AlphaFold Distillation for Protein Design

Igor Melnyk[*†] **Aurelie Lozano**[*] **Payel Das**[*] **Vijil Chenthamarakshan**[*]

Abstract

Inverse protein folding, the process of designing sequences that fold into a specific 3D structure, is crucial in bio-engineering and drug discovery. Traditional methods rely on experimentally resolved structures, but these cover only a small fraction of protein sequences. Forward folding models like AlphaFold offer a potential solution by accurately predicting structures from sequences. However, these models are too slow for integration into the optimization loop of inverse folding models during training. To address this, we propose using knowledge distillation on folding model confidence metrics, such as pTM or pLDDT scores, to create a faster and end-to-end differentiable distilled model. This model can then be used as a structure consistency regularizer in training the inverse folding model. Our technique is versatile and can be applied to other design tasks, such as sequence-based protein infilling. Experimental results show that our method outperforms non-regularized baselines, yielding up to 3% improvement in sequence recovery and up to 45% improvement in protein diversity while maintaining structural consistency in generated sequences. Code is available at `https://github.com/IBM/AFDistill`.

1 Introduction

Eight of the top ten best-selling drugs are engineered proteins, underscoring the importance of inverse protein folding in bio-engineering and drug discovery [3]. This task, termed computational protein design, traditionally optimizes amino acid sequences against a physics-based function [20]. Recently, deep generative models have addressed this challenge [17, 5, 27, 19, 14, 11]. However, these models often miss the key objective: designing novel, *diverse* sequences with new functions.

In parallel, recent advancements have also improved protein representation learning [24, 28], structure prediction [18, 4], and sequence generation [6, 2]. Inverse protein folding traditionally used sequences with resolved structures, but performance improved using AlphaFold-predicted structures [14]. However, this method is computationally costly. A more efficient method could be to use a pre-trained forward folding model to guide the training of the inverse folding model.

In this work, we introduce a framework training the inverse folding model with a combined sequence reconstruction and *structure consistency loss (SC)* (Fig. 1). While one could use folding models like AlphaFold for structure estimation, its computational cost (Fig. 2) and the requirement for ground truth reference structure pose challenges. Using internal confidence metrics from AlphaFold is an alternative, but still remains slow for in-the-loop optimization. To address this, we: **(i)** Perform knowledge distillation on AlphaFold and incorporate the resulting model, AFDistill (fixed), into the regularized training of the inverse folding model, which is referred to as structure consistency (SC) loss. The major *novelty* here is that AFDistill enables direct prediction of TM or LDDT of a given protein sequence bypassing the structure estimation or the access to ground truth structure. Our model is fast, accurate, and fully differentiable. **(ii)** Perform extensive evaluations, demonstrating

[*]IBM Research, Yorktown Heights, NY 10598
[†]Corresponding author: `igor.melnyk@ibm.com`

Machine Learning for Structural Biology Workshop, NeurIPS 2023.

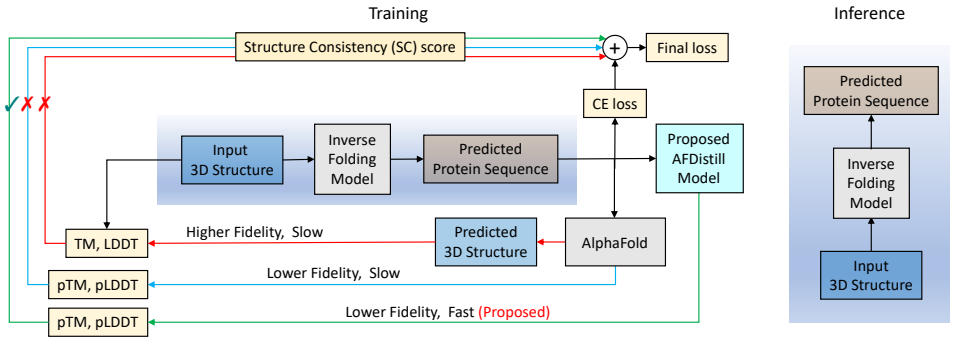

Figure 1: Overview of the proposed AFDistill system. AFDistill contrasts with traditional methods (red line) that use models like AlphaFold to predict protein structure, which is then compared to the actual structure. This method is slow due to model inference times (refer Fig. 2). An alternative (blue line) uses internal metrics from folding model without structure prediction but remains slow and less precise. Our solution, distills AlphaFold's confidence metrics into a faster, differentiable model that offers accuracy akin to AlphaFold, allowing seamless integration into the training process (green line). The improved inverse folding model's inference is shown on the right.

that our proposed system surpasses existing benchmarks in structure-guided protein sequence design by achieving lower perplexity, higher amino acid recovery, and maintaining proximity to the original protein structure. It also boosts sequence diversity, a crucial goal in protein design. Despite the trade-off between sequence and structure recovery, our regularized model ensures sequence diversity without compromising structural integrity. **(iii)** Lastly, our SC metric can be utilized for regularization in tasks like inverse folding or other protein optimization (e.g., [23]) benefiting from structural consistency estimation. It can also serve as an affordable AlphaFold alternative that provides scoring of a given protein, reflecting its structural content.

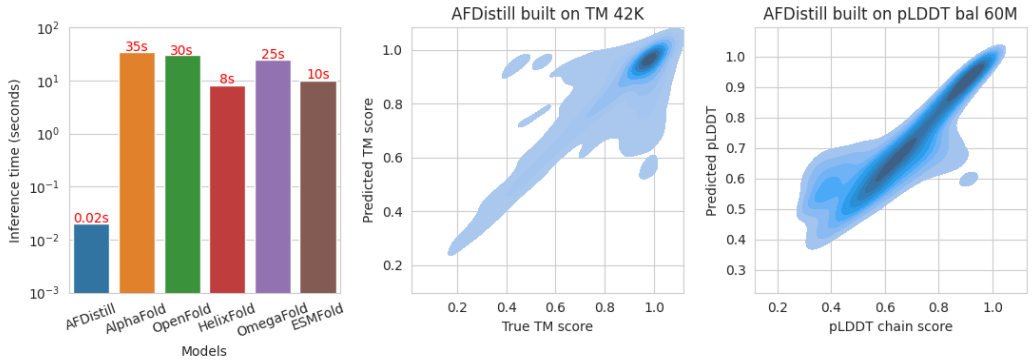

Figure 2: Inference times for protein sequences using our AFDistill model compared to alternatives are displayed on the left. AFDistill maintains fast inference for longer sequences: 0.028s for 1024-length and 0.035s for 2048-length. Timings for AlphaFold and OpenFold [1] do not include MSA search times, which can range from minutes to hours. Values for HelixFold [10], OmegaFold [26], and ESMFold [22] are from their publications. The center plot shows kernel density of true vs. AFDistill-predicted TM scores (Pearson's correlation: 0.77), while the right displays a similar plot for pLDDT values (Pearson's correlation: 0.76). Refer to Section 2 for details.

