nature portfolio

https://doi.org/10.1038/s43588-021-00098-9

Supplementary information

Fast and effective protein model refinement using deep graph neural networks

In the format provided by the authors and unedited

Fast and effective protein model refinement by deep graph neural networks

Supplementary information

Xiaoyang Jing, Jinbo Xu

Toyota Technological Institute at Chicago, Chicago, IL 60637, USA

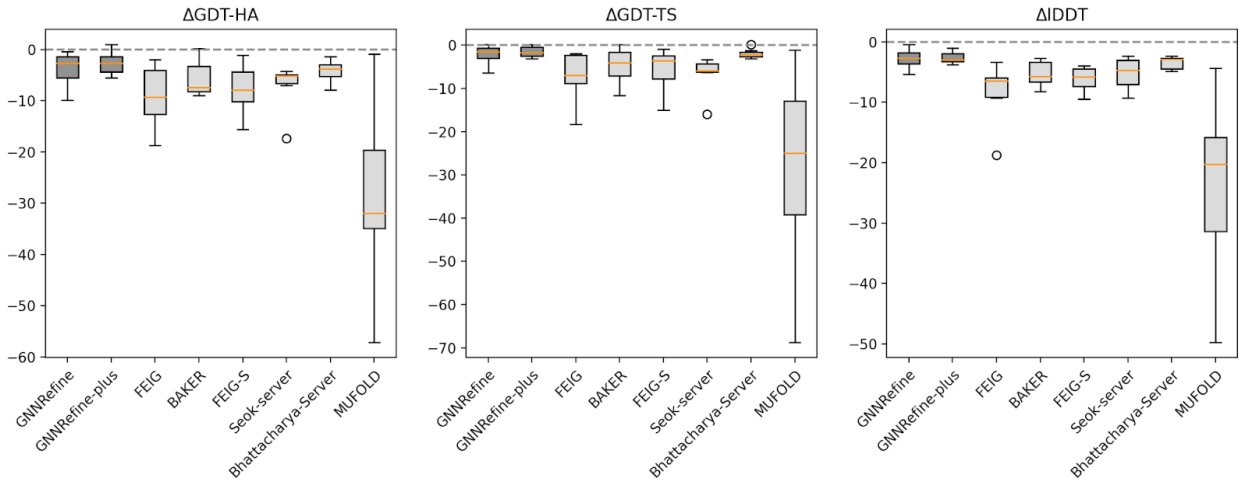
1. Performance on the CASP14 targets

Supplementary Table 1. Performance on the CASP14 refinement targets when the AlphaFold2 models are excluded. GNNRefine and GNNRefine-plus generate 5 and 50 refined models, respectively, for each starting model. Only the first-ranked refined models are evaluated in this table. The performance of our methods is highlighted boldly.

Type	Methods	GDT-HA	GDT-TS	lDDT	Degradation		
					0	-1	-2
	Starting	50.40	69.6	62.51			
Human	FEIG	+4.59	+3.48	+3.36	7	5	2
	BAKER	+2.71	+1.06	+2.34	11	9	8
Server	**GNNRefine**	**+1.94**	**+1.51**	**+1.26**	**10**	**3**	**3**
	GNNRefine-plus	**+1.63**	**+1.29**	**+1.35**	**8**	**4**	**2**
	FEIG-S	+3.75	+2.64	+2.86	8	7	5
	Seok-server	+0.27	-0.09	+0.58	14	8	4
	Bhattacharya-Server	-0.54	-0.39	+0.27	22	15	4
	MUFOLD	-12.32	-15.45	-10.75	29	29	26

Supplementary Table 2. Performance on the AlphaFold2 refinement models in CASP14. The performance of our methods is highlighted boldly.

Type	Methods	GDT-HA	GDT-TS	lDDT	Degradation		
					ΔGDT-HA< -5	ΔGDT-TS < -5	ΔlDDT < -5
	Starting	70.13	85.56	80.84			
Human	FEIG	-9.04	-7.06	-8.39	4	4	6
	BAKER	-5.61	-4.69	-5.26	5	3	4
Server	GNNRefine	**-3.84**	**-2.14**	**-2.76**	**2**	**1**	**1**
	GNNRefine-plus	**-2.69**	**-1.49**	**-2.24**	**1**	**0**	**0**
	FEIG-S	-7.67	-5.73	-6.14	5	3	4
	Seok-server	-7.16	-6.59	-5.23	5	5	3
	Bhattacharya-Server	-4.23	-1.96	-3.51	2	0	0
	MUFOLD	-28.44	-28.47	-24.12	6	5	7



Supplementary Figure 1. Box plot of the distribution of ΔGDT-HA, ΔGDT-TS, and ΔlDDT values on the AlphaFold2 refinement models.

