STEERING PROTEIN LANGUAGE MODELS VIA UNSUPERVISED REWARD OPTIMIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Protein language models (PLMs) have emerged as powerful tools for controllable biomolecular design, yet their post-training adaptation typically relies on costly wet-lab validation or curated preference datasets. To overcome this supervision bottleneck, we introduce unsupervised reward optimization of PLMs, a comprehensive framework for steerable protein generation without ground-truth labels. Our key insight is that task-agnostic rewards, which combine intrinsic model uncertainty with extrinsic semantic consistency informed by protein representation models, exhibit strong correlation with controllability measures across base models and temperature regimes. Building upon this discovery, we propose two offline algorithms: Soft Reward Optimization (SRO) and Binarized Reward Optimization (BRO), which effectively maximize the classical RLHF objective induced by these proxy rewards. Extensive experiments on compositional out-of-distribution prompts demonstrate that both methods significantly outperform competitive baselines (DPO, KTO), while approaching supervised performance across multiple sampling temperatures, model scales and protein families. Moreover, PLMs fine-tuned with unsupervised rewards can achieve consistently higher coverage compared to their base model in pass@k evaluations. By enabling self-improvement of PLMs through their own generated experience, our framework provides a scalable pathway toward controllable biomolecular design in settings where labeled preferences or experimental feedback are scarce or unavailable.

1 INTRODUCTION

Just as large language models (LLMs) encode, express and reason over natural languages of human intelligence, biomolecular foundation models (BFMs) have been developed (Rives et al., 2019; Alley et al., 2019; Elnaggar et al., 2022; Jumper et al., 2021; Baek et al., 2021; Chen et al., 2022; Madani et al., 2023; Watson et al., 2023; Su et al., 2024; Nguyen et al., 2024; Hayes et al., 2025; Lewis et al., 2025; Passaro et al., 2025; Chen et al., 2025; Mo & Li, 2026) to decipher the language of life. Exemplified by protein language models (PLMs) which were trained on billions of biological sequences[1] of proteins, BFMs learn expressive deep representations emerging within language models that reflect the evolution, structure and function of biomolecules without explicit supervision on those properties (Heinzinger et al., 2019; Meier et al., 2021; Rao et al., 2021). Moreover, frontier PLMs like ProGen (Madani et al., 2023; Nijkamp et al., 2023) and ESM3 (Hayes et al., 2025) enable controllable generation and design (Ferruz & Höcker, 2022) by following prompts that specify protein families, functions, or other design objectives. It has been extensively shown that the generation quality and fidelity can be significantly improved with scale by post-training (Hayes et al., 2025; Widatalla et al., 2024; Stocco et al., 2024).

Supervised fine-tuning (SFT) and reinforcement learning from human feedback (RLHF) have emerged as the dominant paradigm for aligning large language models (LLMs) with human preferences during post-training adaptation (Ziegler et al., 2019; Stiennon et al., 2020; Bai et al., 2022; Ouyang et al., 2022; Korbak et al., 2023). More recently, reinforcement learning with verifiable rewards (RLVR) has been popularized to achieve remarkable reasoning capability of LLMs on mathematics, coding and

[1] By *biological sequences*, we mean molecular representations that can be tokenized and serialized for sequential modeling. For instance, multimodal PLMs such as ESM3 employ different input tracks to encode protein sequence, structure, and function.

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

science benchmarks (Shao et al., 2024; Guo et al., 2025; Team, 2025; Yang et al., 2025; Comanici et al., 2025). However, these approaches typically incur high costs for data labeling, and when scaling beyond human expertise in specialized domains, obtaining reliable ground-truth supervision becomes increasingly infeasible (Burns et al.; Silver & Sutton, 2025; He et al., 2026). For post-training BFMs toward generalist biological artificial intelligence (GBAI) (Rao et al., 2026), the challenge intensifies: data curation requires prohibitively expensive wet-lab validation or clinical trials and poses substantial safety risks in healthcare and biomedical applications.

In the domain of LLMs, this supervision bottleneck (He et al., 2026) has spurred growing interest in unsupervised post-training, which leverages intrinsic and extrinsic reward signals without ground-truth labels, achieving encouraging gains in language tasks such as free-form question-answering and reasoning (Zhao et al.; Zuo et al., 2025; Zhang et al., 2025a). Yet in biological domains, how to devise reliable proxy rewards for BFM post-training and equally importantly, how to design algorithms robust to the presumably noisy supervision signals remain largely unexplored.

To address the outstanding challenge of unsupervised post-training for steerable generation of BFMs, we take protein language models as testbeds and conduct a holistic study covering the design of rewards, data sampling and optimization algorithms as well as benchmark tasks for *in silico* evaluation. By examining a broad spectrum of inherent and external statistical metrics across different sampling temperatures, we identify reward functions that exhibit good correlation with the PLM's ability to follow high-level functional and structural instructions. Based on the reward design, we propose two offline algorithms, namely *Soft Reward Optimization (SRO)* and *Binarized Reward Optimization (BRO)* to optimize the classical RLHF objective, by leveraging continuous and discrete reward signals respectively. We find that both proposed methods significantly enhance the steerability of frontier PLMs despite relying solely on noisy supervision, and attain performance comparable to that of the supervised baseline (base model fine-tuned with ground-truth labels) in certain temperature regimes. Our main contributions include:

- We present, to the best of our knowledge, the first investigation to fine-tune protein language models (PLMs) with *unsupervised rewards* for controllable generation. This work paves the way for steerable adaptation and self-improvement of PLMs on offline, unlabeled corpora, with promising implications for biomedical domains that suffer from limited annotated data.
- We provide a novel, comprehensive recipe for unsupervised post-training of PLMs, involving (i) task-agnostic metrics as proxy rewards, (ii) a multi-temperature sampling strategy for creating diverse training sets, and (iii) SRO and BRO algorithms that effectively enhance the instruction-following capability of frontier PLMs, even when trained with noisy supervision.
- Our empirical study demonstrates that under the same unsupervised reward labeling protocol, both SRO and BRO outperform competitive baselines like DPO (Rafailov et al., 2023) and KTO (Ethayarajh et al., 2024) by a substantial margin across various out-of-distribution (OOD) generation tasks, sampling temperatures and model scales.
- To facilitate future research, we provide novel datasets and benchmark tasks by constructing realistic compositional OOD prompts from 7 Pfam (Finn et al., 2014) protein families, to probe the controllability and generalization of PLMs in *de novo* protein design (Ferruz & Höcker, 2022; Winnifrith et al., 2024; Kortemme, 2024; Yang et al., 2026).

2 Related Work

Protein Language Models Inspired by the inherent similarity between natural languages and biological sequences, protein language models (PLMs) have revolutionized computational biology by learning rich, contextual representations from massive unlabeled corpora of protein sequences through NLP-inspired architectures (Vaswani et al., 2017; Devlin et al., 2019) and self-supervised pre-training (Rives et al., 2019; Elnaggar et al., 2022; Heinzinger et al., 2019; Meier et al., 2021; Rao et al., 2021). Pioneering models like ESM (Rives et al., 2019) and ProtTrans (Elnaggar et al., 2022) demonstrate that transformer architectures can implicitly capture evolutionary constraints, structural stability, and functional motifs. This foundational capability has catalyzed a new generation of controllable generative PLMs. For instance, ProGen (Madani et al., 2023; Nijkamp et al., 2023) and ProLLaMA (Lv et al., 2025) enable conditional generation to create novel proteins across diverse families, while frontier multimodal foundation model like ESM3 (Hayes et al., 2025) can jointly

reason over sequence, structure, and function to perform complex, instruction-guided design tasks, such as generating a functional fluorescent protein with low homology to wild types. Recent studies (Widatalla et al., 2024; Stocco et al., 2024; Hayes et al., 2025) apply preference-based (Rafailov et al., 2023), directed evolution (Jiang et al., 2024; Yang et al., 2019) and activation steering (Huang et al., 2025) methods to greatly enhance the steerability of PLMs. However, the reliance on curated datasets of preference pairs and experimental observations significantly constrains their applicability in biological domains. To this end, our work charts a course for PLM self-improvement through their own generated experience instead of human data (Silver & Sutton, 2025), thereby opening a promising avenue towards scientific superintelligence (Burns et al.; Song et al., 2025b).

Unsupervised Fine-Tuning of Language Models Aligning LLMs with complex primitives typically relies on Reinforcement Learning from Human Feedback (RLHF) (Ziegler et al., 2019; Stiennon et al., 2020; Bai et al., 2022; Ouyang et al., 2022). However, such approach is severely hampered by the "supervision bottleneck" (He et al., 2026): acquiring high-quality labels incurs substantial human labor and even worse, requires costly and time-consuming wet-lab experiments or clinical data in biological domains. This challenge has motivated the study of unsupervised post-training methods that use proxy rewards instead of ground-truth labels. A key line of research is Unsupervised Reinforcement Learning with Verifiable Rewards (URLVR), where existing methods can be classified as either *intrinsic* or *extrinsic* based on their source of supervision signals (He et al., 2026). Intrinsic reward methods such as those based on self-certainty (Zhao et al., 2026), entropy (Zhang et al., 2025a; Cui et al., 2025; Agarwal et al., 2025), or majority voting (Zuo et al., 2025) leverage proxy rewards generated solely by the model itself. They operate through a common "sharpening" mechanism that reinforces the model's initial confident predictions, achieving early training gains at the risk of reward hacking and model collapse at later stages (Agarwal et al., 2025; Shafayat et al., 2025; Zhang et al., 2025b). In contrast, extrinsic reward methods generate rewards via mechanisms such as self-supervised objectives on unlabeled corpora (Dong et al., 2025; Hatamizadeh et al., 2025; She et al., 2025; Akter et al., 2026) or external computation from compilers, proof assistants or game engines (Zhao et al.; Simonds & Yoshiyama, 2025; Shao et al., 2025; Hubert et al., 2025), which are entirely independent of the model's internal state. These rewards are generally more robust and scalable than intrinsic ones, however, their reliance on feedback from verifiable environments renders them infeasible in many scientific domains that require large-scale simulation or wet-lab/clinical validation. In this paper, we propose hybrid, task-agnostic reward functions for PLM fine-tuning that combine intrinsic uncertainty measures with self-consistency metrics from an external protein representation model ESMC (ESM Team, 2024). Our empirical study shows that this unique design exhibits consistent agreement with controllability measures (e.g. keyword/structure recovery) without *any prior knowledge* of the ground-truth labels, and is demonstrated to elicit strong instruction-following capability of PLMs via reward optimization algorithms.

3 METHOD

3.1 PRELIMINARIES

To achieve *de novo* protein design via controllable generation, such as generating highly specialized functional proteins, frontier generative PLMs (Madani et al., 2023; Hayes et al., 2025; Nijkamp et al., 2023) are generally[2] trained via a three-stage procedure similar to modern LLMs (Ouyang et al., 2022):

Pretraining Given a large corpus of protein sequences y, train the model π_θ to minimize a generative masked language modeling objective:

$$\mathcal{L}_{\text{pretrain}} = -\mathbb{E}_{y,m} \sum_{i \in m} \log \pi_\theta(y_i | y_{\setminus m}). \tag{1}$$

If m is a causal mask, Eq. (1) becomes the classical autoregressive next-token prediction. This stage endows the model with general knowledge of the statistical and evolutionary patterns of natural protein sequences. The pretrained model π_0 can therefore be used for *unconditional generation*.

[2]Some multimodal PLMs like ESM3 may skip the SFT stage by tokenizing control tags (e.g. structural and functional tokens) in the pretraining stage. However, if the downstream task represents a novel distribution that cannot be naively characterized by known prompts during pretraining, SFT should still be used for adaptation to the new distribution and prompts/conditions.

Supervised Finetuning (SFT) Finetune the model via masked language modeling on data that are more relevant to the downstream task. Analogous to instruction finetuning (Wei et al., 2022) in LLM, SFT data often comprise control tags as instructions x and a completed protein sequence as the response y. The finetuned model π_{ref} is therefore trained by the objective

$$\mathcal{L}_{\text{SFT}} = -\mathbb{E}_{x,y,m} \sum_{i \in m} \log \pi_\theta(y_i | y_{\setminus m}, x) \tag{2}$$

and applicable to *conditional generation*.