2 AlphaFold Distill

Knowledge distillation [13] transfers knowledge from a large model like AlphaFold to a smaller one, such as the AFDistill model. While traditionally done using soft labels from the AlphaFold model and hard labels as ground truth, we instead use the model's predictions (pTM/pLDDT) and hard labels, TM/LDDT scores, derived from AlphaFold's predicted 3D structures.

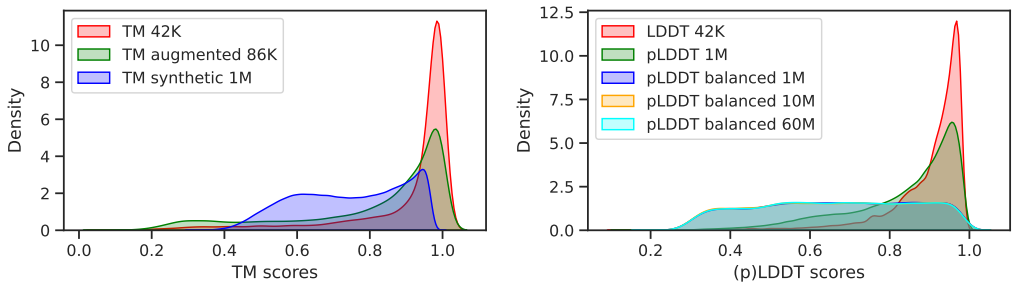

Figure 3: Distribution of the (p)TM/(p)LDDT scores in various datasets used in AFDistill training.

Data Using Release 3 (January 2022) of AlphaFold Protein Structure Database [25], we obtained 907,578 predicted structures, each with 3D residue coordinates and pLDDT confidence scores. Structures with $\geq 40\%$ sequence similarity to CATH 4.2 dataset's validation/test splits were excluded. From the filtered structures, we formed our pLDDT 1M dataset, pairing protein sequences with per-residue pLDDTs. Protein lengths were capped at 500 via random subsequence cropping. We created datasets from true TM and LDDT values using predicted AlphaFold structures. Using the PDB-to-UniProt mapping, we curated 42,605 structures with corresponding ground truth PDB sequences, named TM 42K and LDDT 42K. Fig. 3 displays their score distribution, leaning towards higher scores. Addressing this imbalance, we introduced two more TM-based datasets. TM augmented 86K augments TM 42K with altered protein sequences and their AlphaFold-estimated structures with mainly low and medium TM scores. pTM synthetic 1M, created by random protein sequences processed by AFDistill (trained on TM 42K), offers more lower-range pTM values. Both datasets provide a more balanced score distribution.

Using Release 4 (July 2022) with over 214M predicted structures, we observed a similar high skewness in pLDDT values. We filtered out high mean-pLDDT samples, yielding a dataset of 60M sequences, and also produced 10M and 1M versions (Fig. 3). In summary, AFDistill predicts actual structural measures (TM, LDDT) and AlphaFold's estimated scores (pTM and pLDDT). In either case, the structural consistency (SC) score correlates well with its target (see Fig.2) and can be used as an indicator of protein sequence quality or validity.

Model AFDistill model is based on ProtBert [9], a Transformer BERT model (420M parameters) pretrained on a large corpus of protein sequences using masked language modeling. For our task we modify ProtBert head by setting the vocabulary size to 50 (bins), corresponding to discretizing pTM/pLDDT in range (0,1). For pTM (scalar) the output corresponds to the first $\langle \texttt{CLS} \rangle$ token of the output sequence, while for pLDDT (sequence) the predictions are made for each residue position.

Distillation Results Here we discuss the model evaluation results after training on the presented datasets. To address data imbalance, we used weighted sampling during minibatch generation and Focal loss [21] instead of traditional cross-entropy loss. In Fig. 2, kernel density plots depict true vs pTM scores and pLDDT values for the validation set. Most density aligns with the diagonal, suggesting accurate predictions. Mismatches in off-diagonal regions have low density. Given the TM 42K data skews towards 1.0 (top in Fig. 3), it clusters in the upper left. For the balanced pLDDT bal 60M dataset (bottom in Fig. 3), predictions and true values distribute uniformly along the diagonal.

3 Inverse Protein Folding Design

In this section we demonstrate the benefit of applying AFDistill as a structure consistency (SC) score for solving the task of inverse protein folding. The framework is shown in Fig. 1 (green line), where inverse folding model is regularized by SC score. Specifically, during training, the generated protein is fed into AFDistill, for which it predicts pTM or pLDDT score, and combined with the original CE training objective results in

$$\mathcal{L} = \mathcal{L}_{\text{CE}} + \alpha \mathcal{L}_{\text{SC}}, \tag{1}$$

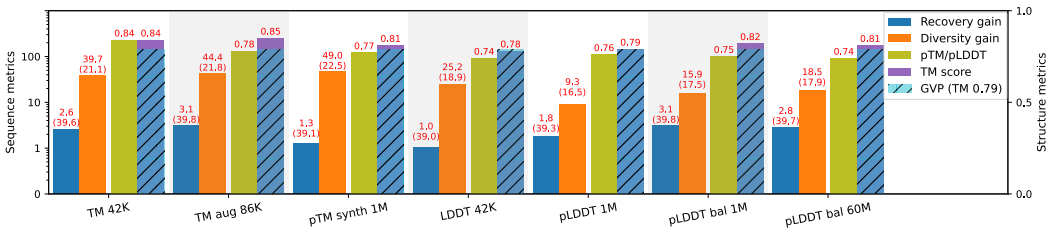

Figure 4: Results for GVP with SC regularization are displayed against various AFDistill datasets on the x-axis. The left y-axis presents sequence metrics, and the right y-axis structure metrics (TM and SC scores). Blue and orange bars denote recovery and diversity gains over the GVP baseline (38.6 recovery, 15.1 diversity, 0.79 TM). Olive and purple bars show predicted SC and TM test scores, while the dashed cyan bar indicates the baseline GVP TM score. TM 42K and TM augmented 86K AFDistill models offer the best performance, highlighting high diversity and notable sequence/structure recovery improvements.

where $\mathcal{L}_{\text{CE}} = \sum_1^N \mathcal{L}_{\text{CE}}(\mathbf{s}_i, \hat{\mathbf{s}}_i)$ is the CE loss, $\mathbf{s}_i$ is the ground truth and $\hat{\mathbf{s}}_i$ is the generated protein sequence, $\mathcal{L}_{\text{SC}} = \sum_{i=1}^N (1 - SC(\hat{\mathbf{s}}_i))$ is the structure consistency loss, N the number of training sequences, and α is the weighting scalar for the SC loss, in our experiment it is set to 1. To evaluate performance, we use the standard metrics such as recovery, diversity, perplexity and TM-score.

3.1 Results

We present experimental results for several recently proposed deep generative models for protein sequence design accounting for 3D structural constraints. For the inverse folding tasks we use CATH 4.2 dataset, curated by [15]. The training, validation, and test sets have 18204, 608, and 1120 structures, respectively.