2. Performance on the AlphaFold2 regular models in CASP14

We tested our method on all the first models submitted by AlphaFold2 for regular targets in CASP14, the result is shown in Table 3. As shown in the table, since most of the AlphaFold2 models are of high quality, on average our method GNNRefine degrades the AlphaFold2 models by -2.22, -1.13 and -2.12 in terms of GDT-HA, GDT-TS and lDDT, respectively. GNNRefine-plus degrades the quality by -2.40, -1.24 and -2.04, respectively. As shown in Figure 2, our method performs slightly better on initial protein models of lower quality that predicted by

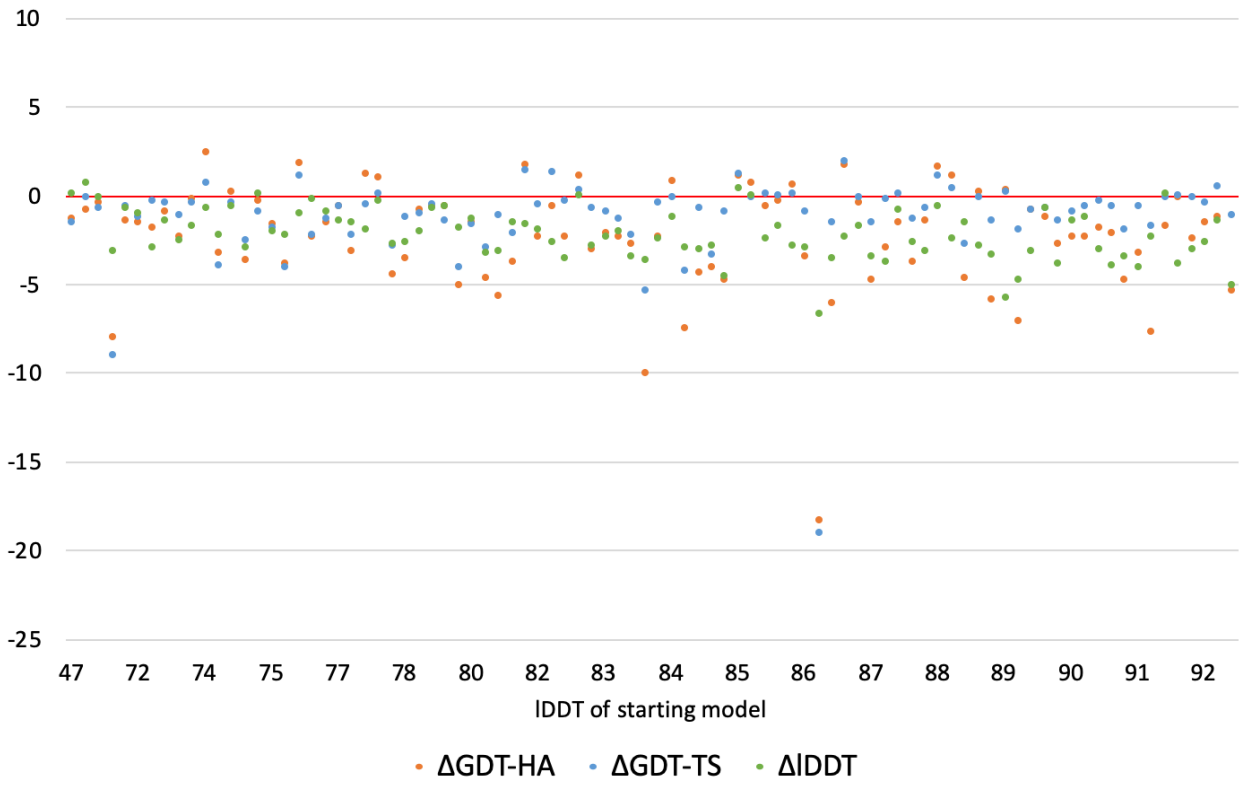
AlphaFold2, while not as good on initial protein models of very high quality. Our methods improve GDT-HA of the AlphaFold2 first models for T1100-D1, T1055-D1, T1095-D1, T1096-D1, T1094-D2, T1046s2-D1, T1038-D1, T1030-D2, T1061-D1, T1030-D1, T1100-D2, T1061-D0, T1070-D4, T1047s2-D1, T1053-D1, T1037-D1, T1092-D1 and T1034-D1. In particular, our method GNNRefine-plus may improve the GDT-HA of T1100-D1 and T1055-D1 by ~4 units, which have starting GDT-HA 58.70 and 68.00, respectively.

