequence-level similarity metrics, we employ MMseqs2 [12] for sequence identity against all UniProt sequences up until **December 31, 2025** [67], following Kuang et al. [33]. Novelty is calculated as the nearest-neighbor sequence novelty:

$$\mathrm{Novelty}(x) = 1 - \max_y \mathrm{SeqID}(x, y),$$

Note that for the Swiss-Prot test sequence (Natural), we intentionally remove the identical matches as the sequence database being searched against contains the reference sequence itself.

For functional recovery, we first run `InterProScan-5.75-106.0` [68] over generated sequences. Then, we

https://ftp.ebi.ac.uk/pub/software/unix/iprscan/5/5.75-106.0/

compute the recovery rate between the generated sequence s and reference sequence s_{ref} as

$$\text{Recovery Rate} = \begin{cases} \dfrac{|\text{InterProScan}(s) \cap \text{InterProScan}(s_{\text{ref}})|}{|\text{InterProScan}(s_{\text{ref}})|}, & \text{if InterProScan}(s_{\text{ref}}) \neq \varnothing, \\ \text{N/A}, & \text{if InterProScan}(s_{\text{ref}}) = \varnothing. \end{cases}$$

D Extended Experimental Results

For the **Functional De Novo Design** task, we provide additional focused results on the intersection of samples shared by all baselines. For the hierarchical classification tasks of **EC Prediction** and **CATH Prediction**, we provide evaluation results on proteins corresponding to a greater sequence identity split than 30% with ProteinArena. This indicates cases where the proteins evaluated share sequences with higher similarity to those readily trained for each model, thus resulting in better overall performance.

Functional De Novo Design As described in Appendix Section C.1, protein-specialized language models impose more restrictive input requirements, being inherently limited by its training functional keywords and cannot generalize to unseen data, therefore some of them cannot be evaluated on the full test set. To enable a better controlled apples-to-apples comparison, we report in Table 10 results on the 207-sample intersection subset commonly evaluated by all baselines. This subset is generally easier for protein-specialized models, since corresponding functional keywords are in-distribution under the evaluation settings of all included baselines. Accordingly, both model groups improve on this subset: Bio-models gain an average of 2.48 in pLDDT and 9.05% in IPR, while LLMs improve by 2.05 in pLDDT and 8.52% in IPR.

The overall trend remains unchanged. Protein-specialized models may achieve stronger pLDDT, but are generally less effective at controlling functional keywords, suggesting limitations in projecting textual constraints into the protein space. On the other hand, as exemplified by AMix-2 dLLM and Claude Opus 4.7, LLMs properly trained to capture the protein modality can enjoy the advantage of both high-fidelity protein sequence modeling, and high IPR, which is supported by their strong instruction-following capabilities.

Category	Model	Repetition			Quality		Seq Distribution	
		Rep	Rep2	Rep5	pLDDT	IPR (%)	Novelty	Unique (%)
Reference	Natural	1.78	44.68	0.14	86.06	100.00	4.19	100.00
Bio-Model	CFPGen	10.98	57.86	10.79	<u>78.48</u>	34.00	50.05	100.00
	Chroma	2.41	56.49	0.47	58.69	0.08	59.21	100.00
	ESM3	23.79	71.36	17.85	64.84	31.51	67.68	100.00
	Pinal	11.49	52.28	3.01	75.40	<u>59.57</u>	48.49	100.00
	ProDVa	7.41	30.92	11.62	71.97	32.32	17.66	86.43
	ProteinDT	2.57	51.60	0.34	39.11	21.55	50.40	100.00
LLM	Qwen3.5-27B	13.28	86.45	74.35	29.97	0.12	79.25	100.00
	GLM-5.1	1.21	48.08	1.18	33.98	7.95	51.36	100.00
	DeepSeek-V4-Pro	4.96	53.66	3.98	45.78	37.15	43.35	100.00
	GPT-5.5	2.52	55.84	2.03	40.22	35.88	34.92	100.00
	Gemini 3.1 Pro	2.30	46.02	1.75	47.96	31.90	40.80	99.52
	Claude Opus 4.6	0.98	47.27	0.34	43.47	35.42	38.71	95.17
	Claude Opus 4.7	1.79	43.83	0.05	81.14	**77.83**	35.54	100.00
	AMix-2 AR	1.63	45.26	0.26	75.60	55.45	13.67	51.21
	AMix-2 dLLM	3.14	45.34	0.37	**81.62**	72.25	30.49	96.62

* Results are recomputed on the 207 out of 870 reference sequences shared across all listed baselines.