GVP Geometric Vector Perceptron GNNs (GVP) [17] is the inverse folding model, that for a given target backbone structure, represented as a graph over the residues, replaces dense layers in a GNN by simpler layers, called GVP layers, directly leveraging both scalar and geometric features. This allows for the embedding of geometric information at nodes and edges without reducing such information to scalars that may not fully capture complex geometry. The results of augmenting GVP training with SC score regularization are shown in Fig. 4.

Baseline GVP without regularization achieves 38.6 in recovery, 15.1 in diversity, and 0.79 in TM score on the test set. Employing SC regularization leads to consistent improvements in sequence recovery (1-3%) and significant diversity gain (up to 45%) while maintaining high TM scores. pTM-based SC scores show a better overall influence on performance compared to pLDDT-based ones. It's important to note that AFDistill's validation performance on distillation data doesn't always reflect downstream application performance. For example, TM augmented 86K outperforms TM 42K, despite having slightly worse validation CE loss. This suggests that augmented models may enable more generalized sequence-structure learning and provide a greater performance boost for inverse folding models.

Table 1: Evaluation results of ProteinMPNN trained with and without SC regularization (AFDistill trained on TM aug 86K dataset). The values in parenthesis show gain on test set of using SC-regularized training as compared to original training.

	Recovery		Diversity		Perplexity	
	ProteinMPNN	ProteinMPNN +SC	ProteinMPNN	ProteinMPNN +SC	ProteinMPNN	ProteinMPNN +SC
Backbone Noise 0.02	47.7	47.5 (-0.4%)	22.5	24.3 (+8.0%)	5.1	5.1 (+0.0%)
Backbone Noise 0.1	43.8	44.0 (+0.5%)	28.1	30.4 (+8.2%)	5.3	5.4 (+1.9%)
Backbone Noise 0.2	39.5	39.9 (+1.0%)	31.3	34.4 (+9.9%)	5.8	5.8 (+0.0%)
Backbone Noise 0.3	36.3	36.4 (+0.0%)	33.0	37.8 (+14.6%)	6.2	6.3 (+1.6%)

Note on sequence diversity In Section F of Appendix we offer a set of experiments to shed some light on why SC regularization leads to improved sequence diversity. In particular in Fig.9 we show

Table 2: Experiments on PiFold comparing the performance metrics on the test set of CATH 4.2 for different model variants (original vs SC-regularized training based on different AFDistill models) using greedy and sampled decoding strategies. The values in parentheses represent the percentage change with respect to the original PiFold model.

	Original		TM 42K		TM aug 86K		TM synth 1M		LDDT 42K		pLDDT 1M		pLDDT bal 60M	
	Rec	Perp	Rec	Perp	Rec	Perp	Rec	Perp	Rec	Perp	Rec	Perp	Rec	Perp
Greedy	51.1	4.8	50.9 (-0.4%)	5.0 (+4.0%)	51.0 (-0.2%)	4.8 (+0.0%)	50.5 (-1.2%)	5.2 (+8.3%)	50.8 (-0.6%)	4.9 (+2.1%)	50.9 (-0.4%)	4.8 (+0.0%)	51.1 (+0.0%)	4.7 (-2.1%)
	Rec	Div	Rec	Div	Rec	Div	Rec	Div	Rec	Div	Rec	Div	Rec	Div
Sampled	42.6	52.4	42.5 (-0.2%)	60.7 (+15.8%)	42.8 (+0.5%)	60.2 (+14.9%)	42.4 (-0.5%)	61.1 (+16.6%)	42.3 (-0.7%)	60.9 (+16.2%)	42.5 (-0.2%)	60.5 (+15.5%)	42.9 (+0.7%)	60.0 (+14.5%)

that the main source of diversity is in the limited guidance from AFDistill about the specific sequence to generate to match a given 3D structure, since it does not have access to the structural information, allowing many relevant sequences with high pTM/pLDDT to be considered as good candidates. AFDistill regularization during training injects candidate sequences which have high pTM/pLDDT scores, therefore likely matching the input structure better, thus ensuring high recovery rate. At the same time these sequences differ from the ground truth, thus promoting diversity (see Section F for more details).

ProteinMPNN ProteinMPNN model [7] is a recent protein design model, which is based on message passing neural network (MPNN) with specific modifications to improve amino acid sequences recovery of proteins given their backbone structures. The model incorporates structure features, edge updates, and an autoregressive approach for decoding the sequences. In Table 1 we compared the results of original unmodified training of ProteinMPNN to the SC-regulared training (AFDistill model trained on TM aug 86K dataset). We also varied ProteinMPNN internal parameter, which adds noise to the input backbone protein structure. As can be seen, SC regularization maintains high recovery and perplexity rates while improving the diversity of the generated protein sequences. Backbone noise, which is a part of ProteinMPNN model, can also be seen as a form of regularization, however while the increase in noise leads to improved sequence diversity it also leads to the decrease in amino acid recovery rate. SC regularization, on the other hand, promotes diverse generation and maintains high sequence recovery rates.

PiFold PiFold [12] is another recent protein design model which introduces a new residue featurizer and stacked PiGNNs (protein inverse folding graph neural networks). The residue featurizer constructs residue features and introduces learnable virtual atoms to capture information that could be missed by real atoms. The PiGNN layer learns representations from multi-scale residue interactions by considering feature dependencies at the node, edge, and global levels. In Table 2 we present the results of original and SC-regularized PiFold (using different AFDistill models). We note that the original PiFold evaluation was based on using greedy decoding to generate a sequence. Following the standard practice (GVP, GraphTrans, ProteinMPNN, etc), we have included also the results based on sampling (using 100 samples per sequence) to match other works and compute sequence diversity score. The results show that SC regularization based on AFDistill trained on TM aug 86K results in a near-identical recovery rate compared to the original model, while notably enhancing sequence diversity. This indicates an improvement in PiFold's performance by maintaining recovery rates while increasing the variety of generated sequences. Also observe a decrease in recovery rates for sampled generation as compared to greedy decoding across all the models.

4 Conclusion

In this work we introduce AFDistill, a distillation model based on AlphaFold, which for a given protein sequence estimates its structural consistency (SC: pTM or pLDDT) score. We provide experimental results to showcase the efficiency and efficacy of the AFDistill model in high-quality protein sequence design, when used together with many of the current state of the art protein inverse folding models. Our AFDistill model is fast and accurate enough so that it can be efficiently used for regularizing structural consistency in protein optimization tasks, maintaining sequence and structural integrity, while introducing diversity and variability in the generated proteins.

References

[1] G. Ahdritz, N. Bouatta, S. Kadyan, Q. Xia, W. Gerecke, T. J. O'Donnell, D. Berenberg, I. Fisk, N. Zanichelli, B. Zhang, A. Nowaczynski, B. Wang, M. M. Stepniewska-Dziubinska, S. Zhang, A. Ojewole, M. E. Guney, S. Biderman, A. M. Watkins, S. Ra, P. R. Lorenzo, L. Nivon, B. Weitzner, Y.-E. A. Ban, P. K. Sorger, E. Mostaque, Z. Zhang, R. Bonneau, and M. AlQuraishi. OpenFold: Retraining AlphaFold2 yields new insights into its learning mechanisms and capacity for generalization. *bioRxiv*, 2022.