Reinforcement Learning (RL) To further align π_{ref} for controllable generation and design, a reward/score function $r(x, y)$ operates to quantify the quality of the generated sequence y given prompt x. Similar to RLHF/RLVR for LLM post-training, this RL stage can be formulated as solving a KL-constrained reward maximization problem:

$$\max_{\pi_\theta} \mathbb{E}_{x \sim \mathcal{D}, y \sim \pi_\theta(y|x)}[r(x, y)] - \beta D_{\text{KL}}[\pi_\theta(y|x) \,\|\, \pi_{\text{ref}}(y|x)]. \tag{3}$$

Following (Rafailov et al., 2023), the problem above has a closed-form solution:

$$\pi^*(y|x) = \frac{1}{Z(x)} \pi_{\text{ref}}(y|x) \exp\left(\frac{1}{\beta} r(x, y)\right), \tag{4}$$

where $Z(x) = \sum_y \pi_{\text{ref}}(y|x) \exp(\frac{1}{\beta} r(x, y))$ is the partition function.

Theorem 3.1. *The reward optimization problem in Eq. (3) is equivalent to minimizing the KL-divergence $D_{KL}[\pi_\theta(y|x) \,\|\, \pi^*(y|x)]$, where $\pi^*(y|x)$ is the optimal policy defined in Eq. (4).*

All proofs are deferred to Appendix B. LLM post-training normally learns a parameterized reward model r_ϕ or applies an implicit reward (Rafailov et al., 2023) $r \propto \log[\pi_\theta / \pi_{\text{ref}}]$ of the model π_θ as a proxy for human preference. However, in this paper, we propose a novel problem named *unsupervised reward optimization for PLMs*. Our goal is to develop an *explicit, a priori* score function $r(x, y)$ without *any knowledge* of ground-truth labels for direct optimization on offline data. In the next section, we will present our unique design choices for such rewards along with sampling strategies for constructing the training sets.

3.2 Task-Agnostic Reward Design and Data Sampling

The guiding principle of our unsupervised reward design is to find a proxy for *controllability*, to measure how well a generated sample aligns with its prompt. Moreover, the reward function should be *task-agnostic* for it to be truly unsupervised. Following current progress in URLVR (He et al., 2026), various intrinsic and extrinsic rewards induced by the reference policy π_{ref}[3] are examined. For the former, we investigate the sequence-level negative log-likelihood (predictive entropy) and its length-normalized variant (normalized entropy)

$$\mathcal{H}_\pi(x, y) = -\sum_{i=1}^{|y|} \log \pi_{\text{ref}}(y_i | y_{<i}, x), \quad \tilde{\mathcal{H}}_\pi(x, y) = -\frac{1}{|y|} \sum_{i=1}^{|y|} \log \pi_{\text{ref}}(y_i | y_{<i}, x) \tag{5}$$

as measures of the model's uncertainty in the generated sequence y given prompt x, which is widely adopted in entropy-based RL (Cui et al., 2025; Ziebart et al., 2008; Haarnoja et al., 2017; 2018). We define the intrinsic reward $r_i(x, y) \triangleq -\mathcal{H}_\pi(x, y)$ or $-\tilde{\mathcal{H}}_\pi(x, y)$ to steer model towards more confident generations. For the latter, inspired by the notion of "semantic equivalence" in language tasks (Kuhn et al., 2023; Farquhar et al., 2024), we obtain protein embeddings using an off-the-shelf representation model $q(z|y)$ and evaluate the distance between generated sequences

$$\bar{d}_q(x, y) = \mathbb{E}_{y' \sim \pi_{\text{ref}}(\cdot|x)}[d(q(y), q(y'))] \simeq \frac{1}{N} \sum_{i=1}^{N} d(q(y), q(y^{(i)})), \quad \text{where } y^{(i)} \sim \pi_{\text{ref}}(\cdot \mid x), \tag{6}$$

d can be any well-defined metric on the latent space, which provides a measure of semantic distance[4] between generated samples. Intuitively, if a generated sequence is semantically distant from other

[3] This enables offline training and self-improvement of the base model as the reference policy.

[4] For proteins, we hypothesize that this semantic distance serves as a faithful measure of the evolutionary, functional and structural similarity between sequences, given that the representation model q is well-trained.

generations conditioning on the same prompt, it is highly likely to be a statistical outlier or "hallucination" of the model. Therefore, we propose to define the extrinsic reward $r_e(x, y) \triangleq -\bar{d}_q(x, y)$ as the *negative semantic distance* to encourage consistent generations. Our construction scales linearly with $\mathcal{O}(N)$ inference calls to model q, significantly improving upon prior LLM-based approaches (Zhang et al., 2025a; Kuhn et al., 2023; Farquhar et al., 2024) that rely on pairwise equivalence checks and exhibit complexity $\mathcal{O}(N^2)$.

Beyond designing proxy rewards, we further study temperature as a key factor in both offline data construction and reward labeling. Our rationale is two-fold. First, it's well-known that LLM generation statistics are highly sensitive to sampling temperatures (Chen et al., 2021; Zhu et al., 2024; Zhang et al., 2024), and in the same vein, PLMs could benefit from a multi-temperature sampling strategy to create more diverse offline dataset for unsupervised reward optimization (verified in Appendix D). Second, we empirically observe that the proposed metrics in Eqs. (5) and (6) exhibit distinct correlation patterns with ground-truth labels at different temperatures. Consequently, for explicit reward optimization, the most strongly correlated metric should be selected at each temperature to minimize the noise of the reward signal. Therefore, given a prompt set $\mathcal{X}$ and temperature set $\mathcal{T}$, we define our multi-temperature sampling and reward labeling strategy for constructing offline training set $\mathcal{D}$ as follows:

$$\mathcal{D} = \bigcup_{x\in\mathcal{X}} \bigcup_{T\in\mathcal{T}} \left\{ \left(x, y^{(i)}, r_T(x, y^{(i)}) \right) \right\}_{i=1}^{N}, \quad \text{where } y^{(i)} \sim \pi_{\text{ref}}(\cdot \mid x; T), \tag{7}$$

r_T is the best-performing reward at temperature T. We defer our implementation of $\mathcal{D}$ to Section 4.2 since the optimal design choice is contingent on empirical evaluation. However, the final reward functions (one continuous and one binary) achieve consistent correlations with controllability metrics and deliver robust model improvements across conditional generation tasks (Tab. 10) and model scales (Tab. 1), highlighting the generality and effectiveness of our task-agnostic design.

3.3 Soft Reward Optimization (SRO)

Write $\rho_\theta(x, y) = \pi_\theta(y|x)/\pi_{\text{ref}}(y|x)$ and $\mathbb{E}_{\text{ref}} = \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{y\sim\pi_{\text{ref}}(\cdot|x)}$. Substituting Eq. (4) into Theorem 3.1 gives

$$\min_{\pi_\theta} \mathcal{L}_{\text{KL}} \equiv \min_{\pi_\theta} \mathbb{E}_{\text{ref}}[\rho_\theta \left(\log \rho_\theta - r/\beta\right)] \tag{8}$$

$$\equiv \min_{\pi_\theta} \mathbb{E}_{\text{ref}}[\rho_\theta(r_\theta - r)/\beta], \tag{9}$$

where the policy-independent term $\mathbb{E}_x \log Z(x)$ is dropped in Eq. (8) or added in Eq. (9), and the implicit reward (Rafailov et al., 2023) is

$$r_\theta(x, y) = \beta \log \rho_\theta(x, y) + \beta \log Z(x). \tag{10}$$

Since Eq. (9) can be interpreted as the KL-divergence between the distributions induced by the implicit reward $r_\theta(x, y)$ and explicit reward $r(x, y)$, prior methods (Widatalla et al., 2024; Stocco et al., 2024) approximate it using batch-normalized distributions $\exp(r_\theta(x, y^i))/\sum_i \exp(r_\theta(x, y^i))$ and $\exp(r(x, y^i))/\sum_i \exp(r(x, y^i))$ up to a scaling factor β. *To circumvent the batch effect*[5], we instead propose to directly estimate the RHS of Eq. (8) on $\mathcal{D}$ via importance-weighted Monte Carlo sampling:

$$\begin{aligned} \mathcal{L}_{\text{SRO}} &= \mathbb{E}_{\text{ref}}[\rho_\theta(\log \rho_\theta - r/\beta - \alpha)] \\ &\simeq \frac{1}{|\mathcal{D}|} \sum_{i=1}^{|\mathcal{D}|} \rho_\theta^i \left(\log \rho_\theta^i - r^i/\beta - \alpha\right), \end{aligned} \tag{11}$$

where $\rho_\theta^i = \rho_\theta(x^i, y^i)$, $r^i = r(x^i, y^i)$, and α is a multiplier that constrains the overall probability mass of π_θ on $\mathcal{D}$. Eq. (11) operates as the objective of our proposed **soft reward optimization (SRO)**.

[5] In Tab. 11, we show that SRO variants substantially outperform this "weighted DPO" introduced by (Widatalla et al., 2024; Stocco et al., 2024).

Theorem 3.2. *Given a reward function $r(x, y)$, SRO in Eq. (11) has a closed-form solution*

$$\pi_\theta^* = \pi_{\text{ref}} \exp\left(\frac{r(x,y)}{\beta} + \alpha - 1\right). \tag{12}$$

For implementation, we sample a group of responses $\{y^j\}_{j=1}^G$ for each prompt x^i with π_{ref}. We *normalize*[6] the reward as an advantage function following RLVR methods like GRPO (Shao et al., 2024)

$$\hat{r}(x,y) = \frac{r(x,y) - \text{mean}_y(r(x,y))}{\text{std}_y(r(x,y))}. \tag{13}$$

By Theorem 3.2, we choose $\alpha = 1$ to ensure that $\pi_\theta^*(y|x) = \pi_{\text{ref}}(y|x)$ for y that receives zero (average) reward[7].

3.4 Binarized Reward Optimization (BRO)

In practice, as a proxy for ground-truth labels, the reward signal $r(x, y)$ can be noisy and inaccurate (see Section 4.2), which may render SRO unreliable. To this end, previous methods such as DPO (Rafailov et al., 2023) circumvent the need for explicit reward by adopting the Bradley-Terry formalism (Bradley & Terry, 1952) for preference distribution modeling. However, DPO and its variants require strictly ranked samples. As a more general alternative, we derive a novel optimization objective by leveraging *unpaired* binary rewards introduced by KTO (Ethayarajh et al., 2024). Specifically, we assume that instead of assigning a continuous score, the generated response y can be classified as positive ($y \in \mathcal{D}^+$) or negative ($y \in \mathcal{D}^-$) in terms of alignment with its prompt x. For **binarized reward optimization (BRO)**, let $b = \mathbb{1}\{y \in \mathcal{D}^+\}$ and $\pi_{\theta,\text{BRO}} = \sigma(\log \rho_\theta) = \rho_\theta/(1 + \rho_\theta)$. Minimizing the Bernoulli KL from the binary target b yields the logistic regression loss

$$\mathcal{L}_{\text{BRO}} = -\mathbb{E}_{(x,y)\sim\mathcal{D}}\left[b \log \frac{\rho_\theta}{1+\rho_\theta} + (1-b)\log\frac{1}{1+\rho_\theta}\right]. \tag{14}$$

The intermediate reward-space formulation is given in Appendix B.3. From a game-theoretic view, the reward maximization problem induced by $\mathcal{L}_{\text{BRO}}$ in Eq. 14 can be interpreted as a zero-sum game where the learned policy π_θ competes against the reference policy π_{ref} over a labeled dataset $\mathcal{D} = \mathcal{D}^+ \cup \mathcal{D}^-$. For a detailed derivation, please see Appendix B.3.

4 Experiments

We present an empirical study of the proposed *unsupervised reward optimization* framework for PLMs through two main experimental components: (1) evaluating the proposed statistical metrics in Section 3.2 for reward labeling, and (2) benchmarking PLMs fine-tuned by SRO/BRO on novel compositional OOD prompt generalization tasks with rewards from part (1). Our proposed methods show competitive performance despite being trained with noisy supervision, and in certain regimes achieve performance comparable to that of the supervised model fine-tuned with ground-truth labels.