On the AlphaFold2 first models, the correlation coefficient between the GNNQA-predicted quality of the starting model and the improvement by GNNRefine is 0.31, although we can only improve a smaller percentage of AlphaFold2 models. As shown in Fig. 2, all the AlphaFold2 models that can be improved by GNNRefine have lDDT less than 88.

Supplementary Table 3. Performance on the AlphaFold2 regular models[1] in CASP14

Type	Num	Methods	GDT-HA	GDT-TS	lDDT	Degradation		
						ΔGDT-HA < -5	ΔGDT-TS < -5	ΔlDDT < -5
FM	23	GNNRefine	-2.16	-1.64	-1.75	2	1	0
		GNNRefine-plus	-2.28	-1.80	-1.64	3	1	0
FM/TBM	14	GNNRefine	-3.36	-1.84	-2.54	4	1	1
		GNNRefine-plus	-3.39	-1.90	-2.37	5	1	2
TBM-hard	28	GNNRefine	-1.43	-0.50	-1.89	1	0	1
		GNNRefine-plus	-1.68	-0.64	-1.76	3	0	1
TBM-easy	22	GNNRefine	-2.65	-0.99	-2.58	4	1	0
		GNNRefine-plus	-2.94	-1.07	-2.63	5	1	1
MultiDom	1	GNNRefine	-0.10	-0.30	-1.61	0	0	0
		GNNRefine-plus	+0.60	+0.20	-1.43	0	0	0
All	88	GNNRefine	-2.22 (+2.50/-18.20)[2]	-1.13 (+2.00/-19.00)	-2.12 (+0.77/-6.62)	11	3	2
		GNNRefine-plus	-2.40 (+4.30/-16.90)	-1.24 (+3.80/-21.00)	-2.04 (+1.75/ -6.56)	16	3	4

1. The refinement was conducted on the first models submitted by AlphaFold2 for the whole target, while the evaluation is conducted on the 88 official domains defined by CASP14.
2. The corresponding maximum and minimum values.



Supplementary Figure 2. The model quality improvement by GNNRefine with respect to the starting model quality (of the AlphaFold2 regular models). The data are sorted ascendingly by the lDDT of the starting models.

3. Performance on the CAMEO targets

Supplementary Table 4. Performance of standalone software on the CAMEO targets.Bold values are the best performance on the corresponding metric.

Methods	GDT-HA	GDT-TS	lDDT	Degradation		
				0	-1	-2
Starting	45.55	63.44	60.87			
GNNRefine	+1.91	+1.18	**+2.25**	42	26	17
GNNRefine-plus	**+1.99**	**+1.20**	+2.23	**39**	21	12
GalaxyRefine	+0.03	+0.09	+0.96	98	61	37
ModRefiner-100	-0.02	-0.02	+0.44	90	**11**	**3**
ModRefiner-50	-0.07	-0.05	+0.44	95	27	7
ModRefiner-0	-0.72	-0.83	+0.25	128	59	25

FastRelax	-2.30	-2.28	-0.19	139	121	92

Tested on the CAMEO targets, there is a weak correlation (0.20) between the GNNQA-predicted quality of the starting model and the improvement by GNNRefine. As shown in Fig 3, most of the CAMEO starting models have GNNQA-predicted quality (lDDT) from 47 to 80 and GNNRefine may refine most of them. GNNRefine has a slightly better chance to improve the quality of CAMEO targets when the GNNQA-predicted lDDT score is between 60 and 80.

Supplementary Figure 3. The model quality improvement by GNNRefine with respect to the GNNQA-predicted quality of the starting model (of the CAMEO set). The data are sorted ascendingly by the GNNQA-predicted global lDDT of the starting models.

Supplementary Table 5. The GNNQA-predicted quality of starting models vs. the quality improvements by GNNRefine on the CAMEO targets.