[2] I. Anishchenko, S. J. Pellock, T. M. Chidyausiku, T. A. Ramelot, S. Ovchinnikov, J. Hao, K. Bafna, C. Norn, A. Kang, A. K. Bera, et al. De novo protein design by deep network hallucination. *Nature*, 600(7889):547–552, 2021.

[3] P. V. Arnum. *Pharma Pulse: Top-Selling Small Molecules and Biologics*, 2022.

[4] M. Baek, F. DiMaio, I. Anishchenko, J. Dauparas, S. Ovchinnikov, G. R. Lee, J. Wang, Q. Cong, L. N. Kinch, R. D. Schaeffer, et al. Accurate prediction of protein structures and interactions using a three-track neural network. *Science*, 373(6557):871–876, 2021.

[5] Y. Cao, P. Das, V. Chenthamarakshan, P.-Y. Chen, I. Melnyk, and Y. Shen. Fold2seq: A joint sequence (1d)-fold (3d) embedding-based generative model for protein design. In *International Conference on Machine Learning*, pages 1261–1271. PMLR, 2021.

[6] P. Das, T. Sercu, K. Wadhawan, I. Padhi, S. Gehrmann, F. Cipcigan, V. Chenthamarakshan, H. Strobelt, C. Dos Santos, P.-Y. Chen, et al. Accelerated antimicrobial discovery via deep generative models and molecular dynamics simulations. *Nature Biomedical Engineering*, 5(6):613–623, 2021.

[7] J. Dauparas, I. Anishchenko, N. Bennett, H. Bai, R. J. Ragotte, L. F. Milles, B. I. Wicky, A. Courbet, R. J. de Haas, N. Bethel, et al. Robust deep learning–based protein sequence design using proteinmpnn. *Science*, 378(6615):49–56, 2022.

[8] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[9] A. Elnaggar, M. Heinzinger, C. Dallago, G. Rihawi, Y. Wang, L. Jones, T. Gibbs, T. Feher, C. Angerer, M. Steinegger, et al. ProtTrans: towards cracking the language of life's code through self-supervised deep learning and high performance computing. *arXiv preprint arXiv:2007.06225*, 2020.

[10] X. Fang, F. Wang, L. Liu, J. He, D. Lin, Y. Xiang, X. Zhang, H. Wu, H. Li, and L. Song. HelixFold-Single: Msa-free protein structure prediction by using protein language model as an alternative. *arXiv preprint arXiv:2207.13921*, 2022.

[11] T. Fu and J. Sun. SIPF: Sampling Method for Inverse Protein Folding. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 378–388, 2022.

[12] Z. Gao, C. Tan, and S. Z. Li. Pifold: Toward effective and efficient protein inverse folding. In *International Conference on Learning Representations*, 2023.

[13] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. In *NIPS Deep Learning and Representation Learning Workshop*, 2015.

[14] C. Hsu, R. Verkuil, J. Liu, Z. Lin, B. Hie, T. Sercu, A. Lerer, and A. Rives. Learning inverse folding from millions of predicted structures. *bioRxiv*, 2022.

[15] J. Ingraham, V. Garg, R. Barzilay, and T. Jaakkola. Generative models for graph-based protein design. *Advances in neural information processing systems*, 32, 2019.

[16] W. Jin, J. Wohlwend, R. Barzilay, and T. Jaakkola. Iterative refinement graph neural network for antibody sequence-structure co-design. *arXiv preprint arXiv:2110.04624*, 2021.

[17] B. Jing, S. Eismann, P. Suriana, R. J. Townshend, and R. Dror. Learning from protein structure with geometric vector perceptrons. *arXiv preprint arXiv:2009.01411*, 2020.

[18] J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Žídek, and A. Potapenko. Highly accurate protein structure prediction with AlphaFold. *Nature*, 596(7873):583–589, 2021.

[19] M. Karimi, S. Zhu, Y. Cao, and Y. Shen. De novo protein design for novel folds using guided conditional wasserstein generative adversarial networks. *Journal of chemical information and modeling*, 60(12):5667–5681, 2020.

[20] B. Kuhlman, G. Dantas, G. C. Ireton, G. Varani, B. L. Stoddard, and D. Baker. Design of a novel globular protein fold with atomic-level accuracy. *Science*, 302(5649):1364–1368, 2003.

[21] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.

[22] Z. Lin, H. Akin, R. Rao, B. Hie, Z. Zhu, W. Lu, A. dos Santos Costa, M. Fazel-Zarandi, T. Sercu, S. Candido, et al. Language models of protein sequences at the scale of evolution enable accurate structure prediction. *bioRxiv*, 2022.

[23] L. Moffat, J. G. Greener, and D. T. Jones. Using AlphaFold for rapid and accurate fixed backbone protein design. *bioRxiv*, 2021.

[24] A. Rives, J. Meier, T. Sercu, S. Goyal, Z. Lin, J. Liu, D. Guo, M. Ott, C. L. Zitnick, J. 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] M. Varadi, S. Anyango, M. Deshpande, S. Nair, C. Natassia, G. Yordanova, D. Yuan, O. Stroe, G. Wood, A. Laydon, A. Žídek, T. Green, K. Tunyasuvunakool, S. Petersen, J. Jumper, E. Clancy, R. Green, A. Vora, M. Lutfi, M. Figurnov, A. Cowie, N. Hobbs, P. Kohli, G. Kleywegt, E. Birney, D. Hassabis, and S. Velankar. AlphaFold Protein Structure Database: massively expanding the structural coverage of protein-sequence space with high-accuracy models. *Nucleic Acids Research*, 50(D1):D439–D444, 11 2021.

[26] R. Wu, F. Ding, R. Wang, R. Shen, X. Zhang, S. Luo, C. Su, Z. Wu, Q. Xie, B. Berger, J. Ma, and J. Peng. High-resolution de novo structure prediction from primary sequence. *bioRxiv*, 2022.

[27] Z. Wu, P. Jain, M. Wright, A. Mirhoseini, J. E. Gonzalez, and I. Stoica. Representing long-range context for graph neural networks with global attention. *Advances in Neural Information Processing Systems*, 34:13266–13279, 2021.

[28] Z. Zhang, M. Xu, A. Jamasb, V. Chenthamarakshan, A. Lozano, P. Das, and J. Tang. Protein representation learning by geometric structure pretraining. *arXiv preprint arXiv:2203.06125*, 2022.