4.1 Experimental Setup

Benchmarking the intrinsic and extrinsic rewards proposed in Section 3.2 requires (1) a prompt set $\mathcal{X}$ as conditions, (2) a base model π_{ref} to sample protein sequences $\mathcal{Y}$, (3) a representation model q to compute semantic distance, and (4) a ground-truth labeler $l : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ for evaluating alignment with the prompt. To this end, we consider two frontier PLMs with customized setups:

• **ProGen2** (Nijkamp et al., 2023): An autoregressive PLM pre-trained over one billion natural protein sequences and can be finetuned to generate highly specialized functional protein by instruction

[6] In principle, this normalization is valid for algorithms like SRO which use global labels rather than groupwise or pairwise labels, *only if* the overall and the promptwise reward statistics are consistent (e.g., exhibit similar high correlation with the ground-truth label), which is verified in Section 4.2.

[7] Note that $\alpha = 1$ does not strictly preserve the probability simplex constraint $\sum_y \pi_\theta(y|x) = 1$ since we are making estimations on a finite, offline dataset. This choice empirically proves to be robust and effective in our experiments.

tuning. We use a 151M version called progen2-small-mix7 as the primary reference model, which is fine-tuned by Hrbáň & Hoksza (2024) via SFT on a dataset of 7 Pfam (Finn et al., 2014) protein families. For a detailed description of these Pfam proteins, please see Appendix C. The SFT prompts consist of special family tokens like <|pf00002|>. Following ESM3 (Hayes et al., 2025), we apply InterProScan (Paysan-Lafosse et al., 2023) as the ground-truth labeler by letting $l(x, y) = 1$ if the Pfam family keyword is recovered by InterProScan and 0 otherwise. We also train a 764M progen2-medium-mix7 model via SFT on the same dataset for scaling experiments.

• **ESM3** (Hayes et al., 2025): A state-of-the-art multimodal PLM with three tracks (sequence, structure and function) and pretrained via masked language modeling in Eq. (1). Due to its multimodal nature, ESM3 can perform diverse conditional generation tasks with customized functional and structural primitives. For Func2Seq tasks, we follow the same procedure as ProGen2 and use the keyword (InterPro entries) recovery rate by InterProScan as the ground-truth label. For Struct2Seq tasks, namely inverse folding (Hsu et al., 2022; Dauparas et al., 2022), we predict the structure of generated sequence by ESMfold (Lin et al., 2023) and compute the backbone cRMSD against the prompted structure. The target structure is considered recovered if the cRMSD < 2 Å. The only open-weight model, 1.4B esm3-sm-open-v1, is used in this study.

Prompt and Dataset Construction Since no prior work considers the exact setting of unsupervised reward optimization for PLMs, we introduce novel prompt sets for biologically meaningful Func2Seq and Struct2Seq tasks. For ESM3, we randomly sample 1000 proteins from the *Dracunculus medinensis* (**DRAME**)[8] subset of AlphaFoldDB (Varadi et al., 2022), and employ their InterPro entries and 3D structures as the Func2Seq and Struct2Seq prompts respectively. The ESM3 prompt sets are used for benchmarking unsupervised rewards only. For ProGen2, based on the 7 Pfam control tags of progen2-small-mix7, we concatenate each family token with the first 10 residues of sequences from *another* family to construct a novel OOD prompt set. For fine-tuning, the training and testing set comprise 1400 (**Pfam1400**) and 700 (**Pfam700**) prompts in total, with 200 and 100 entries from each Pfam family. These prompts are used for both reward benchmarking and BRO/SRO post-training to evaluate the *compositional generalization* (Kirk et al., 2023; Yang et al., 2024) of our proposed framework[9]. Our full datasets $\mathcal{D}$ are sampled over five temperatures $T \in \{0.3, 0.5, 0.7, 1.0, 1.5\}$ according to Eq. (7) with $N = 64$, and therefore contain $448k$ and $224k$ samples, respectively. We apply a state-of-the-art representation PLM ESMC-600M (ESM Team, 2024) for generating protein embeddings and extrinsic rewards. An alternative model is also assessed and produces consistent results, see details in Appendix E.

4.2 Benchmarking Unsupervised Rewards

We examine our proposed unsupervised rewards in Section 3.2 on three combinations of prompt sets and tasks: Pfam700-Func2Seq, DRAME-Func2Seq and DRAME-Struct2Seq. For extrinsic rewards defined in Eq. (6), we evaluate a broad spectrum of metrics d (see details in Appendix E) and select two representatives for illustration in Figure 1: L1-mean and PCA1-bos.

For the Func2Seq tasks shown in Figure 1(a) and (b), all metrics exhibit consistent trends across different PLMs and prompt sets. The entropy-based intrinsic rewards increase monotonically in temperature, starting from strongly negative correlation with the ground truth at $T \leq 0.7$ and drastically reaching high positive correlation at $T > 1.0$. This suggests that both ProGen2 and ESM3 models are *over-confident* at low temperatures, leading to degenerate and repetitive sequences with high probability (Welleck et al., 2020; Holtzman et al., 2020) and *under-confident* at high temperatures, generating incoherent and random outputs. Intriguingly, as observed in Tab. 1 and appendix H, the "critical temperature" at which the models transition from being over-confident to under-confident, which is $T \sim 0.7$ for ProGen2 and $T \sim 1.0$ for ESM3, coincides with the best-performing temperature. However, at this critical temperature, intrinsic rewards are close to random, offering no predictive power.

Compared to predictive and normalized entropy, the extrinsic rewards exhibit more nuanced patterns. For PCA1-bos, the projection is up to a sign, which we always choose for it to be positively correlated

[8]We chose this dataset to prevent data leakage as *Dracunculus medinensis* was added to AlphaFoldDB since v5 release, whereas ESM3 reported using v4 for training.

[9]Biologically, this setup can be interpreted to test the model's controllability as well as creativity in "reprogramming" functional context from one protein family onto the N-terminal scaffold of another.

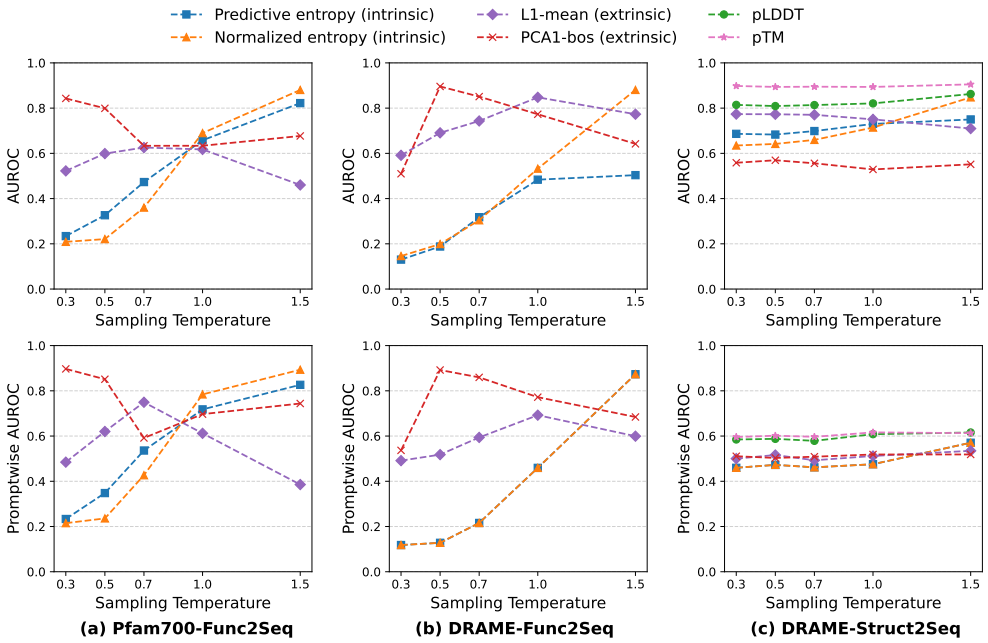

Figure 1: **Correlations between intrinsic/extrinsic rewards and the ground-truth label in terms of AUROC** on (a) Pfam700 prompt set for Func2Seq task, and DRAME prompt set for (b) Func2Seq and (c) Struct2Seq tasks, across five sampling temperatures. Promptwise AUROC evaluates the correlation within the generated sequences for each prompt.

with the ground truth[10]. Therefore, the AUROCs for PCA1-bos are always above 0.5, while generally decreasing in temperature. In contrast, L1-mean always peaks at the critical temperature, making it a *great complement* to other metrics, especially the intrinsic uncertainty measures.

For Struct2Seq tasks in Figure 1(c), we also consider pLDDT and pTM (Jumper et al., 2021; Lin et al., 2023) from ESMfold prediction as competitive baselines. Compared to Func2Seq tasks, all proposed metrics demonstrate positive correlations with the ground truth and much more stable trends w.r.t. temperature. Despite being more computationally efficient[11], they underperform relative to pLDDT and pTM. Moreover, unlike in Func2Seq tasks, the promptwise correlations for all metrics, including pLDDT and pTM, deteriorate markedly. We interpret this as evidence that correlations in Struct2Seq tasks are largely prompt-dependent: some structures are inherently easier to recover than others. This also implies that, for many prompts, all generated sequences are infeasible, as we empirically confirm, leaving few positive samples for self-improvement. As a result, we exclude Struct2Seq tasks from the subsequent post-training study, while retaining them as diagnostic evidence for our task-agnostic reward design.

The temperature-dependent piecewise reward r_T and the associated sample-selection procedure are detailed in Appendix E.

4.3 Compositional OOD Prompt Generalization

To demonstrate the effectiveness of our proposed framework, we fine-tune two SFT models, Progen2-small-mix7 and Progen2-medium-mix7, via SRO/BRO (ours) and two competitive baselines (DPO (Rafailov et al., 2023) and KTO (Ethayarajh et al., 2024)) with the unsupervised rewards in Eq. (28). For binary rewards used by BRO, DPO and KTO, we find that adopting L1-mean alone regardless of T produces robust improvements for all post-training methods (Tab. 2), which we report

[10]We empirically observe that at lower T, PCA1 effectively separates generated sequences into feasible-looking ones and degenerate gibberish, which aligns with a recent study (Prabakaran & Bromberg, 2026). Therefore, the sign of PCA can be determined by assigning the projection score of the non-biological sequences as negative, without any knowledge of ground-truth labels.

[11]For our experiments with a single A40 GPU, the average inference time per sequence for ESMC-600M is less than 0.1s, whereas ESMfold takes more than 2.0s.

here. As supervised baselines, we also train base models with ground-truth reward via BRO, since the labels are inherently binary.

Table 1: **Keyword recovery on Pfam700 Func2Seq tasks**. We report pass@1 success rate with top-k decoding, where $k = 15$. For each temperature, the best model except Supervised is **bolded** and the second best is underlined. More details can be found in Appendix G.

	$T = 0.3$	$T = 0.5$	$T = 0.7$	$T = 1.0$	$T = 1.5$	Average
151M model						
Progen2-small-mix7	0.210	0.288	0.390	0.401	0.118	0.281
Progen2-small-mix7 DPO	0.283	0.395	**0.512**	0.435	0.122	0.350
Progen2-small-mix7 KTO	0.272	0.347	0.430	0.458	0.175	0.337
Progen2-small-mix7 BRO	**0.406**	0.445	0.494	**0.520**	0.298	**0.433**
Progen2-small-mix7 SRO	0.383	**0.455**	0.498	0.499	**0.301**	0.427
Progen2-small-mix7 Supervised	0.537	0.573	0.579	0.506	0.247	0.488
764M model						
Progen2-medium-mix7	0.383	0.471	0.567	0.428	0.074	0.385
Progen2-medium-mix7 DPO	0.591	0.693	0.670	0.379	0.067	0.480
Progen2-medium-mix7 KTO	0.507	0.620	0.681	0.557	0.159	0.505
Progen2-medium-mix7 BRO	**0.678**	0.697	0.701	0.645	**0.342**	**0.613**
Progen2-medium-mix7 SRO	0.655	**0.706**	**0.708**	**0.648**	0.292	0.602
Progen2-medium-mix7 Supervised	0.661	0.700	0.704	0.642	0.308	0.603

Shown in Tabs. 1 and 10, both SRO and BRO significantly improve the generalization of the base model on the challenging compositional OOD prompt set **Pfam700** across temperatures, model scales and protein families. They outperform DPO and KTO by a substantial margin and, in some regimes, achieve performance comparable to the Supervised baseline. For example, for the 764M base model, BRO attains an average recovery rate of 61.3%, compared with 60.3% for the Supervised baseline. Moreover, we conduct pass@k analysis (Chen et al., 2021) in Figure 2 to further dissect the nature of the improvements conferred by BRO and SRO. We observe that they generally boost pass@k scores for small k yet converge with the base model at high k values. This implies that BRO/SRO incentivize the model to prioritize and consistently generate patterns that align with the prompts, rather than instilling fundamentally novel ones, reconfirming recent studies on post-training reasoning LLMs (Zhang et al., 2025a; Wu et al., 2024; Yue et al., 2025; Song et al., 2025a). However, DPO, despite underperforming BRO/SRO at small k, exhibits stronger performance compared to the base model for nearly all k values, suggesting that it not only enhances the probability of successful generations, but also pushes the boundary of the model's compositional generalization capabilities. Overall, these observations highlight the effectiveness and potential of our refined strategy for reward labeling, data sampling and algorithm design.