Table 10 Intersection-only comparison of various models on generation quality, novelty, and uniqueness metrics.

EC Prediction Results in high-homology regimes ($\geq 30\%$ sequence identity) are detailed in Table 11. In these scenarios, protein-specialized models and bioinformatics tools typically define the performance ceiling due to their direct exploitation of evolutionary conservation. Notably, AMix-2 outperforms frontier LLMs across all high-homology regimes, with AMix-2 AR being nearly on par with specialized models like ESM2 and ESM3. This indicates that while most general LLMs fail to utilize high-homology templates effectively, AMix-2 successfully bridges the gap between generative language modeling and classical bioinformatics.

CATH Prediction Evaluation results in Table 12 reveals a sharp contrast between generative LLMs and structure-aware protein models and tools. All six baselines compared achieved near-perfect scores across all identity tiers, as they are specifically optimized for structural fold recognition. However, within the LLM cohort, AMix-2 demonstrates a dominant lead, greatly improving over the strongest frontier Claude Opus 4.7. This suggests that AMix-2's training regime allows it to recognize structural hierarchies from sequences alone far more effectively than standard LLMs, establishing it as a highly reliable generative proxy for structural classification when 3D coordinates are unavailable.

Identity	Method Type	Model	Level 1	Level 2	Level 3	Level 4
30%–50%	Bio-Tool	BLAST	84.26	77.78	75.00	44.44
		Foldseek	90.74	85.19	83.33	49.07
	Bio-Model	ESM2	85.19	79.63	78.70	44.44
		ESM3	85.19	75.00	72.22	41.67
		ProTrek	81.48	73.15	71.30	38.89
		SaProt	83.33	74.07	71.30	41.67
	LLM	Qwen3.5-27B	6.48	1.85	1.85	0.00
		GLM-5.1	44.44	28.70	22.22	8.33
		DeepSeek-V4-Pro	61.11	49.07	46.30	26.85
		GPT-5.5	42.59	28.70	23.15	16.67
		Gemini 3.1 Pro	61.11	49.07	44.44	21.30
		Claude Opus 4.6	54.63	37.04	33.33	18.52
		Claude Opus 4.7	79.63	70.37	65.74	32.41
		AMix-2 AR	80.56	**74.07**	**73.15**	**40.74**
		AMix-2 dLLM	**82.41**	73.15	70.37	38.89
50%–70%	Bio-Tool	BLAST	96.84	89.47	88.42	77.89
		Foldseek	96.84	88.42	87.37	76.84
	Bio-Model	ESM2	96.84	91.58	90.53	78.95
		ESM3	95.79	87.37	87.37	74.74
		ProTrek	93.68	87.37	85.26	75.79
		SaProt	96.84	89.47	88.42	76.84
	LLM	Qwen3.5-27B	14.74	4.21	4.21	0.00
		GLM-5.1	37.89	21.05	17.89	9.47
		DeepSeek-V4-Pro	70.53	60.00	55.79	32.63
		GPT-5.5	57.89	43.16	42.11	24.21
		Gemini 3.1 Pro	68.42	55.79	52.63	30.53
		Claude Opus 4.6	49.47	31.58	28.42	11.58
		Claude Opus 4.7	92.63	83.16	83.16	60.00
		AMix-2 AR	**95.79**	87.37	**87.37**	**72.63**
		AMix-2 dLLM	**95.79**	**89.47**	**87.37**	**72.63**
70%–100%	Bio-Tool	BLAST	98.75	97.50	96.25	88.75
		Foldseek	97.50	96.25	96.25	87.50
	Bio-Model	ESM2	98.75	96.25	95.00	85.00
		ESM3	93.75	90.00	87.50	77.50
		ProTrek	95.00	92.50	90.00	78.75
		SaProt	93.75	90.00	87.50	76.25
	LLM	Qwen3.5-27B	16.25	6.25	5.00	0.00
		GLM-5.1	52.50	35.00	28.75	13.75
		DeepSeek-V4-Pro	83.75	63.75	63.75	41.25
		GPT-5.5	65.00	52.50	50.00	30.00
		Gemini 3.1 Pro	70.00	56.25	55.00	33.75
		Claude Opus 4.6	67.50	52.50	47.50	21.25
		Claude Opus 4.7	92.50	83.75	81.25	58.75
		AMix-2 AR	93.75	**90.00**	**87.50**	**76.25**
		AMix-2 dLLM	**95.00**	88.75	**87.50**	75.00