Supplementary Material for Paper Submission
AlphaFold Distillation for Inverse Protein Folding

Table of Contents

A Limitations of the Proposed Work

Although our proposed AFDistill system is novel, efficient and showed promising results during evaluations, there are a number of limitations of the current approach:

- AFDistill dependency on the accuracy of the AlphaFold forward folding model: The quality of the distilled model is directly related to the accuracy of the original forward folding model, including the biases inherited from it.
- Limited coverage of protein sequence space: Despite the advances in AlphaFold forward folding models, they are still limited in their ability to accurately predict the structure of many protein sequences, including the TM score and pLDDT confidence metrics, that AFDistill relies on.
- Uncertainty in structural predictions: The confidence metrics (TM score and pLDDT) used in the distillation process are subject to uncertainty, which can lead to errors in the distilled model's predictions and ultimately impact the quality of the generated sequences in downstream applications.
- The need for a large amount of computational resources: The training process of AFDistill model requires significant computational resources. However, this might be mitigated by the amortization effect where the high upfront training cost in downstream applications pays in terms of cheap and fast inference through the model.

B AlphaFold Model Overview

A schematic overview of AlphaFold model is shown in Fig. 5, which it takes as input a protein sequence and produces as output, among others, the predicted 3D structure, as well as the confidence estimates of its prediction, pTM and pLDDT, which measure the estimated confidence of how well the predicted and ground truth structures match.

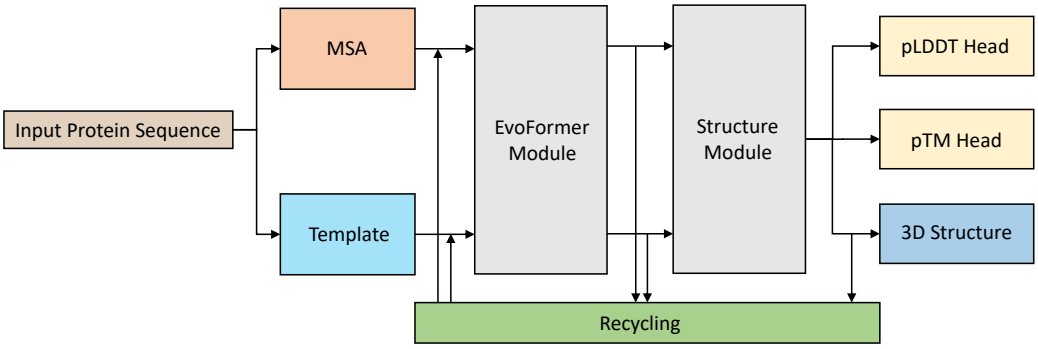

Figure 5: Overview of the inference stage in AlphaFold model. Given an input protein sequence, first the search is performed in genetic database to find similar sequences and construct multiple sequence alignments (MSA). Then a structure database search is done to find similar 3D structures and construct templates. The MSA and templates are fed into EvoFormer module, whose output is then sent to the Structure module, which is finally completed with the multiple output heads. The 3D structure head generates predicted 3D protein structure, while pLDDT and pTM heads estimate the confidence of the computed structure. Optionally, the generated structure together with the intermediate states are recycled and sent back to update/correct MSA and template representations for further processing and improvement.

C AFDistill Training

Tables 3, 4 show the validation performance of AFDistill trained on each of the (p)TM-based and (p)LDDT-based datasets, respectively. Table 5 shows results on (p)LDDT chain-based datasets. Note that (p)LDDT chain is the dataset, similar to (p)TM datasets, where for each sequence we associate a single scalar, in this case the average of all the per-reside (p)LDDT values.

Table 3: Validation CE loss for the AFDistill model trained on each of the (p)TM-based datasets. To address data imbalance during training, we employed weighted sampling for minibatch generation to so that the TM-scores cover their range (0,1) close to uniform distribution. Moreover, we also used Focal loss [21] in place of the standard cross-entropy (CE) loss (the evaluation is still done using CE loss across all the training setups). Based on the validation loss, we see that the AFDistill model trained on TM 42K dataset performed the best, followed by the dataset with augmentations, and the synthetic performed the worst. We also see that weighted sampling and focal loss do help in addressing the data imbalance problem, although for TM augmented 86K, the balanced augmentation seemed to help better and the best performance was for the case when no weighted sampling is applied and the traditional CE loss is used. As shown in Section 3, the validation performance on the distillation data may not always indicate the performance on the downstream applications, where in particular we observed that the Distill model, trained on TM augmented 86K dataset, overall performed better than TM 42K, while having slightly worse validation CE loss.

Data	Training		Validation
	Weighted sampling	Focal loss (γ)	CE loss
TM 42K	–	–	1.33
	+	–	1.37
	+	1.0	1.16
	+	3.0	**1.10**
	+	10.0	1.29
TM augmented 86K	–	–	**2.12**
	+	1.0	2.15
	+	3.0	2.19
	+	10.0	2.25
pTM synthetic 1M	–	–	2.90
	+	1.0	2.75
	+	3.0	**2.55**
	+	10.0	3.20

Table 4: Validation CE loss for AFDistill trained on each of the (p)LDDT-based datasets. We see that weighted sampling coupled with Focal loss, performed the best.

Data	Training		Validation
	Weighted sampling	Focal loss (γ)	CE loss
LDDT 42K	-	-	3.47
	+	1.0	3.44
	+	3.0	3.42
	+	10.0	**3.39**
pLDDT 1M	-	-	3.27
	+	1.0	3.28
	+	3.0	**3.25**
	+	10.0	3.24

D AFDistill scatter plots of predictions

In Fig. 6 we show scatter plots of the true vs pTM scores and pLDDT values on the entire validation set. We see a clear diagonal pattern in both plots, where the predicted and true values match. There are also some number of incorrect predictions (reflected along the off-diagonal), where we see that for the true scores in the upper range, the predicted scores are lower, indicating that AFDistill tends to underestimate them.

Table 5: Validation CE loss for the AFDistill model trained on each of the (p)LDDT chain-based datasets. (p)LDDT chain is the dataset, similar to (p)TM datasets, where for each sequence we associate a single scalar, in this case the average of all the per-reside (p)LDDT values. Similar as before, we see that the use of weighted sampling coupled with Focal loss helps in boosting the model performance. We also see that increasing the scale of data (which is already balanced) improves the performance even further.