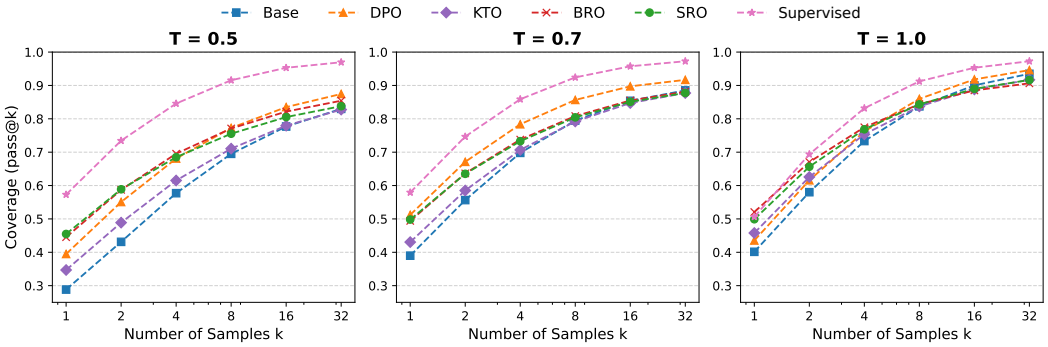

Figure 2: **Pass@k curves for Progen2-small-mix7 on Pfam700 Func2Seq tasks.** For each prompt, $N = 64$ sequences are sampled for evaluation.

Appendix I provides additional best-of-N comparisons, ESM3 adaptation results, sequence-diversity analyses, and DPO/KTO sensitivity to β.

Ethics Statement

Our study consists of computational protein generation and retrospective evaluation using existing protein datasets and pretrained models. It does not report new human-subject studies, animal experiments, or experimental testing of generated proteins. The proposed proxy rewards measure model-derived properties of generated sequences; they do not assess toxicity, immunogenicity, or other safety properties. Consequently, our results should not be interpreted as establishing the safety of generated sequences. Reward computation uses model-derived scores, while the pretrained models, reward diagnostics, and evaluation benchmarks may incorporate supervised information.

Reproducibility Statement

Section 3 defines the proxy rewards and the SRO and BRO objectives, and Section 4 describes the benchmarks, reward diagnostics, and fine-tuning evaluations. Appendix B provides the theoretical derivations, Appendix C specifies the prompt sets, and Appendix G documents training and evaluation settings. Appendix I provides additional comparisons and their evaluation protocols.

AI Use Statement

We also used an AI assistant for polishing writing, literature search, and assistance with plotting and dataset inspection.

References

Shivam Agarwal, Zimin Zhang, Lifan Yuan, Jiawei Han, and Hao Peng. The unreasonable effectiveness of entropy minimization in LLM reasoning. In *Thirty-ninth Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=UfFTBEsLgI. Poster Presentation.

Syeda Nahida Akter, Shrimai Prabhumoye, Matvei Novikov, Seungju Han, Ying Lin, Evelina Bakhturina, Eric Nyberg, Yejin Choi, Mostofa Patwary, Mohammad Shoeybi, et al. Nemotron-crossthink: Scaling self-learning beyond math reasoning. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 984–1002, 2026.

Ethan C Alley, Grigory Khimulya, Surojit Biswas, Mohammed AlQuraishi, and George M Church. Unified rational protein engineering with sequence-based deep representation learning. *Nature methods*, 16(12):1315–1322, 2019.

Minkyung Baek, Frank DiMaio, Ivan Anishchenko, Justas Dauparas, Sergey Ovchinnikov, Gyu Rie Lee, Jue Wang, Qian Cong, Lisa N Kinch, R Dustin Schaeffer, et al. Accurate prediction of protein structures and interactions using a three-track neural network. *Science*, 373(6557):871–876, 2021.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

Ralph Allan Bradley and Milton E Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

Collin Burns, Pavel Izmailov, Jan Hendrik Kirchner, Bowen Baker, Leo Gao, Leopold Aschenbrenner, Yining Chen, Adrien Ecoffet, Manas Joglekar, Jan Leike, et al. Weak-to-strong generalization: Eliciting strong capabilities with weak supervision. In *Forty-first International Conference on Machine Learning*.

Bo Chen, Xingyi Cheng, Pan Li, Yangli-ao Geng, Jing Gong, Shen Li, Zhilei Bei, Xu Tan, Boyan Wang, Xin Zeng, et al. xtrimopglm: unified 100-billion-parameter pretrained transformer for deciphering the language of proteins. *Nature Methods*, 22(5):1028–1039, 2025.

Jiayang Chen, Zhihang Hu, Siqi Sun, Qingxiong Tan, Yixuan Wang, Qinze Yu, Licheng Zong, Liang Hong, Jin Xiao, Tao Shen, et al. Interpretable rna foundation model from unannotated data for highly accurate rna structure and function predictions. *arXiv preprint arXiv:2204.00300*, 2022.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

Ganqu Cui, Yuchen Zhang, Jiacheng Chen, Lifan Yuan, Zhi Wang, Yuxin Zuo, Haozhan Li, Yuchen Fan, Huayu Chen, Weize Chen, et al. The entropy mechanism of reinforcement learning for reasoning language models. *arXiv preprint arXiv:2505.22617*, 2025.

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.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pp. 4171–4186, 2019.

Qingxiu Dong, Li Dong, Yao Tang, Tianzhu Ye, Yutao Sun, Zhifang Sui, and Furu Wei. Reinforcement pre-training. *arXiv preprint arXiv:2506.08007*, 2025.

Ahmed Elnaggar, Michael Heinzinger, Christian Dallago, Ghalia Rehawi, Yu Wang, Llion Jones, Tom Gibbs, Tamas Feher, Christoph Angerer, Martin Steinegger, Debsindhu Bhowmik, and Burkhard Rost. Prottrans: Toward understanding the language of life through self-supervised learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):7112–7127, 2022. doi: 10.1109/TPAMI.2021.3095381.

ESM Team. Esm cambrian: Revealing the mysteries of proteins with unsupervised learning, 2024. URL https://evolutionaryscale.ai/blog/esm-cambrian.

Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. Model alignment as prospect theoretic optimization. In *International Conference on Machine Learning*, pp. 12634–12651. PMLR, 2024.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017):625–630, 2024.

Noelia Ferruz and Birte Höcker. Controllable protein design with language models. *Nature Machine Intelligence*, 4(6):521–532, 2022.

Robert D Finn, Alex Bateman, Jody Clements, Penelope Coggill, Ruth Y Eberhardt, Sean R Eddy, Andreas Heger, Kirstie Hetherington, Liisa Holm, Jaina Mistry, et al. Pfam: the protein families database. *Nucleic acids research*, 42(D1):D222–D230, 2014.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638, 2025.

Tuomas Haarnoja, Haoran Tang, Pieter Abbeel, and Sergey Levine. Reinforcement learning with deep energy-based policies. In *International conference on machine learning*, pp. 1352–1361. PMLR, 2017.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*, pp. 1861–1870. Pmlr, 2018.

Ali Hatamizadeh, Syeda Nahida Akter, Shrimai Prabhumoye, Jan Kautz, Mostofa Patwary, Mohammad Shoeybi, Bryan Catanzaro, and Yejin Choi. Rlp: Reinforcement as a pretraining objective. *arXiv preprint arXiv:2510.01265*, 2025.

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

Bingxiang He, Yuxin Zuo, Zeyuan Liu, Shangziqi Zhao, Zixuan Fu, Junlin Yang, Cheng Qian, Kaiyan Zhang, Yuchen Fan, Ganqu Cui, Xiusi Chen, Youbang Sun, Xingtai Lv, Xuekai Zhu, Li Sheng, Ran Li, Huan ang Gao, Yuchen Zhang, Lifan Yuan, Bowen Zhou, et al. How far can unsupervised RLVR scale LLM training? In *The Thirteenth International Conference on Learning Representations*, 2026.

Michael Heinzinger, Ahmed Elnaggar, Yu Wang, Christian Dallago, Dmitrii Nechaev, Florian Matthes, and Burkhard Rost. Modeling aspects of the language of life through transfer-learning protein sequences. *BMC bioinformatics*, 20(1):723, 2019.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration. In *International Conference on Learning Representations*, 2020.

Hugo Hrbáň and David Hoksza. Protein family sequence generation through progen2 fine-tuning. In *2024 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pp. 7037–7039, 2024. doi: 10.1109/BIBM62325.2024.10821712.

Chloe Hsu, Robert Verkuil, Jason Liu, Zeming Lin, Brian Hie, Tom Sercu, Adam Lerer, and Alexander Rives. Learning inverse folding from millions of predicted structures. In *International conference on machine learning*, pp. 8946–8970. PMLR, 2022.

Long-Kai Huang, Rongyi Zhu, Bing He, and Jianhua Yao. Steering protein language models. In *International Conference on Machine Learning*, pp. 26247–26260. PMLR, 2025.

Thomas Hubert, Rishi Mehta, Laurent Sartran, Miklós Z Horváth, Goran Žužić, Eric Wieser, Aja Huang, Julian Schrittwieser, Yannick Schroecker, Hussain Masoom, et al. Olympiad-level formal mathematical reasoning with reinforcement learning. *Nature*, pp. 1–3, 2025.

Kaiyi Jiang, Zhaoqing Yan, Matteo Di Bernardo, Samantha R Sgrizzi, Lukas Villiger, Alisan Kayabolen, BJ Kim, Josephine K Carscadden, Masahiro Hiraizumi, Hiroshi Nishimasu, et al. Rapid in silico directed evolution by a protein language model with evolvepro. *Science*, 387(6732): eadr6006, 2024.

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.

Robert Kirk, Amy Zhang, Edward Grefenstette, and Tim Rocktäschel. A survey of zero-shot generalisation in deep reinforcement learning. *Journal of Artificial Intelligence Research*, 76: 201–264, 2023.

Tomasz Korbak, Kejian Shi, Angelica Chen, Rasika Vinayak Bhalerao, Christopher Buckley, Jason Phang, Samuel R Bowman, and Ethan Perez. Pretraining language models with human preferences. In *International conference on machine learning*, pp. 17506–17533. PMLR, 2023.

Tanja Kortemme. De novo protein design—from new structures to programmable functions. *Cell*, 187(3):526–544, 2024.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *The Eleventh International Conference on Learning Representations*, 2023.

Sarah Lewis, Tim Hempel, José Jiménez-Luna, Michael Gastegger, Yu Xie, Andrew YK Foong, Victor García Satorras, Osama Abdin, Bastiaan S Veeling, Iryna Zaporozhets, et al. Scalable emulation of protein equilibrium ensembles with generative deep learning. *Science*, 389(6761): eadv9817, 2025.

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.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Liuzhenghao Lv, Zongying Lin, Hao Li, Yuyang Liu, Jiaxi Cui, Calvin Yu-Chian Chen, Li Yuan, and Yonghong Tian. Prollama: A protein large language model for multi-task protein language processing. *IEEE Transactions on Artificial Intelligence*, 2025.

Ali Madani, Ben Krause, Eric R Greene, Subu Subramanian, Benjamin P Mohr, James M Holton, Jose Luis Olmos Jr, 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.

Joshua Meier, Roshan Rao, Robert Verkuil, Jason Liu, Tom Sercu, and Alex Rives. Language models enable zero-shot prediction of the effects of mutations on protein function. *Advances in neural information processing systems*, 34:29287–29303, 2021.

Shentong Mo and Lanqing Li. Sadit: Efficient protein backbone design via latent structural tokenization and diffusion transformers. *arXiv preprint arXiv:2602.06706*, 2026.