GNNQA-predicted lDDT of starting model	# starting models	Improvement		
		ΔGDT-HA	ΔGDT-TS	ΔlDDT
>40	206	+2.00	+1.32	+2.26
>50	186	+2.11	+1.42	+2.24
>60	121	+2.38	+1.66	+2.19
>70	46	+2.48	+1.61	+2.26
>80	8	+1.60	+1.08	+2.00

4. Performance on the in-house CASP13 FM models

Supplementary Table 6. Performance on our in-house decoys for the CASP 13 FM target. For each target, 150 initial models were generated by our own template-free modeling method and then for each initial model, one refined model was generated by GNNRefine. Bold values are the improvement of refined models.

Models	GDT-HA		GDT-TS		lDDT		Degradation		
	Starting	Refined	Starting	Refined	Starting	Refined	0	-1	-2
Max[1]	40.46	**+3.44**	59.27	**+2.79**	52.02	**+6.43**	1	0	0
Cluster10[2]	32.38	**+2.13**	48.49	**+2.01**	46.93	**+4.20**	9	5	1
Mean[3]	34.85	**+1.18**	51.81	**+0.71**	48.24	**+3.87**	8	1	1

1. The average quality of the best models (ranked by GDT-HA) generated for each target.
2. All models of a target are clustered into 10 groups by SPICKER[1]. This row shows the average quality of the 10 cluster centroids.
3. The mean quality of all the models for each target.

5. Performance of standalone software and servers on the CASP13 refinement targets

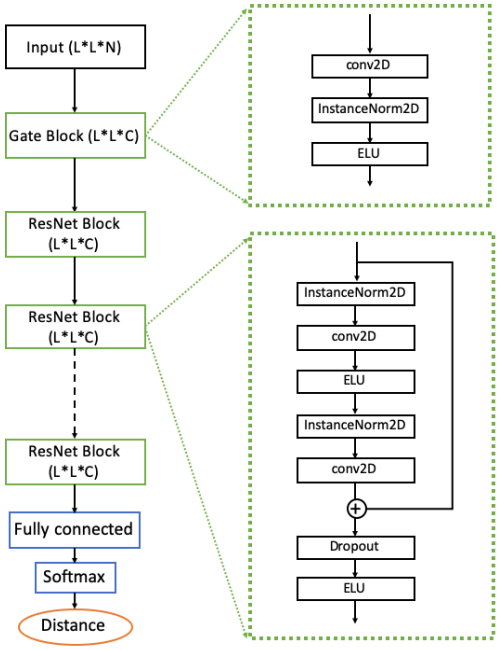
Supplementary Table 7. Performance of standalone software and servers on the CASP13 refinement targets. Bold values are the best performance on the corresponding metric.

Methods	GDT-HA	GDT-TS	lDDT	Degradation			Running Time[1]
				0	-1	-2	
Starting	52.27	71.51	61.74				
GNNRefine	**+3.83**	**+2.31**	**+3.19**	**3**	1	**0**	**~0.16**
GalaxyRefine	+0.21	+0.05	+1.23	15	8	4	~2.53
ModRefiner-100	+0.16	+0.05	+0.73	13	3	0	~0.57
ModRefiner-50	-0.05	+0.04	+0.78	11	6	0	~0.59
ModRefiner-0	-0.70	-0.58	+0.99	17	12	2	~0.54
FastRelax	-2.00	-1.96	+0.17	18	14	13	~0.03
3DRefine	+0.24	+0.12	+0.30	9	**0**	0	--[2]
ReFold	-0.45	-0.24	-0.04	17	14	6	--[2]

1. The average running time (hour) needed to refine a single protein model. All programs were run on one CPU of the same computer to fairly compare their running time.
2. We could not run these two methods locally and thus, cannot measure their running time accurately.

6. Distance prediction by GNN, in-house ResNet and DeepAccNet

Our in-house 2D ResNet consists of one gate block, 20 ResNet blocks, and one output layer, as shown in Figure 4. It uses the same input feature as GNNRefine. To fit the input shape of 2D ResNet, we pairwisely concatenate the node feature with the edge feature. The gate block is used to connect pairwise input features to the 2D ResNet, which is composed of one 2D convolutional layer, one instance norm layer and one ELU activation layer. Each ResNet block consists of 2 instance norm layers, 2 convolutional layers, 2 ELU activation layers and 1 dropout layer. In each ResNet block, there is a shortcut connecting its input to the output of the second convolutional layer. The dilation ratio is 2 and the kernel size is 5 in each 2D convolutional layer. The inner channel size is 64. The output layer consists of one fully connected layer and one softmax layer. Similar to GNNRefine, our in-house ResNet predicts pairwise distance probability distribution.