Data	Training		Validation
	Weighted Sampling	Focal loss (γ)	CE loss
LDDT chain 42K	-	-	3.69
	+	1.0	3.57
	+	3.0	3.63
	+	10.0	**3.59**
pLDDT chain 1M	-	-	3.29
	+	1.0	3.36
	+	3.0	**3.30**
	+	10.0	3.31
pLDDT chain balanced 1M	–	–	2.45
pLDDT chain balanced 10M	–	–	2.24
pLDDT chain balanced 60M	–	–	**2.21**

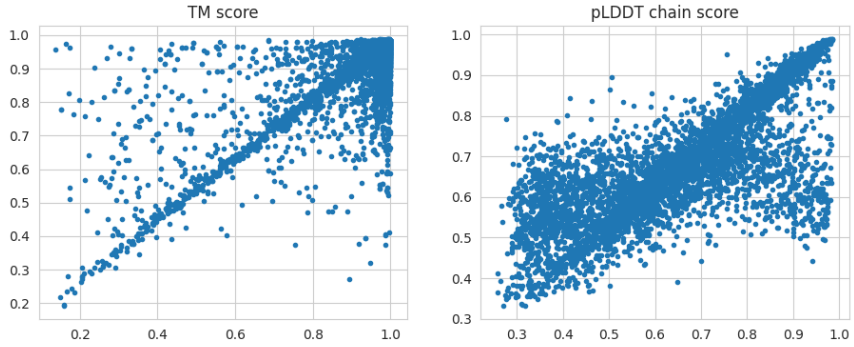

Figure 6: A scatter plot of the true and predicted TM score (left panel, data: TM 42K) and pLDDT (right panel, data: pLDDT bal 60M) for the data presented in main text in Fig. 2.

E Architectural and Training Details

E.1 AFDistill

Table 6 shows architectural details of AFDistill and ProtBert, while in Table 7 we present training details for AFDistill for two experimental setups. In all the experiments we used A100 GPUs. From the tables we can see that the AFDistill (420 M parameters) training takes approximately 24 hours on 1 GPU for TM 42K dataset, and 7 days on 8 GPUs for pLDDT balanced 60M dataset. Note that AFDistill training cost is amortized: the model is trained once and reused it many downstream applications. During training of the downstream applications, AFDistill model needs to be kept in memory to compute SC (structural consistency) score.

E.2 Protein Design

Table 8 shows training details for GVP and ProteinMPNN models. Additionally, we note that for the original PiFold model it takes 60 epochs (6 hours on 1 GPU) to train the model, while for PiFold+SC it takes 60 epochs (8 hours on 1 GPU) to do the training. The increased training time is due to

Table 6: Architectural details of AFDistill and ProtBert (which is used to initialize AFDistill training).

Model	Number of parameters	Number of layers	Hidden layer size	Number of heads	Vocab size	Pretraining Data	Reference
ProtBert	420M	30	1024	16	30 (20 amino acids + 10 aux tokens)	BFD100 (572 GB, 2B proteins) Uniref100 (150 GB, 216M proteins)	[8]
AFDistill	420M	30	1024	16	50 (50 bins, TM/pLDDT (0,1))	TM 42K (20 MB, 42K sequences) pLDDT balanced 60M (100 GB, 60M sequences)	–

Table 7: Training details for AFDistill for two experimental setups: small - using TM 42K dataset, and large - using AFDistill pLDDT balanced 60M.

Model	Learning rate	Batch size	Optimizer	GPUs	Training time
AFDistill TM 42K	$1e^{-6}$	10	Adam	1 × A100, 40GB	1 day
AFDistill pLDDT bal 60M	$1e^{-6}$	10	Adam	8 × A100, 40GB	7 days

frequent validations (which involves sampling 100 samples per sequence for recovery and diversity computations). Note, that once the downstream application is trained, AFDistill is not used during inference.

Table 8: Training details for GVP and ProteinMPNN (original, as well as SC-regularized using our AFDistill model). Note that although AFDistill has 420M parameters, these are not part of the learnable model parameters, therefore are not counted towards the total.

Setup	Parameters	Learning rate	Batch size	Optimizer	GPUs	Training time
GVP / GVP+SC	1M	$1e^{-3}$	3000 res/batch	Adam	1 × A100, 40GB	1 day
ProteinMPNN / ProteinMPNN + SC	1.6M	varied	5000 res/batch	Adam	1 × A100, 40GB	2 days

F GVP Training Details

An example of GVP training progress regularized by the structure consistency (SC) score computed by the AFDistill model (pre-trained on various (p)TM-based datasets) is shown in Fig. 7. This figure shows that although SC score may be less accurate on the absolute scale, on the relative scale we can see it accurately detecting decays and improvements in the sequence quality as the GVP trains. Similarly, in Fig. 8 we show scatter plots of estimated pTM versus true TM scorefor GVP-generated protein sequences regularized by SC score.

F.1 Effect Of Using AFDistill Trained From Scratch

We also experimented with AFDistill models trained from scratch (as opposed to starting from pre-trained ProtBert), but observed worse performance. As an example, we trained AFDistill from scratch on TM42K dataset. The validation CE loss during distillation was 1.5 (versus 1.1 when using pre-trained ProtBert model). Moreover, training of AFDistill model from scratch takes longer (3 days vs 1 day). When regularizing GVP with AFDistill from scratch, we get similar recovery rate (39.4 vs 39.6) but lower sequence diversity (15.9 vs 21.1), which confirms the benefit of common practice of fine-tuning the pretrained models as opposed to starting from random models weights.

F.2 Effect of Structure Consistency (SC) Score On GVP Performance

For protein design (e.g., using GVP as a base model) the objective is CE + SC (cross-entropy + AFDistill structure consistency score). In Fig. 9 we present the effect of SC magnitude on the GVP performance on the test set of CATH dataset. As can be seen, when only the CE term is present (the blue left most bar in both panels, representing the original GVP), the model is encouraged to recover the specific ground truth protein sequence for a given 3D structure, and this promotes model accuracy, and high amino acid recovery rate, while also resulting in low diversity. On the other hand,

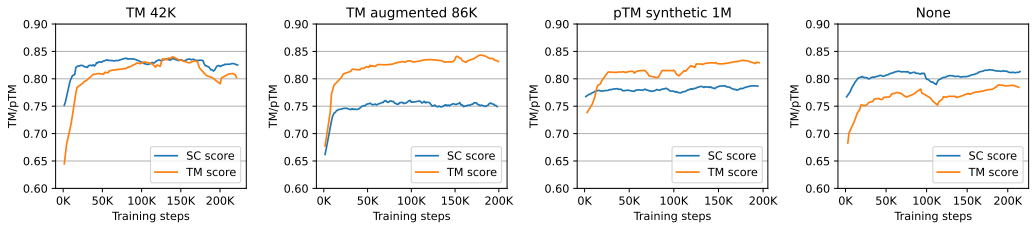

Figure 7: Example of the training progress (on CATH 4.2 dataset) of the GVP model regularized by the structure consistency (SC) score computed by the AFDistill model pre-trained on different datasets. Each plot shows the results for one of the Distill pre-training datasets, where the blue line represents the SC score computed by the AFDistill model (in this case generating pTM value), while the orange line shows the actual TM score computed between the ground truth structure and the AlphaFold's estimated 3D structures for a GVP-generated protein sequences. The last plot on the right shows the original, unregularized GVP training, where SC score was computed but never applied as part of the loss. It can be seen that SC correlates well with the TM score for TM 42K, while for others (TM augmented 86K and pTM synthetic 1M datasets) it tends to underestimate true TM score. Therefore, SC score may be less accurate on the absolute scale, while on the relative scale we can see that it can accurately detect decays and improvements in the sequence quality as the GVP trains. And the latter is of particular importance for SC to be a regularization loss during training, since it can clearly identify the ill-generated protein sequences early in the training (lower SC scores) and recognize well-defined sequences later during the training (higher SC scores).