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

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.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

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

Typhaine Paysan-Lafosse, Matthias Blum, Sara Chuguransky, Tiago Grego, Beatriz Lázaro Pinto, Gustavo A Salazar, Maxwell L Bileschi, Peer Bork, Alan Bridge, Lucy Colwell, et al. Interpro in 2022. *Nucleic acids research*, 51(D1):D418–D427, 2023.

R Prabakaran and Yana Bromberg. Quantifying uncertainty in protein representations across models and tasks. *Nature Methods*, pp. 1–9, 2026.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741, 2023.

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.

Vishwanatha M Rao, Serena Zhang, Brian S Plosky, Patrick D Hsu, Bo Wang, James Zou, Marinka Zitnik, Eric J Topol, and Pranav Rajpurkar. Generalist biological artificial intelligence in modeling the language of life. *Nature Biotechnology*, pp. 1–16, 2026.

Alexander Rives, Joshua Meier, Tom Sercu, Siddharth Goyal, Zeming Lin, Jason Liu, Demi Guo, Myle Ott, C. Lawrence Zitnick, Jerry Ma, and Rob Fergus. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *PNAS*, 2019. doi: 10.1101/622803. URL https://www.biorxiv.org/content/10.1101/622803v4.

Sheikh Shafayat, Fahim Tajwar, Ruslan Salakhutdinov, Jeff Schneider, and Andrea Zanette. Can large reasoning models self-train? *arXiv preprint arXiv:2505.21444*, 2025.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Zhihong Shao, Yuxiang Luo, Chengda Lu, ZZ Ren, Jiewen Hu, Tian Ye, Zhibin Gou, Shirong Ma, and Xiaokang Zhang. Deepseekmath-v2: Towards self-verifiable mathematical reasoning. *arXiv preprint arXiv:2511.22570*, 2025.

Shuaijie She, Yu Bao, Yu Lu, Lu Xu, Tao Li, Wenhao Zhu, Shujian Huang, Shanbo Cheng, Lu Lu, and Yuxuan Wang. Dupo: Enabling reliable llm self-verification via dual preference optimization. *arXiv preprint arXiv:2508.14460*, 2025.

David Silver and Richard S Sutton. Welcome to the era of experience. *Google AI*, 1:11, 2025.

Toby Simonds and Akira Yoshiyama. Ladder: Self-improving llms through recursive problem decomposition. *arXiv preprint arXiv:2503.00735*, 2025.

Yuda Song, Hanlin Zhang, Carson Eisenach, Sham M Kakade, Dean Foster, and Udaya Ghai. Mind the gap: Examining the self-improvement capabilities of large language models. In *The Thirteenth International Conference on Learning Representations*, 2025a.

Zhangde Song, Jieyu Lu, Yuanqi Du, Botao Yu, Thomas M Pruyn, Yue Huang, Kehan Guo, Xiuzhe Luo, Yuanhao Qu, Yi Qu, et al. Evaluating large language models in scientific discovery. *arXiv preprint arXiv:2512.15567*, 2025b.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. Learning to summarize with human feedback. *Advances in neural information processing systems*, 33:3008–3021, 2020.

Filippo Stocco, Maria Artigues-Lleixa, Andrea Hunklinger, Talal Widatalla, Marc Guell, and Noelia Ferruz. Guiding generative protein language models with reinforcement learning, 2024. URL https://arxiv.org/abs/2412.12979.

Jin Su, Chenchen Han, Yuyang Zhou, Junjie Shan, Xibin Zhou, and Fajie Yuan. Saprot: Protein language modeling with structure-aware vocabulary. In *The Twelfth International Conference on Learning Representations*, 2024.

Qwen Team. Qwq-32b: Embracing the power of reinforcement learning, 2025. URL https://qwenlm.github.io/blog/qwq-32b/.

Mihaly Varadi, Stephen Anyango, Mandar Deshpande, Sreenath Nair, Cindy Natassia, Galabina Yordanova, David Yuan, Oana Stroe, Gemma Wood, Agata Laydon, et al. Alphafold protein structure database: massively expanding the structural coverage of protein-sequence space with high-accuracy models. *Nucleic acids research*, 50(D1):D439–D444, 2022.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

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.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*, 2022.

Sean Welleck, Ilia Kulikov, Stephen Roller, Emily Dinan, Kyunghyun Cho, and Jason Weston. Neural text generation with unlikelihood training. In *International Conference on Learning Representations*, 2020.

Talal Widatalla, Rafael Rafailov, and Brian Hie. Aligning protein generative models with experimental fitness via direct preference optimization. *bioRxiv*, pp. 2024–05, 2024.

Adam Winnifrith, Carlos Outeiral, and Brian L Hie. Generative artificial intelligence for de novo protein design. *Current Opinion in Structural Biology*, 86:102794, 2024.

Zhengxuan Wu, Aryaman Arora, Zheng Wang, Atticus Geiger, Dan Jurafsky, Christopher D Manning, and Christopher Potts. Reft: Representation finetuning for language models. *Advances in Neural Information Processing Systems*, 37:63908–63962, 2024.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Haoran Yang, Hongyuan Lu, Wai Lam, and Deng Cai. Exploring compositional generalization of large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 4: Student Research Workshop)*, pp. 16–24, 2024.

Kevin K Yang, Zachary Wu, and Frances H Arnold. Machine-learning-guided directed evolution for protein engineering. *Nature methods*, 16(8):687–694, 2019.

Wei Yang, Shunzhi Wang, Gyu Rie Lee, Jason Z Zhang, Alexis Courbet, David Juergens, Xinru Wang, Thomas Schlichthaerle, Mohamad Abedi, Robert Ragotte, et al. The past, present and future of de novo protein design. *Nature*, 652(8112):1139–1152, 2026.

Yang Yue, Zhiqi Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Shiji Song, and Gao Huang. Does reinforcement learning really incentivize reasoning capacity in llms beyond the base model? In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Qingyang Zhang, Haitao Wu, Changqing Zhang, Peilin Zhao, and Yatao Bian. Right question is already half the answer: Fully unsupervised llm reasoning incentivization. In *Thirty-ninth Conference on Neural Information Processing Systems*, 2025a. URL https://openreview.net/forum?id=k8Mim6RI5O. Spotlight Presentation.

Shimao Zhang, Yu Bao, and Shujian Huang. Edt: Improving large language models' generation by entropy-based dynamic temperature sampling. *arXiv preprint arXiv:2403.14541*, 2024.

Yanzhi Zhang, Zhaoxi Zhang, Haoxiang Guan, Yilin Cheng, Yitong Duan, Chen Wang, Yue Wang, Shuxin Zheng, and Jiyan He. No free lunch: Rethinking internal feedback for llm reasoning. *arXiv preprint arXiv:2506.17219*, 2025b.

Andrew Zhao, Yiran Wu, Yang Yue, Tong Wu, Quentin Xu, Matthieu Lin, Shenzhi Wang, Qingyun Wu, Zilong Zheng, and Gao Huang. Absolute zero: Reinforced self-play reasoning with zero data. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Xuandong Zhao, Zhewei Kang, Aosong Feng, Sergey Levine, and Dawn Song. Learning to reason without external rewards. 2026. URL https://openreview.net/forum?id=OU9nFEYR2M.

Yuqi Zhu, Jia Li, Ge Li, YunFei Zhao, Zhi Jin, and Hong Mei. Hot or cold? adaptive temperature sampling for code generation with large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 437–445, 2024.

Brian D Ziebart, Andrew L Maas, J Andrew Bagnell, Anind K Dey, et al. Maximum entropy inverse reinforcement learning. In *Aaai*, volume 8, pp. 1433–1438. Chicago, IL, USA, 2008.

Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*, 2019.

Yuxin Zuo, Kaiyan Zhang, Li Sheng, Shang Qu, Ganqu Cui, Xuekai Zhu, Haozhan Li, Yuchen Zhang, Xinwei Long, Ermo Hua, Biqing Qi, Youbang Sun, Zhiyuan Ma, Lifan Yuan, Ning Ding, and Bowen Zhou. TTRL: Test-time reinforcement learning. In *Thirty-ninth Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=VuVhgEiu20. Poster Presentation.

A Discussion and Limitations

While our unsupervised reward optimization framework demonstrates strong performance in steering protein language models, several limitations remain. First, our unsupervised metrics, though effective for both ProGen2 and ESM3 on Func2Seq tasks, underperform structure-based baselines like pLDDT and pTM for inverse folding tasks (despite being much more computationally efficient as discussed in Section 4.2). Second, we only conduct *in silico* evaluations in this study, and wet-lab or clinical validations are required to demonstrate the real-world impact of our proposed framework in biological applications. Finally, the reliance on proxy rewards inherently introduces approximation error compared to supervised training, which may limit performance in domains where ground-truth labels are available (Tab. 1 and fig. 2). Future work may investigate more sophisticated reward design, uncertainty-aware optimization, and hybrid approaches that combine unsupervised signals with minimal human feedback.

Broader impacts. The proposed framework may support beneficial protein engineering by reducing reliance on costly experimental labels, potentially accelerating applications in enzyme design, therapeutics, vaccines, and biomaterials. At the same time, improved controllability of PLMs is a dual-use capability: similar techniques could be misused to search for proteins with harmful biological activity or to optimize pathogen-, toxin-, or immune-evasion-related functions. Since our work is evaluated only *in silico*, any deployment should be coupled with biosecurity screening, restrictions on sensitive targets, access control, request auditing, and institutional biosafety/ethics review.

B Proofs and Derivations

B.1 Proof of Theorem 3.1

Proof. Let

$$\mathcal{V}_{\pi_\theta} = \mathbb{E}_{x\sim\mathcal{D}, y\sim\pi_\theta(y|x)}[r(x,y)] - \beta D_{\mathrm{KL}}[\pi_\theta(y|x) \,\|\, \pi_{\mathrm{ref}}(y|x)] \tag{15}$$

$$\mathcal{V}^* \triangleq \mathbb{E}_{x\sim\mathcal{D}, y\sim\pi^*(y|x)}[r(x,y)] - \beta D_{\mathrm{KL}}[\pi^*(y|x) \,\|\, \pi_{\mathrm{ref}}(y|x)], \tag{16}$$

From the definition of π^* in Eq. 4, take log

$$\log \pi^*(y|x) = \log \pi_{\mathrm{ref}}(y|x) + \frac{r(x,y)}{\beta} - \log Z(x) \tag{17}$$

and rearrange

$$r(x,y) = \beta \log \frac{\pi^*(y|x)}{\pi_{\mathrm{ref}}(y|x)} + \beta \log Z(x). \tag{18}$$

Substitute into $\mathcal{V}_{\pi_\theta}$ in Eq. 15:

$$\begin{aligned}
\mathcal{V}_{\pi_\theta} &= \mathbb{E}_x \mathbb{E}_{y\sim\pi_\theta}\left[\beta \log \frac{\pi^*}{\pi_{\mathrm{ref}}} + \beta \log Z(x) - \beta \log \frac{\pi_\theta}{\pi_{\mathrm{ref}}}\right] \\
&= \beta \mathbb{E}_x \left[\log Z(x) - \mathbb{E}_{y\sim\pi_\theta(\cdot|x)}\left[\log \frac{\pi_\theta(y|x)}{\pi^*(y|x)}\right]\right] \\
&= \underbrace{\beta \mathbb{E}_x[\log Z(x)]}_{\mathcal{V}^*} - \beta \mathbb{E}_x[D_{\mathrm{KL}}(\pi_\theta(\cdot|x)||\pi^*(\cdot|x))]
\end{aligned} \tag{19}$$

Remark: The first term $\mathcal{V}^*$ is exactly the value of the optimal policy:

$$\mathcal{V}^* = \mathbb{E}_x \mathbb{E}_{y\sim\pi^*}\left[r(x,y) - \beta \log \frac{\pi^*}{\pi_{\text{ref}}}\right], \tag{20}$$

which can be verified by plugging π^* into the original objective in Eq. 16.

□

B.2 Proof of Theorem 3.2

Proof. We minimize the population objective

$$\mathcal{L}_{\text{SRO}} = \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{y\sim\pi_{\text{ref}}(y|x)}\left[\frac{\pi_\theta(x,y)}{\pi_{\text{ref}}(x,y)}\left(\log\frac{\pi_\theta(x,y)}{\pi_{\text{ref}}(x,y)} - \frac{r(x,y)}{\beta} - \alpha\right)\right],$$

where we write $\pi_\theta(x,y) = \pi_\theta(y|x)\, p_{\mathcal{D}}(x)$ and similarly for π_{ref}, so that the joint densities share the same marginal $p_{\mathcal{D}}(x)$. Since the expectation over x is fixed, we can optimize the integrand pointwise for each (x,y).