Supplementary Figure 4. The detailed architecture of our in-house 2D ResNet for protein model refinement.

Supplementary Table 8. Comparison between predicted distance and the distance calculated from the starting models. Only one GNNRefine deep model is used here. Bold values are the performance of GNNRefine.

Dataset	Methods	Top L Contact Precision (%)		C_β lDDT	
		Medium+Long	Long	Medium+Long	Long
CASP13	Starting	77.33	66.16	67.92	66.38

	GNNRefine	**84.85**	**69.65**	**70.89**	**69.48**
CAMEO	Starting	57.66	49.67	57.77	54.89
	GNNRefine	**65.34**	**54.10**	**60.19**	**57.38**

Supplementary Table 9. Performance of GNN-based and ResNet-based methods on the CASP13 and CASP14 refinement targets. Only 1 instead of 5 GNNRefine deep models is used here.Bold values are the best performance on the corresponding metric.

Dataset	**Methods**	**GDT-HA**	**GDT-TS**	**lDDT**	**Degradation**		
					0	**-1**	**-2**
CASP13	Starting	52.27	71.51	61.74			
	GNNRefine	**+3.15**	**+1.96**	**+2.88**	**1**	**0**	**0**
	2D ResNet (in-house)	+0.49	+0.29	+0.91	8	4	1
	DeepAccNet[1]	+0.07	-1.23	+0.51	11	8	6
CASP14	Starting	54.12	72.65	65.98			
	GNNRefine	**+0.61**	**+0.60**	**+0.44**	**16**	**8**	**5**
	2D ResNet (in-house)	-0.58	-0.51	-0.44	24	14	6
	DeepAccNet[1]	-0.60	-0.85	-0.06	19	16	9

1. Without extensive conformational sampling.

Supplementary Table 10. The improvement in distance prediction by GNN and ResNet on the CASP13 and CASP14 refinement models. Bold values are the best performance on the corresponding metric.

Dataset	**Methods**	**Top L Contact Precision**		**C_β lDDT**	
		Medium+Long	**Long**	**Medium+Long**	**Long**
CASP13	GNN	**+6.99**	**+2.89**	**+3.26**	**+3.59**
	2D ResNet (in-house)	+4.83	+1.06	-1.07	-1.16
	DeepAccNet DistPot[1]	-39.19	-30.14	-5.38	-5.14
CASP14	GNN	**+3.41**	**+1.45**	**+0.27**	**+0.96**
	2D ResNet (in-house)	+1.03	-1.20	-2.00	-1.72
	DeepAccNet DistPot[1]	-28.00	-21.53	-4.89	-4.88

1. The predicted distance by DeepAccNet is derived from its distance potential. For each residue pair, the distance with the lowest potential is used as its predicted distance.

7. Ablation Study of GNNRefine

7.1 The improvement in contact and distance prediction by GNNRefine trained with different input features and training data

Supplementary Table 11. The improvement in distance predicted by GNNRefine with different input features and training data, tested on the CASP13 data.Bold values are the best performance on the corresponding metric.

Features	Training data	Top L Contact Precision		C_β lDDT	
		Medium+Long	Long	Medium+Long	Long
All features	In-house	+7.52	+3.48	+2.98	+3.10
All features	DeepAccNet data	+6.43	+2.59	+3.15	+3.60
All features	CASP models only	+3.94	+1.09	+0.55	+1.11
no Orientation	In-house	+7.38	+2.54	+2.15	+2.22
no Dihedral&SS&RSA	In-house	+6.99	+2.89	**+3.26**	**+3.59**
no AtomEmb	In-house	+7.40	**+4.11**	+3.14	+3.09
AtomEmb (with local frame)[1]	In-house	**+7.64**	+3.30	+2.98	+3.09

1. Using Cα, N, and C to define the reference frame of atom coordinates for each residue.

7.2 Performance of an ensemble of GNNRefine models

Supplementary Table 12. Iterative refinement on the CASP13 targets using 5 different GNNRefine models trained on different datasets. Model 1, 2 and 3 are trained by 3 different splits of our in-house data and DAN 1 and 2 are trained by 2 different splits of the DeepAccNet data.