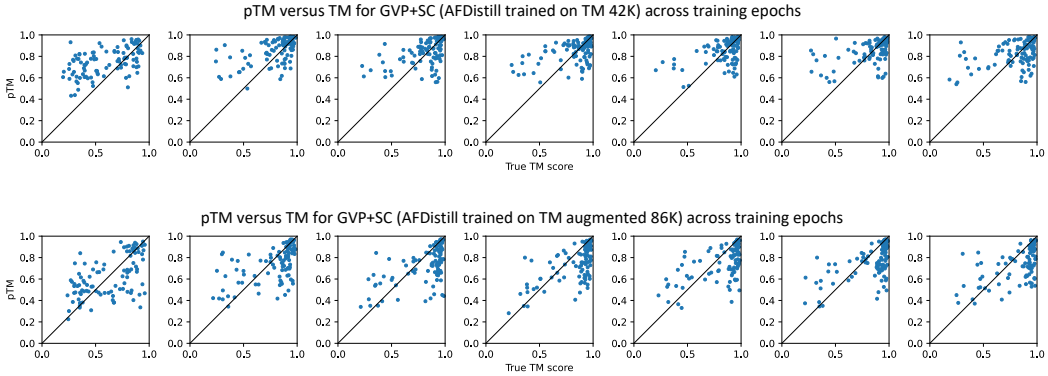

Figure 8: Estimated pTM versus true TM score (based on AlphaFold structure prediction) for GVP-generated protein sequences regularized by SC score. The top row shows results for SC computed by AFDistill model trained on TM 42K, while the bottom row is for AFDistill trained on TM augmented 86K. The columns in each row correspond to the progress as GVP trains. Note that the top row corresponds to the first left plot in Fig. 7, while the bottom row corresponds to the second plot in Fig. 7. It can be observed that in the earlier stages of GVP training, the generated protein sequences are of poor quality, reflected in pTM and TM scores that are spread across the (0,1) range. On the other hand, as the training progresses, the generated sequences are getting better and the pTM/TM score is concentrated more in the upper range. Another observation is that for AFDistill trained on TM 42K dataset, the predicted and true TM score are better aligned across the diagonal (compare with orange and blue lines on the left plot in Fig. 7), while for AFDistill trained on TM augmented 86K dataset, pTM tends to underestimate true TM score. These plots show that AFDistill is viable sequence scoring tool, which fairly accurately measures the structural consistency of the generated protein sequences. Combined with the fact that it is fast and end-to-end differentiable, shows its potential for many of the protein optimization problems.

when only the SC term is present (the pink right most bar, reprenting CE+32*SC, i.e., when SC completely dominates and CE can be ignored), this results in poor and degenerated protein sequences. This is expected, since AFDistill alone cannot guide GVP which sequence it should generate to match the given input 3D structure. Recall, that AFDistill has no information about the structure, and since many of the relevant protein sequences can have high pTM/pLDDT, all of them could be good

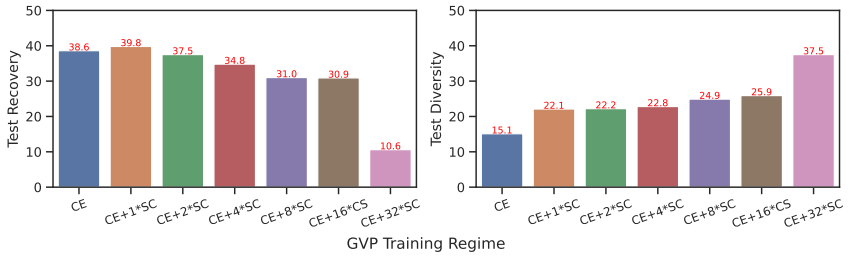

Figure 9: The effect of Structure Consistency (SC) loss on the performance of GVP. Left panel shows the amino acid recovery rate and the right panel shows the diveristy rate on the test set of CATH dataset. The horizontal y-axis shows the different choices of objective function during training: CE is the cross-entropy loss, SC is the Structure Consistency score computed by AFDistill.

	Model	Recovery	Recovery Change	Diversity	Diversity Change
1	GVP-GNN (1M) [17]	40.2	–	NA	–
2	GVP-GNN (1M) [14]	42.2	–	NA	–
3	GVP-GNN (1M) + AlphaFold2 data [14]	38.6	-3.6 (-8.5%)	NA	–
4	GVP-GNN (1M) (our experiment)	38.6	–	15.1	–
5	GVP-GNN (1M) + SC (our experiment)	39.6	+1.0 (+2.6%)	21.1	+6.0(+39.7%)
6	GVP-GNN-large (21M) [14]	39.2	–	NA	–
7	GVP-GNN-large (21M) (our experiment)	39.0	–	16.7	–
8	GVP-GNN-large (21M) + SC (our experiment)	40.1	+1.1(+2.8%)	19.3	+2.6(+15.6%)

Table 9: Comparison of amino acid recovery rate of protein sequences generated by GVP on the test split of CATH dataset. First row is the original result from GVP authors, rows 2 and 3 show the results from ESM authors, and row 4 shows the result from our experiments. A small difference between the values in first, second and forth rows can be attributed to some discrepancies in experimental settings as well as model initialization. We can see that a simple data augmentation baseline results in 3.6 (or 8.5%) drop of recovery relative to the unaugmented GVP (1M). On the other hand, the use of SC regularization leads to 1.0 (or 2.6%) gain in recovery, signaling the benefit of the proposed distillation approach. On GVP-GNN-large (21M) model, shown in rows 6, 7 and 8 we were able to recover results closer to the published ones (39.0 vs their 39.2). And when SC is applied, we again see a boost in recovery (40.1 vs 39.0), and diversity (19.3 vs 16.7).

candidates, and this promotes high diversity and low recovery. Consequently, when both CE and SC terms are present and when appropriate balance between them is found (in our case it is CE+SC, corresponding to the orange bar in both panels), we get a full benefit, i.e., the accurate recovery and high diversity of the generated protein sequences.