Define the pointwise loss for a fixed (x,y):

$$\ell(\pi_\theta) = \frac{\pi_\theta(x,y)}{\pi_{\text{ref}}(x,y)}\left(\log\frac{\pi_\theta(x,y)}{\pi_{\text{ref}}(x,y)} - \frac{r(x,y)}{\beta} - \alpha\right).$$

Let $u = \pi_\theta(x,y) > 0$ and treat $\pi_{\text{ref}}(x,y)$ as constant. Then

$$\ell(u) = \frac{u}{\pi_{\text{ref}}}\left(\log\frac{u}{\pi_{\text{ref}}} - \frac{r}{\beta} - \alpha\right).$$

Differentiate w.r.t. u:

$$\frac{d\ell}{du} = \frac{1}{\pi_{\text{ref}}}\left(\log\frac{u}{\pi_{\text{ref}}} - \frac{r}{\beta} - \alpha\right) + \frac{u}{\pi_{\text{ref}}}\cdot\frac{1}{u} = \frac{1}{\pi_{\text{ref}}}\left(\log\frac{u}{\pi_{\text{ref}}} - \frac{r}{\beta} - \alpha + 1\right).$$

Set derivative to zero for optimality:

$$\log\frac{u^*}{\pi_{\text{ref}}} - \frac{r}{\beta} - \alpha + 1 = 0 \quad\Rightarrow\quad \log\frac{u^*}{\pi_{\text{ref}}} = \frac{r}{\beta} + \alpha - 1.$$

Exponentiating both sides yields

$$u^* = \pi_{\text{ref}}(x,y)\exp\left(\frac{r(x,y)}{\beta} + \alpha - 1\right).$$

Since this holds for all (x,y), the optimal policy satisfies

$$\pi^*_\theta(x,y) = \pi_{\text{ref}}(x,y)\exp\left(\frac{r(x,y)}{\beta} + \alpha - 1\right),$$

which implies the conditional form

$$\pi^*_\theta(y|x) = \pi_{\text{ref}}(y|x)\exp\left(\frac{r(x,y)}{\beta} + \alpha - 1\right).$$

This completes the proof. □

B.3 Game-Theoretic View of Binarized Reward Optimization (BRO)

Binary-reward construction. Let $\mathcal{D}^+$ and $\mathcal{D}^-$ contain the positive and negative samples, respectively, and let $\rho_\theta = \pi_\theta/\pi_{\text{ref}}$ as in Section 3.3. This formulation, termed **binarized reward optimization (BRO)**, can be interpreted as an extreme case of $\mathcal{L}_{\text{SRO}}$ in the limit $\beta \to 0$:

$$\mathcal{L}_{\text{BRO}} \triangleq \lim_{\beta\to 0}\mathcal{L}_{\text{SRO}} = \mathbb{E}_{(x,y)\sim\mathcal{D}^+\cup\mathcal{D}^-}\left[\rho_\theta(\log\rho_\theta - r_{\text{BRO}})\right], \tag{21}$$

where $r_{\text{BRO}}(x, y) = +\infty$ for $y \in \mathcal{D}^+$ and $-\infty$ for $y \in \mathcal{D}^-$. Write $b = \mathbb{1}\{y \in \mathcal{D}^+\}$ for the binary label.

To make $\mathcal{L}_{\text{BRO}}$ tractable, inspired by pre-existing methods (Widatalla et al., 2024; Stocco et al., 2024) that approximate the two probabilities in Eq. (9) by their batch-normalized softmax variants, we propose optimizing their normalized variants in the binary label space:

$$\pi_{\theta,\text{BRO}} \triangleq \sigma(\log \rho_\theta) = \frac{\rho_\theta}{1+\rho_\theta}, \qquad \pi^*_{\text{BRO}} \triangleq \sigma(r_{\text{BRO}}) = b. \tag{22}$$

Optimizing $D_{\text{KL}}(\pi_{\theta,\text{BRO}} \,\|\, \pi^*_{\text{BRO}})$ as a proxy for $D_{\text{KL}}(\pi_\theta \,\|\, \pi^*)$ requires constraining the probability mass of π_θ on $\mathcal{D}$ as in Eq. (11). Alternatively, we propose to directly minimize $D_{\text{KL}}(\pi^*_{\text{BRO}} \,\|\, \pi_{\theta,\text{BRO}})$ instead:

$$\begin{aligned}\mathcal{L}_{\text{BRO}} &\simeq D_{\text{KL}}(\pi^*_{\text{BRO}} \| \pi_{\theta,\text{BRO}}) = H(\pi^*_{\text{BRO}}, \pi_{\theta,\text{BRO}}) \\ &= -\mathbb{E}_{(x,y)\sim\mathcal{D}}\left[b \log \frac{\rho_\theta}{1+\rho_\theta} + (1-b)\log\frac{1}{1+\rho_\theta}\right],\end{aligned} \tag{23}$$

since the binary target has zero entropy. This is precisely the logistic regression loss for binary classification.

Game-theoretic interpretation. From a game-theoretic view, the reward maximization problem induced by $\mathcal{L}_{\text{BRO}}$ in Eq. (14) can be interpreted as zero-sum game where the learned policy π_θ competes against the reference policy π_{ref} over a labeled dataset $\mathcal{D} = \mathcal{D}^+ \cup \mathcal{D}^-$. For each prompt-response pair (x, y), a *win* for π_θ is defined as:

- assigning higher relative likelihood than π_{ref} to positive samples ($y \in \mathcal{D}^+$), i.e., $\frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)} > 1$, and
- assigning lower relative likelihood than π_{ref} to negative samples ($y \in \mathcal{D}^-$), i.e., $\frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)} < 1$.

Under the BT model (Bradley & Terry, 1952), the probability that π_θ wins against π_{ref} on (x, y) is modeled via logistic pairwise comparison. Define the signed reward

$$s_{\text{BRO}}(x, y) = \begin{cases} +1, & y \in \mathcal{D}^+ \\ -1, & y \in \mathcal{D}^- \end{cases}.$$

Then the win probability is

$$P_{\text{win}}(x, y) = \sigma\left(s_{\text{BRO}}(x, y) \cdot (r_\theta(x, y) - r_{\text{ref}}(x, y))\right) \tag{24}$$

$$= \sigma\left(s_{\text{BRO}}(x, y) \cdot \beta \log \frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)}\right), \tag{25}$$

where $\sigma(z) = \frac{1}{1+e^{-z}}$ is the logistic sigmoid function.

Maximizing the expected winrate is equivalent to maximizing the log-likelihood of observed wins, which yields the BRO objective by letting $\beta = 1$:

$$\mathcal{L}_{\text{BRO}} = -\mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\log\sigma\left(s_{\text{BRO}}(x, y) \cdot \beta \log \frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)}\right)\right] \tag{26}$$

$$= -\mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\mathbb{1}\{y \in \mathcal{D}^+\} \log \frac{\pi_\theta}{\pi_{\text{ref}} + \pi_\theta} + \left(1 - \mathbb{1}\{y \in \mathcal{D}^+\}\right) \log \frac{\pi_{\text{ref}}}{\pi_{\text{ref}} + \pi_\theta}\right]. \tag{27}$$

Thus, minimizing $\mathcal{L}_{\text{BRO}}$ corresponds to optimizing π_θ to dominate π_{ref} in a zero-sum competition: π_θ seeks to increase its relative likelihood on $\mathcal{D}^+$ and decrease it on $\mathcal{D}^-$, with victory governed by a Bradley–Terry (BT) ranking where the "skill" of a policy on (x, y) is proportional to its log-probability $\log \pi(y|x)$ by Eq. (10). This frames policy improvement as maximizing the empirical winrate against the reference policy under pairwise logistic comparison.

C Details of Prompt Sets

Following (Hrbáň & Hoksza, 2024), the seven Pfam families used for our prompt sets are:

- PF00002 - GPCRs
- PF00042 - Globins
- PF00125 - Core histones
- PF00127 - Copper binding proteins
- PF00257 - Dehydrins
- PF00262 - Calreticulins
- PF03668 - P-loop ATPase

To construct compositional OOD prompt sets, for each family token, we append the first 10 residues of proteins from the next family on the list above to create 7 compositional prompt templates:

- `<|pf00002|>[first 10 tokens from seqs in PF00042]`
- `<|pf00042|>[first 10 tokens from seqs in PF00125]`
- `<|pf00125|>[first 10 tokens from seqs in PF00127]`
- `<|pf00127|>[first 10 tokens from seqs in PF00257]`
- `<|pf00257|>[first 10 tokens from seqs in PF00262]`
- `<|pf00262|>[first 10 tokens from seqs in PF03668]`
- `<|pf03668|>[first 10 tokens from seqs in PF00002]`

D Ablation Study

We provide comparative analysis of various design choices of reward in Tab. 2. We conclude that BRO and SRO achieve consistent improvement over the base model with different reward functions across all temperatures, demonstrating its robustness to noisy supervision signals. Among which, BRO + L1-mean and SRO + r_T bring the most significant gain, which are reported in Tab. 1.

Table 2: BRO/SRO with different choices of rewards for fine-tuning Progen2-small-mix7 on Pfam700 Func2Seq task.

Method	$T = 0.3$	$T = 0.5$	$T = 0.7$	$T = 1.0$	$T = 1.5$	Average
base	0.210	0.288	0.390	0.401	0.118	0.281
BRO (r_T)	0.327	0.406	0.470	0.453	0.276	0.386
BRO (L1, Tab. 1)	0.407	0.445	0.494	0.521	0.298	0.433
BRO (entropy)	0.249	0.323	0.414	0.453	0.266	0.341
BRO (PCA)	0.228	0.302	0.359	0.325	0.129	0.268
SRO (r_T, Tab. 1)	0.383	0.455	0.498	0.499	0.301	0.427
SRO (L1)	0.339	0.384	0.458	0.504	0.261	0.389
SRO (entropy)	0.269	0.350	0.434	0.480	0.287	0.364

Additionally, we present ablation on our proposed multi-temperature sampling strategy in Tab. 3. For BRO (L1, $T = 0.7$) and BRO (entropy, $T = 1.0$), we sample the same amount of data as the reported BRO (L1, multi-T) with 5$\times$ prompts and the best-performing metric at a single temperature according to Figure 1(a). BRO trained on multi-temperature data significantly outperforms its single-temperature counterparts, highlighting the effectiveness of the proposed reward design and sampling strategy in Section 3.2.

Table 3: Performance comparison across temperatures (values rounded to 3 significant figures).

Method	$T = 0.3$	$T = 0.5$	$T = 0.7$	$T = 1.0$	$T = 1.5$	Average
base	0.210	0.288	0.390	0.401	0.118	0.281
BRO (L1, multi-T)	0.407	0.445	0.494	0.521	0.298	0.433
BRO (L1, $T = 0.7$)	0.279	0.325	0.393	0.487	0.348	0.366
BRO (entropy, $T = 1.0$)	0.225	0.308	0.406	0.410	0.206	0.311

E More Experiments on Reward Design

Reward Selection. Based on the analysis in Section 4.2, we instantiate the multi-temperature sampling and reward-labeling strategy in Eq. (7) by defining a piecewise reward function r_T w.r.t. the critical temperature T_c:

$$r_T = \begin{cases} -\bar{d}_{\text{PCA1-bos}}, & \text{if } T < T_c \\ -\bar{d}_{\text{L1-mean}}, & \text{if } T = T_c \\ -\tilde{H}_\pi, & \text{if } T > T_c \end{cases} \tag{28}$$

which is motivated by the diagnostic correlations in Figure 1. See Appendix D for the ablation study of design choices. For each prompt, we sample $N = 64$ sequences and retain only the top-4 and bottom-4 for post-training to reduce reward noise. Moreover, this construction naturally produces positive and negative samples for BRO/DPO/KTO. More detail can be found in Appendix G.

Candidate Metrics. For our reward design, we report all candidate metrics considered in the following tables, evaluated with ESMC-600M and a smaller version, ESMC-300M[12], which demonstrate consistent correlations:

Table 4: Ground-truth AUROC on DRAME-Func2Seq with ESMC-300M embedding model.