Methods	GDT-HA	GDT-TS	lDDT	GDT-HA Degradation			lDDT Degradation
				0	-1	-2	
Model 1	+3.06	+1.92	+2.63	2	0	0	0
+ Model 2	+3.34	+2.12	+2.91	3	1	0	1
+ Model 3	+3.39	+2.12	+2.95	3	2	0	1
+ DAN 1	+4.03	+2.47	+3.22	1	0	0	2
+ DAN 2	+4.21	+2.44	+3.31	3	1	0	2

7.3 GNNRefine's performance with different inter-atom relationships predicted

Supplementary Table 13. Model refinement on the CASP13 data using restraints predicted for different atom types. The same set of input features are used here regardless of atom types.

Restraint	GDT-HA	GDT-TS	lDDT	GDT-HA Degradation		
				0	-1	-2
CbCb[1]	+3.15	+1.96	+2.88	1	0	0
CbCb[2]	+2.94	+1.73	+2.55	3	3	0
CaCa[2]	+3.09	+1.95	+2.69	3	2	1
NO[2]	+1.17	+0.47	+1.06	7	2	2
CaCa & CbCb[2]	+3.17	+1.99	+2.73	2	2	1
CaCa & CbCb & NO[2]	+2.52	+1.45	+2.02	4	3	0
CbCb[3]	+2.76	+1.56	+2.45	4	3	1
CbCb & Orientation[3]	+2.76	+1.60	+2.28	4	2	1

1. GNN model trained to predict CbCb distance only.
2. GNN model trained to predict CaCa, CbCb, and NO distances simultaneously.
3. GNN model trained to predict CbCb distance and orientation (ω, θ dihedrals and φ angle) simultaneously.

7.4 Study of distance cutoff for GNN edge definition

Supplementary Table 14. GNNRefine's performance on the CASP13 data with respect to distance cutoff for GNN edge definition[1]

Distance Cutoff	GDT-HA	GDT-TS	lDDT	GDT-HA Degradation		
				0	-1	-2
8Å	+2.99	+1.75	+2.10	4	2	1
10Å	+3.15	+1.96	+2.88	1	0	0
12Å	+1.10	+0.45	+1.42	9	7	1
15Å	+1.44	+0.99	+2.05	8	3	0

1. For each configuration, we trained one GNNRefine model, then generated 10 refined models from each starting model and selected the lowest-energy model as the final refined model.

Supplementary Table 15. Quality of the graph edges derived from initial protein models (distance cutoff =10Å).

Dataset	Precision	Recall	F1

CAMEO	78.96	81.09	79.89
CASP13	81.61	81.65	81.58
CASP14	84.09	83.85	83.90

7.5 Impact of message-passing layers on GNNRefine's performance

Supplementary Table 16. GNNRefine's performance on the CASP13 data with respect to the number of message-passing layers[1]

Number of message-passing layers	GDT-HA	GDT-TS	lDDT	GDT-HA Degradation		
				0	-1	-2
5	+2.16	+1.42	+1.57	4	2	0
8	+2.58	+1.56	+2.01	3	1	0
10	+2.50	+1.72	+2.15	1	1	0
16	+2.01	+1.33	+1.75	4	1	1

1. The models used in this table are trained on our in-house dataset. In each setting, we trained only one GNNRefine model, generated 10 refined models from each starting model and selected the lowest-energy model as the final refined model.

7.6 Impact on performance by the number of refined models

In this subsection, we study the impact on refinement quality by the number of refined models generated by our method. Meanwhile, GNNRefine builds 5 refined models (in serial) for one initial model to be refined. GNNRefine-plus runs GNNRefine 10 times to generate 50 refined models. GNNRefine-250 generates 250 refined models using the following procedure: 1) generate 50 refined models using the first GNNRefine model and keep only the 10 lowest-energy models; 2) generate 5 refined models from each lowest-energy model using the second GNNRefine model to yield 50 new refined models and keep only the 10 lowest-energy models; 3) repeat step 2) until all 5 GNNRefine models have been used. 4) rank the 50 (10*5) lowest-energy models generated by the 5 GNNRefine models using our GNN-based modeling ranking method.

Supplementary Table 17. GNNRefine's performance with respect to the number of refined models generated for each initial model. Bold values are the best performance on the corresponding metric.