G Additional Performance Comparisons of SC Regularization

G.1 ESM-IF

In this Section we compare GVP-GNN (1M and 21M) performance under different training scenarios (CATH only and CATH + AlphaFold2 data) and present the results in Tables 9 and 10. The first row in Table 9 is the recovery rate of the original GVP-GNN (1M) model as reported in [17]. The following two rows (2 and 3) are the results presented in the work of [14] (ESM-IF). Their evaluation showed that the vanilla GVP achieved a slightly higher recovery rate of 42.2. GVP+AlphaFold2 represents the GVP trained on augmented dataset (CATH + AlphaFold2-generated structure/sequence pairs). Interestingly, this simple data augmentation baseline showed worse performance as compared

	Model	Recovery	Recovery Change	Diversity	Diversity Change
1	GVP-GNN-large (21M) [14]	50.8	–	NA	–
2	GVP-GNN-large (21M) (our experiment)	50.5	–	13.8	–
3	GVP-GNN-large (21M) + SC (our experiment)	50.9	+0.4(+0.8%)	18.5	+4.7(+34.0%)

Table 10: Comparison of amino acid recovery rate of protein sequences generated by GVP-GNN-large (21M) on the test split of CATH dataset, while trained on CATH + AlphaFold2 dataset (12M sequences). As observed before on experiment in GVP-GNN-large (21M) + SC when trained on CATH only, here when SC is applied, we see a minor boost in recovery (40.9 vs 50.5), and more significant increase in protein sequence diversity (13.8 vs 18.5).

to the original GVP, and the authors had to significantly increase GVP capacity (from 1M to 21M) to get any benefit from the data augmentation. Moreover, note that such a data augmentation idea can also serve as the baseline for our approach of AFDistill regularization, since AFDistill was trained on AlphaFold2-generated data and it can be thought of as a compressed representation of that data.

The rows (4 and 5) show our evaluation results of the vanilla GVP-GNN (1M), achieving slightly lower base recovery rate of 38.6, while this same GVP but trained with AFDistill regularization achieves a boost in recovery (39.6) and significant increase in the sequence diversity (+39.7% as we showed in Fig. 4). On GVP-GNN-large (21M) model we were able to recover results closer to the published ones (39.0 vs their 39.2). And when SC is applied we again see a boost in recovery (40.1 vs 39.0), and diversity (19.3 vs 16.7).

Finally, in Table 10 we followed the setup of [14] and trained GVP-GNN-large (21M) on large dataset of CATH+AlphaFold2 (12M sequences) and evaluated on CATH test set. The second row in the table shows our that our experiment recovered results similar to the ones reported in [14], while in third row we present the SC-regularized training using our AFDistill model. Clearly, the sequence recovery was improved and even more so the diversity of the generated sequences went up from 13.8 to 18.5.

Therefore, comparing data augmentation and model distillation for the task of protein design, we see that for the GVP models (1M and 21M), AFDistill offers a clear advantage, providing a modest boost in recovery, while significantly increasing diversity of the generated sequences. This improvement after applying SC regularization occurs because the baseline techniques, which rely on CE in training, primarily emphasize accurate sequence recovery, neglecting other protein sequences that can achieve the desired structure. SC regularization encourages the consideration of many relevant and diverse protein sequences with high pTM/pLDDT scores as strong candidate sequences. This results in a moderate improvement in recovery and a significantly larger boost in diversity. Moreover, the distillation overhead is amortized, as we train AFDistill once and use it in many downstream applications. The data augmentation would require additional computational cost in every downstream application.

G.2 Graph Transformer

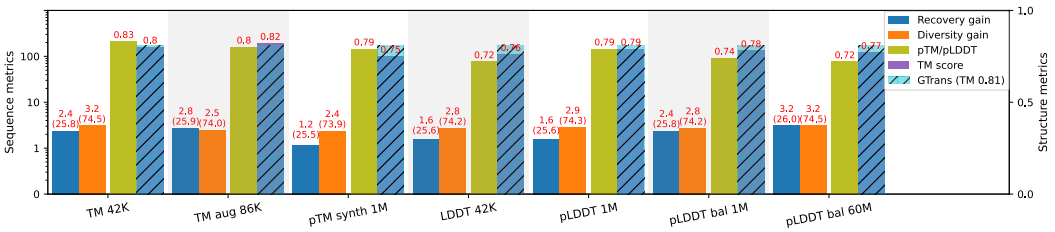

Figure 10: Evaluation results of Graph Transformer model trained with SC score regularization. Baseline model with no regularization achieves 25.2 in recovery, 72.2 in diversity and 0.81 in TM score on the test set.

We evaluated the effect of SC score on Graph Transformer [27], another inverse folding model, which seeks to improve standard GNNs to represent the protein 3D structure. Graph Transformer applies a permutation-invariant transformer module after GNN module to better represent the long-range pair-wise interactions between the graph nodes. The results of augmenting Graph Transformer training with SC score regularization are shown in Fig. 10. Baseline model with no regularization has 25.2 in recovery, 72.2 in diversity and 0.81 in TM score on the test set. As compared to GVP (Fig. 4), we can see that for this model, the recovery and diversity gains upon SC regularization are smaller. We also see that TM score of regularized model (TM 42K and TM augmented 86K pretraining) is slightly higher as compared to pLDDT-based models.

G.3 Protein Infilling

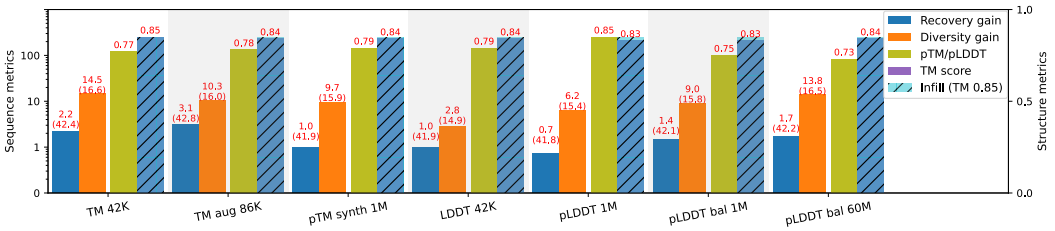

Figure 11: Evaluation results of Protein Infilling model trained with SC regularization. Baseline model achieves 41.5 in recovery, 14.5 in diversity and 0.80 in TM score on the test set. Similar as for the other applications, we see an improvement in the sequence recovery and even bigger gain in diversity. TM score shows that the resulting 3D structure remains close to the original, confirming the benefit of using SC score for training regularization.

Our proposed structure consistency regularization is quite general and not limited to the inverse folding task. Here we show its application on protein infilling task. Recall, that while the inverse folding task considers generating the entire protein sequence, conditioned on a given structure, infilling focuses on filling specific regions of a protein conditioned on a sequence/structure template. The complementarity-determining regions (CDRs) of an antibody protein are of particular interest as they determine the antigen binding affinity and specificity. We follow the framework of [16] which formulates the problem as generation of the CDRs conditioned on a fixed framework region. We focus on CDR-H3 and use a baseline pretrained protein model ProtBERT [9] finetuned on the infilling dataset, and use ProtBERT+SC as an alternative (finetuned with SC regularization). The CDR-H3 is masked and the objective is to reconstruct it using the rest of the protein sequence as a template. The results are shown in Fig. 11. Baseline model achieves 41.5 in recovery, 14.5 in diversity, and 0.80 in TM score on the test set. Similar as for the other applications, we see an improvement in the sequence recovery and even bigger gain in diversity, while using the AFDistill pretrained on TM 42K and TM augmented 86K, together with the pLDDT balanced datasets. TM score shows that the resulting 3D structure remains close to the original, confirming the benefit of using SC for training regularization.