Metric	**T=0**	**T=0.3**	**T=0.5**	**T=0.7**	**T=1.0**	**T=1.5**
cos-mean	0.3852	0.4820	0.4917	0.4996	0.6036	0.5218
cos-bos	0.3451	0.3922	0.4183	0.4899	0.6461	0.6585
cos-eos	0.2726	0.3101	0.3456	0.4205	0.5222	0.5303
euc-mean	0.4488	0.5937	0.6639	0.6931	0.8029	0.6606
euc-bos	0.3003	0.3461	0.3831	0.4639	0.6342	0.6555
euc-eos	0.2499	0.2948	0.3410	0.4216	0.5302	0.5626
$L2^2$-mean	0.5108	0.6469	0.6914	0.7051	0.7924	0.6470
$L2^2$-bos	0.3420	0.3883	0.4135	0.4833	0.6331	0.6519
$L2^2$-eos	0.2815	0.3232	0.3611	0.4373	0.5345	0.5612
L1-mean	0.4682	0.6228	0.7075	0.7432	0.8583	0.7420
L1-bos	0.2738	0.3174	0.3661	0.4701	0.6821	0.6615
L1-eos	0.2435	0.2892	0.3351	0.4156	0.5194	0.5546
Predictive Ent	0.1319	0.1302	0.1881	0.3177	0.4836	0.5037
Normalized Ent	0.1385	0.1459	0.1990	0.3041	0.5323	0.8805
Semantic Ent (Kuhn et al., 2023)	-	-	0.6390	0.6374	0.6763	0.4658
PCA-1-mean	0.6198	0.5294	0.6294	0.7808	0.7009	0.6463
PCA-1-bos	0.8937	0.8936	0.8728	0.8120	0.7099	0.5120
PCA-1-eos	0.8787	0.8747	0.8570	0.7956	0.6664	0.5732
PCA-2-mean	0.9114	0.9095	0.8796	0.7137	0.6202	-
PCA-2-bos	0.5678	0.5248	0.5395	0.5626	0.5333	-
PCA-2-eos	0.5263	0.5037	0.5399	0.5580	0.4298	-

[12]Naming rules: L1-mean measures the pairwise L1 distance between the mean per-residue embeddings of generated sequences, with $d(z_1, z_2) \triangleq \|z_1 - z_2\|_1$. PCA1-bos stands for the first PCA component of the BOS token embedding of the sequence, with $d(z_1, z_2) \triangleq v^\top z_1$ where $v = \arg\max_{\|v\|_2=1} v^\top \Sigma v$ and Σ being the empirical covariance matrix of the centered data.

Table 5: Ground-truth AUROC on Pfam700-IID-Func2Seq with ESMC-300M embedding model.

Metric	T=0	T=0.3	T=0.5	T=0.7	T=1.0	T=1.5
cos-mean	-	0.7627	0.8314	0.8859	0.6005	0.2071
cos-bos	-	0.7786	0.8313	0.8743	0.7847	0.4359
cos-eos	-	0.6361	0.6974	0.7726	0.6073	0.2030
euc-mean	-	0.8339	0.9053	0.9345	0.7827	0.2957
euc-bos	-	0.7780	0.8393	0.8826	0.8083	0.4737
euc-eos	-	0.6446	0.7185	0.7932	0.6634	0.2249
$L2^2$-mean	-	0.8354	0.8914	0.9224	0.7489	0.2865
$L2^2$-bos	-	0.7788	0.8301	0.8717	0.7890	0.4557
$L2^2$-eos	-	0.6550	0.7190	0.7873	0.6482	0.2254
L1-mean	-	0.8629	0.9259	0.9422	0.8171	0.5148
L1-bos	-	0.7800	0.8402	0.8803	0.7984	0.4780
L1-eos	-	0.6463	0.7185	0.7745	0.6357	0.2419
Predictive Ent	-	0.4078	0.5360	0.7291	0.8357	0.8402
Normalized Ent	-	0.3371	0.3851	0.6268	0.8942	0.9054
Semantic Ent (Kuhn et al., 2023)	-	0.8552	-	0.8384	0.6205	-
PCA-1-mean	-	0.9734	0.9500	0.7009	0.8162	0.7532
PCA-1-bos	-	0.8853	0.6380	0.5619	0.5164	0.5820
PCA-1-eos	-	0.9375	0.9196	0.8192	0.5865	0.5841
PCA-2-mean	-	0.7146	0.5458	0.8963	0.5542	0.7971
PCA-2-bos	-	0.8851	0.9372	0.5818	0.6803	0.8801
PCA-2-eos	-	0.6223	0.7382	0.8422	0.7034	0.6819

Table 6: Ground-truth AUROC on Pfam700-OOD-Func2Seq with ESMC-300M embedding model.

Metric	T=0	T=0.3	T=0.5	T=0.7	T=1.0	T=1.5
cos-mean	-	0.3313	0.4095	0.5541	0.5134	0.3053
cos-bos	-	0.3189	0.3907	0.4851	0.4912	0.3188
cos-eos	-	0.2535	0.3044	0.4000	0.4140	0.2238
euc-mean	-	0.4670	0.5590	0.6498	0.5992	0.3192
euc-bos	-	0.3153	0.3812	0.4806	0.4907	0.3140
euc-eos	-	0.2612	0.3151	0.4187	0.4415	0.2288
$L2^2$-mean	-	0.4637	0.5386	0.6360	0.6110	0.3386
$L2^2$-bos	-	0.3158	0.3857	0.4800	0.4926	0.3238
$L2^2$-eos	-	0.2735	0.3306	0.4268	0.4495	0.2343
L1-mean	-	0.5305	0.5979	0.6185	0.5932	0.4214
L1-bos	-	0.3160	0.3770	0.4660	0.4803	0.3091
L1-eos	-	0.2634	0.3123	0.3952	0.4130	0.2359
Predictive Ent	-	0.2341	0.3264	0.4731	0.6578	0.8222
Normalized Ent	-	0.2087	0.2209	0.3602	0.6896	0.8803
Semantic Ent (Kuhn et al., 2023)	-	0.5506	0.5395	0.6066	0.5490	0.3787
PCA-1-mean	-	0.8288	0.8090	0.7536	0.6214	0.5273
PCA-1-bos	-	0.7841	0.7743	0.6462	0.5051	0.5414
PCA-1-eos	-	0.7712	0.7472	0.7095	0.6144	0.6190
PCA-2-mean	-	0.6068	0.6574	0.5620	0.5276	0.7499
PCA-2-bos	-	0.5521	0.5522	0.6378	0.6305	0.8029
PCA-2-eos	-	0.5039	0.5172	0.5604	0.5162	0.5721

Table 7: Ground-truth AUROC on DRAME-Func2Seq with ESMC-600M embedding model.

Metric	T=0	T=0.3	T=0.5	T=0.7	T=1.0	T=1.5
cos-mean	0.4273	0.5565	0.5933	0.6073	0.7129	0.5986
cos-bos	0.4180	0.5473	0.6352	0.7005	0.7958	0.7135
cos-eos	0.2887	0.3764	0.4355	0.4920	0.6006	0.5240
euc-mean	0.4394	0.5929	0.6808	0.7211	0.8135	0.7078
euc-bos	0.3495	0.4752	0.5889	0.6742	0.7876	0.7156
euc-eos	0.2448	0.3133	0.3801	0.4506	0.5759	0.5623
$L2^2$-mean	0.5111	0.6581	0.7170	0.7378	0.8092	0.6957
$L2^2$-bos	0.4126	0.5405	0.6270	0.6912	0.7860	0.7094
$L2^2$-eos	0.2785	0.3559	0.4068	0.4619	0.5773	0.5625
L1-mean	0.4399	0.5913	0.6907	0.7432	0.8477	0.7731
L1-bos	0.3618	0.4975	0.6278	0.7349	0.8512	0.7113
L1-eos	0.2274	0.2822	0.3464	0.4300	0.5699	0.5401
Predictive Ent	0.1319	0.1302	0.1881	0.3177	0.4836	0.5037
Normalized Ent	0.1385	0.1459	0.1990	0.3041	0.5323	0.8805
Semantic Ent (Kuhn et al., 2023)	-	0.6085	-	0.6172	0.6309	0.4942
PCA-1-mean	0.6233	0.5311	0.6387	0.7932	0.6865	0.5581
PCA-1-bos	0.7246	0.5097	0.8958	0.8509	0.7729	0.6420
PCA-1-eos	0.7056	0.5774	0.7023	0.7958	0.6470	0.6686
PCA-2-mean	0.9118	0.9139	0.8784	0.6930	-	0.7128
PCA-2-bos	0.9273	0.9257	0.7568	0.5666	-	0.5788
PCA-2-eos	0.8989	0.9089	0.8047	0.5758	-	0.5596

Table 8: Ground-truth AUROC on Pfam700-IID-Func2Seq with ESMC-600M embedding model.

Metric	T=0	T=0.3	T=0.5	T=0.7	T=1.0	T=1.5
cos-mean	-	0.8069	0.8682	0.9079	0.6255	0.2290
cos-bos	-	0.8326	0.8740	0.9041	0.8048	0.4209
cos-eos	-	0.6925	0.7484	0.8119	0.5591	0.2252
euc-mean	-	0.8449	0.9121	0.9349	0.7941	0.3173
euc-bos	-	0.8167	0.8755	0.9099	0.8212	0.4296
euc-eos	-	0.6621	0.7238	0.8034	0.6390	0.2269
$L2^2$-mean	-	0.8536	0.9024	0.9247	0.7610	0.2998
$L2^2$-bos	-	0.8295	0.8708	0.9010	0.8007	0.4208
$L2^2$-eos	-	0.6752	0.7230	0.7952	0.6261	0.2280
L1-mean	-	0.8590	0.9266	0.9452	0.8358	0.5629
L1-bos	-	0.8212	0.8861	0.9196	0.8137	0.4367
L1-eos	-	0.6499	0.7158	0.7924	0.6300	0.2680
Predictive Ent	-	0.4078	0.5360	0.7291	0.8357	0.8402
Normalized Ent	-	0.3371	0.3851	0.6268	0.8942	0.9054
Semantic Ent (Kuhn et al., 2023)	-	0.8536	0.8602	0.7981	0.6140	0.3131
PCA-1-mean	-	0.9745	0.9513	0.6758	0.7889	0.7080
PCA-1-bos	-	0.8383	0.7507	0.5445	0.7162	0.6951
PCA-1-eos	-	0.9496	0.9255	0.6443	0.5278	0.6219
PCA-2-mean	-	0.5816	0.5897	0.8973	0.6205	0.6484
PCA-2-bos	-	0.9760	0.9654	0.8626	0.6660	0.8669
PCA-2-eos	-	0.5739	0.8063	0.8622	0.7018	0.7693

Table 9: Ground-truth AUROC on Pfam700-OOD-Func2Seq with ESMC-600M embedding model.

Metric	T=0	T=0.3	T=0.5	T=0.7	T=1.0	T=1.5
cos-mean	-	0.3936	0.4685	0.5935	0.5352	0.3302
cos-bos	-	0.4224	0.4841	0.5803	0.6110	0.3445
cos-eos	-	0.2704	0.3297	0.4528	0.4173	0.2503
euc-mean	-	0.4920	0.5792	0.6529	0.6126	0.3414
euc-bos	-	0.4104	0.4809	0.5806	0.6057	0.3317
euc-eos	-	0.2539	0.3026	0.4361	0.4438	0.2455
$L2^2$-mean	-	0.4993	0.5643	0.6414	0.6206	0.3566
$L2^2$-bos	-	0.4152	0.4780	0.5763	0.6099	0.3447
$L2^2$-eos	-	0.2581	0.3077	0.4382	0.4535	0.2499
L1-mean	-	0.5224	0.5990	0.6257	0.6174	0.4602
L1-bos	-	0.4298	0.5035	0.5819	0.5966	0.3311
L1-eos	-	0.2321	0.2731	0.3936	0.4050	0.2518
Predictive Ent	-	0.2341	0.3264	0.4731	0.6578	0.8222
Normalized Ent	-	0.2087	0.2209	0.3602	0.6896	0.8803
Semantic Ent (Kuhn et al., 2023)	-	0.5583	0.5854	0.5866	0.5625	0.4088
PCA-1-mean	-	0.8324	0.8125	0.7470	0.6262	0.5642
PCA-1-bos	-	0.8429	0.7992	0.6334	0.6336	0.6769
PCA-1-eos	-	0.7634	0.7529	0.7180	0.5097	0.5947
PCA-2-mean	-	0.5546	0.6030	0.5601	0.5207	0.6780
PCA-2-bos	-	0.5119	0.7333	0.6406	0.5619	0.5677
PCA-2-eos	-	0.5724	0.5217	0.5634	0.7023	0.6086

F More Experiments on Post-training

In parallel to Tab. 1, we also provide keyword recovery statistics for each Pfam family prompts in Tab. 10 where BRO/SRO consistently achieve the best/second best performance across 7 families. Additionally, we also examine the sensitivity of SRO to hyperparameter β in Eq. (11). Tab. 11 demonstrates that the performance of SRO is stable across different values of β and outperforms the batch-normalized weight DPO (wDPO) proposed in (Widatalla et al., 2024).