Table 11 Level-wise EC prediction accuracy on proteins with $\geq 30\%$ sequence identity.

Identity	Method Type	Model	Level 1	Level 2	Level 3	Level 4
30%-50%	Bio-Tool	BLAST	85.87	85.87	85.87	85.87
		Foldseek	100.00	100.00	100.00	100.00
	Bio-Model	ESM2	98.91	98.91	97.83	97.83
		ESM3	95.65	93.48	90.22	90.22
		ProTrek	100.00	97.83	96.74	96.74
		SaProt	96.74	96.74	95.62	95.62
	LLM	Qwen3.5-27B	19.57	13.04	3.26	1.09
		GLM-5.1	56.52	28.26	17.39	7.61
		DeepSeek-V4-Pro	65.22	40.22	32.61	23.91
		GPT-5.5	68.48	41.30	30.43	19.57
		Gemini 3.1 Pro	76.09	54.35	38.04	23.91
		Claude Opus 4.6	71.74	43.48	29.35	23.91
		Claude Opus 4.7	89.13	80.43	73.91	66.30
		AMix-2 AR	93.48	**83.70**	**76.09**	**72.83**
		AMix-2 dLLM	**97.83**	80.43	**76.09**	68.48
50%-70%	Bio-Tool	BLAST	100.00	100.00	100.00	100.00
		Foldseek	100.00	100.00	100.00	100.00
	Bio-Model	ESM2	98.39	98.39	98.39	98.39
		ESM3	100.00	100.00	100.00	100.00
		ProTrek	100.00	100.00	100.00	100.00
		SaProt	100.00	100.00	100.00	100.00
	LLM	Qwen3.5-27B	33.87	11.29	0.00	0.00
		GLM-5.1	66.13	35.48	16.13	6.45
		DeepSeek-V4-Pro	83.87	58.06	53.23	45.16
		GPT-5.5	75.81	43.55	35.48	22.58
		Gemini 3.1 Pro	83.87	64.52	48.39	37.10
		Claude Opus 4.6	75.81	51.61	43.55	25.81
		Claude Opus 4.7	91.94	87.10	83.87	82.26
		AMix-2 AR	**98.39**	**98.39**	**96.77**	**96.77**
		AMix-2 dLLM	91.94	85.48	83.87	80.65
70%-100%	Bio-Tool	BLAST	100.00	100.00	100.00	100.00
		Foldseek	100.00	100.00	100.00	100.00
	Bio-Model	ESM2	100.00	100.00	100.00	100.00
		ESM3	100.00	100.00	100.00	100.00
		ProTrek	100.00	100.00	100.00	100.00
		SaProt	100.00	100.00	100.00	100.00
	LLM	Qwen3.5-27B	32.43	12.16	2.70	0.00
		GLM-5.1	56.76	22.97	13.51	6.76
		DeepSeek-V4-Pro	71.62	55.41	48.65	31.08
		GPT-5.5	63.51	45.95	33.78	20.27
		Gemini 3.1 Pro	75.68	48.65	35.14	27.03
		Claude Opus 4.6	68.92	47.30	32.43	20.27
		Claude Opus 4.7	85.14	72.97	71.62	64.86
		AMix-2 AR	97.30	**95.95**	**95.95**	**93.24**
		AMix-2 dLLM	**98.65**	94.59	94.59	90.54

Table 12 Level-wise CATH prediction accuracy on proteins with $\geq 30\%$ sequence identity.