Table 10: Keyword recovery for each Pfam family averaged over five sampling temperatures.

	PF00002	PF00042	PF00125	PF00127	PF00257	PF00262	PF03668	Overall
151M model								
Progen2-small-mix7	0.359	0.168	0.295	0.439	0.156	0.146	0.406	0.281
Progen2-small-mix7 DPO	0.502	0.213	0.328	0.516	0.237	0.158	0.494	0.350
Progen2-small-mix7 KTO	0.494	0.197	0.335	0.494	0.165	0.169	0.503	0.337
Progen2-small-mix7 BRO	**0.630**	0.238	**0.503**	0.672	**0.279**	0.180	**0.528**	**0.433**
Progen2-small-mix7 SRO	0.617	**0.266**	0.468	**0.735**	0.197	**0.210**	0.495	0.427
Progen2-small-mix7 Supervised	0.632	0.432	0.513	0.776	0.365	0.304	0.397	0.488
764M model								
Progen2-medium-mix7	0.270	0.187	0.497	0.377	0.223	0.535	0.604	0.385
Progen2-medium-mix7 DPO	0.396	0.275	0.644	0.496	**0.350**	0.523	0.675	0.480
Progen2-medium-mix7 KTO	0.385	0.245	0.661	0.527	0.282	0.681	0.751	0.505
Progen2-medium-mix7 BRO	**0.539**	**0.327**	**0.839**	0.691	0.233	**0.907**	0.754	**0.613**
Progen2-medium-mix7 SRO	0.482	0.315	0.838	**0.693**	0.292	0.827	**0.766**	0.602
Progen2-medium-mix7 Supervised	0.276	0.457	0.812	0.578	0.502	0.824	0.774	0.603

G Details of Model Training and Evaluation

To reduce noise in the unsupervised rewards, for each prompt, we keep only the top-4 and bottom-4 of the 64 generated samples in $\mathcal{D}$ for all post-training experiments (including single-temperature control groups in Appendix D), resulting in $56k$ samples for training and $28k$ for testing. The reward statistics of this selected subset are consistent with those of the full dataset, yet exhibit a stronger correlation with the ground truth (see Tab. 12). For BRO/DPO/KTO that operate on binarized reward, we label top-4 as positive samples and bottom-4 as negative ones. For training DPO specifically,

Table 11: SRO with different β for fine-tuning Progen2-small-mix7 on Pfam700 Func2Seq task.

Method	$T = 0.3$	$T = 0.5$	$T = 0.7$	$T = 1.0$	$T = 1.5$	Average
base	0.210	0.288	0.390	0.401	0.118	0.281
SRO ($\beta = 1$, Tab. 1)	0.383	0.455	0.498	0.499	0.301	0.427
SRO ($\beta = 0.5$)	0.374	0.433	0.483	0.492	0.308	0.418
SRO ($\beta = 2.0$)	0.329	0.448	0.508	0.506	0.286	0.415
Weighted DPO (Widatalla et al., 2024)	0.455	0.463	0.455	0.377	0.153	0.381

we randomly pair the top-4 and bottom-4 samples to produce four preference pairs per prompt. We train all BRO/SRO variants as well as two offline baselines, DPO (Rafailov et al., 2023) and KTO (Ethayarajh et al., 2024) on the same dataset for 1 epoch with AdamW optimizer (Loshchilov & Hutter) and learning rate `5e-7` without extensive hyperparameter search on a single NVIDIA A40 GPU. Due to the memory constraint, we train the 151M model with batch size 32 and the 764M model with batch size 8. All fine-tuned models employ a top-k decoding with $k = 15$. The DPO and KTO baselines are trained with default configurations from the `TRL` package. All SFT and post-training are conducted via an 8:2 train-validation split.

Table 12: Promptwise AUROC on selected test sets with ESMC-300M embedding model.

	$T = 0.3$	$T = 0.5$	$T = 0.7$	$T = 1.0$
Pfam700-OOD-Func2Seq				
L1-mean	0.4949	0.6214	0.7411	0.5875
Predictive Ent	0.2332	0.3479	0.5360	0.7179
Normalized Ent	0.2149	0.2358	0.4267	0.7841
PCA1-bos	0.8839	0.8583	0.6051	0.5929
Pfam700-OOD-Func2Seq Top4-Bot4				
L1-mean	0.5590	0.6963	0.8006	0.6426
Predictive Ent	0.2262	0.3317	0.5429	0.7659
Normalized Ent	0.2072	0.2144	0.4295	0.8185
PCA1-bos	0.9086	0.8829	0.6114	0.6098

H Critical Temperature as the Best-Performing Temperature

To support this claim in Section 4.2, we report the recovery rate of the ESM3 and Progen2-small-mix7 base model on DRAME-Func2Seq and Pfam700-Func2Seq tasks respectively:

Table 13: Keyword recovery on DRAME-Func2Seq and Pfam700-Func2Seq tasks over six sampling temperatures. Pfam700-IID-Func2Seq refers to the 7 Pfam family tokens (without residue tokens) used in the SFT stage of Progen2-small-mix7.

	$T = 0.0$	$T = 0.3$	$T = 0.5$	$T = 0.7$	$T = 1.0$	$T = 1.5$
DRAME-Func2Seq	0.207	0.210	0.229	0.259	0.263	0.094
Pfam700-IID-Func2Seq	-	0.681	0.747	0.809	0.755	0.322
Pfam700-OOD-Func2Seq	-	0.210	0.288	0.390	0.401	0.118

The best-performing temperature for ESM3 on DRAME-Func2Seq and Progen2-small-mix7 on Pfam700-Func2Seq is $T \sim 1.0$ and $T \in [0.7, 1.0]$ respectively, coinciding with the critical temperatures observed in Figure 1(a) and (b).

I Additional Experiments

I.1 Best-of-N Selection

The reported best-of-N protocol scores a pool of 64 base-model candidates independently within each prompt–temperature group using negative mean pairwise L1 distance in ESMC-600M embedding space. It retains the top $64/N$ candidates. Thus, N is a filtering factor: best-of-8 retains eight candidates from 64, whereas best-of-64 retains one. For $N < 64$, this differs from independently drawing N candidates and returning one. Recovery is averaged over the retained outputs. Scoring uses the candidate group, not task labels, and the computation of the full pool must be counted even when few outputs are retained.

Table 14 reports temperature-resolved ProGen2 recovery. Table 15 summarizes best-of-64 for all three models. Neither table establishes a matched-total-compute advantage once offline sampling and training are included. At 764M and $T = 0.7$, best-of-8 filtering reaches 0.790, while BRO and SRO reach 0.701 and 0.708, respectively.

Table 14: Best-of-N recovery on Pfam700. Averages weight the five temperatures equally. Base and fine-tuned results are in Table 1.

Selection	$T = 0.3$	$T = 0.5$	$T = 0.7$	$T = 1.0$	$T = 1.5$	Average
151M ProGen2						
Best-of-2	0.263	0.403	0.528	0.434	0.0682	0.339
Best-of-4	0.289	0.449	0.579	0.474	0.0733	0.373
Best-of-8	0.304	0.460	0.595	0.501	0.0827	0.389
Best-of-16	0.314	0.461	0.597	0.522	0.0875	0.396
Best-of-32	0.323	0.459	0.594	0.529	0.0929	0.400
Best-of-64	0.324	0.469	0.594	0.539	0.0903	0.403
764M ProGen2						
Best-of-2	0.472	0.640	0.726	0.386	0.0396	0.453
Best-of-4	0.498	0.689	0.771	0.417	0.0441	0.484
Best-of-8	0.511	0.708	0.790	0.445	0.0488	0.500
Best-of-16	0.524	0.718	0.800	0.470	0.0496	0.512
Best-of-32	0.536	0.726	0.797	0.488	0.0529	0.520
Best-of-64	0.546	0.723	0.789	0.513	0.0514	0.524

I.2 Best-of-64 Summary and ESM3 Adaptation

For ProGen2, keyword recovery is the binary recovery of the requested Pfam family. For ESM3, it is the fraction of requested InterPro keywords recovered by InterProScan, averaged over generated sequences. These are related but distinct metrics. Table 15 reports the temperature-averaged comparison. ESM3 uses one-pass full-mask decoding, with the sequence log-probability given by the sum of token log-probabilities over output residues for both the policy and reference. These results do not include training-seed uncertainty or a matched-budget baseline search.

Table 15: Func2Seq keyword recovery averaged over five temperatures. Supervised is the label-trained comparator. The ESM3 entries use fractional keyword recovery, not a thresholded pass rate.

Model	Base	L1 best-of-64	BRO (L1)	Supervised
ProGen2 151M	0.281	0.403	0.433	0.488
ProGen2 764M	0.385	0.524	0.613	0.603
ESM3 1.4B	0.197	0.224	0.255	0.260

Table 16 gives preliminary Struct2Seq results, using a backbone cRMSD threshold of 2 Å. Improvements are modest and are not evidence of biological function.

Table 16: ESM3 DRAME-Struct2Seq pass@k (%).

Model	$k=1$	$k=2$	$k=4$	$k=8$	$k=16$	$k=32$
Base	14.76	20.00	24.93	29.40	33.05	36.03
BRO (L1)	16.66	21.29	26.07	30.62	34.39	37.26

I.3 Pooled Sequence Diversity

The diversity analysis uses 64 sequences at each of five temperatures per prompt. Mean pairwise identity and unique-sequence fraction are aggregated over prompts, temperatures, and families. BRO has a slightly higher mean pairwise identity and lower uniqueness than the base model; SRO has the opposite pattern. These pooled results do not determine diversity within each temperature or among successful sequences only. In particular, high uniqueness can include invalid sequences.

Table 17: Pooled diversity for 151M ProGen2 on the compositional prompt task.

Model	Mean pairwise identity	Mean unique rate
Base	0.1297	0.9572
BRO (L1)	0.1358	0.9467
SRO (r_T)	0.1204	0.9686

Cluster-based diversity. A separate analysis clusters generated sequences with MMseqs2 `easy-linclust`, using `–min-seq-id 0.1 -c 0.7 –cov-mode 1`. Diversity is the number of clusters divided by the number of evaluated sequences, retaining duplicates in the denominator. Table 18 reports temperature-averaged values with a budget of 64 candidates per prompt. This statistic differs from exact-sequence uniqueness and should not be compared numerically with Table 17. ProGen2 BRO reduces cluster diversity despite increasing keyword recovery. Reranking diversity is not reported here because its smaller retained-output budget changes the clustering statistic; a matched-output-budget comparison remains necessary.

Table 18: Cluster-based diversity averaged over five temperatures.

Model	Base	BRO (L1)
ProGen2 151M	0.673	0.577
ProGen2 764M	0.623	0.508
ESM3 1.4B	0.548	0.578

I.4 Limited Baseline Tuning

Table 19 records the available additional DPO/KTO runs at 151M. Missing entries were not reported. This restricted sweep changes β only and does not establish that either baseline is optimally tuned. It also does not quantify training-seed variability.

Table 19: 151M baseline sensitivity to β on Pfam700. The $\beta = 0.1$ rows are the primary baseline settings.

Method	$T = 1.0$	$T = 1.5$
DPO ($\beta = 0.1$)	0.435	0.122
DPO ($\beta = 0.05$)	0.424	0.113
DPO ($\beta = 0.2$)	0.443	0.126
KTO ($\beta = 0.1$)	0.458	0.175
KTO ($\beta = 0.05$)	0.464	—
KTO ($\beta = 0.2$)	0.460	—