Dataset	Methods	#refined models	GDT-HA	GDT-TS	lDDT	Degradation		
						0	-1	-2
CASP13	GNNRefine	5	+3.83	+2.31	+3.19	3	1	0
	GNNRefine-plus	50	+3.90	+2.31	+3.33	4	0	0
	GNNRefine-250	250	**+4.13**	**+2.55**	**+3.33**	**3**	**0**	**0**
CASP14	GNNRefine	5	+0.84	+0.82	+0.50	17	9	7
	GNNRefine-plus	50	+0.80	+0.77	+0.67	**14**	**10**	**6**

	GNNRefine-250	250	**+1.00**	**+0.86**	**+0.79**	16	14	7

8. GNN-based model quality assessment and order of refinement

8.1 Performance of GNN-based model selection

Our GNN-based quality assessment method (denoted as GNNQA) has a similar network architecture as GNNRefine. The only difference is that GNNQA is trained to predict global and local lDDT based on the node feature while the GNNRefine is trained to predict distance probability distribution based on the edge feature. As shown in Table 18, GNNQA has comparable performance as two recently developed deep learning-based quality assessment methods DeepAccNet[2] and VoroCNN[3], and outperforms a statistical potential RWplus[4]. Table 19 shows that GNNQA can select decoy models with better quality.

Supplementary Table 18. Model selection for GNNRefine using different QA methods on the CASP13 targets.

QA Method	**GDT-HA**	**GDT-TS**	**lDDT**
Max[1]	57.04	74.52	65.35
GNNQA	56.10	73.81	64.92
DeepAccNet	56.16	73.87	64.96
VoroCNN	56.13	73.86	64.82
RWplus	55.92	73.76	64.88
Min[1]	54.75	72.89	64.09

1. "Max" and "Min" represent the average quality of the best and worst models among all refined models generated by GNNRefine.

Supplementary Table 19. Model selection for GNNRefine-plus using GNNQA on the CASP13 targets.

QA Method	**GDT-HA**	**GDT-TS**	**lDDT**
Max	56.94	74.38	65.48
GNNQA	56.17	73.82	65.06
Min	54.89	72.98	64.34

8.2 Impact of the order of refinement

To evaluate the effect of refinement orders (i.e. the order of 5 GNNRefine models), we randomly generated 10 different refinement orders to implement the refinement process and did statistical analysis on the final refined model quality, as shown in Table 20. Possibly because we use GNNQA to rank the refined models, the standard deviations of the quality improvement are small (<0.2), which means the final refined models are robust to the order of refinement.

Supplementary Table 20. Statistical analysis of 10 random different refinement orders on the CASP13 targets.

Metric	Max	Min	Mean	Std
GDT-HA	4.03	3.33	3.57	±0.20
GDT-TS	2.44	1.75	2.06	±0.18
lDDT	3.60	3.12	3.28	±0.13

9. Structure change after refinement

To measure the structure change by a refinement method, we calculate the average distance deviation (in Angstroms) between the starting models and the refined models. To evaluate the performance of different methods on largely deviated regions, we calculate the average change of unreliable local regions (ULRs)[5] proportion (i.e. the proportion of residues in ULRs) between the starting models and refined models.

Supplementary Table 21. The distance deviation and ULR proportion change after refinement. For URL proportion change, a negative and positive value indicates decrease and increase of URL proportion, respectively. Bold values are the largest changes on the corresponding metric.

CASP13			CASP14		
Methods	**Distance deviation (Å)**	**ULR proportion (%)**	**Methods**	**Distance deviation (Å)**	**ULR proportion (%)**
Starting	0.00	17.57	Starting	0.00	19.10
FEIGLAB	1.28	-2.19	FEIG	1.62	-0.28
BAKER	2.40	+1.28	BAKER	3.04	+5.87
GNNRefine	**1.21**	**-0.71**	**GNNRefine**	**0.98**	**-0.19**
GNNRefine-plus	**1.24**	**-1.14**	**GNNRefine-plus**	**1.02**	**-0.72**
Seok-server	1.21	-1.33	FEIG-S	1.45	+0.06
Bhattacharya-Server	0.93	-0.37	Seok-server	1.37	+1.68
YASARA	1.23	+1.57	Bhattacharya-Server	0.53	+0.45

MUFold_server	2.62	+5.38	MUFOLD	9.02	+29.65
3DCNN	1.78	+2.67			

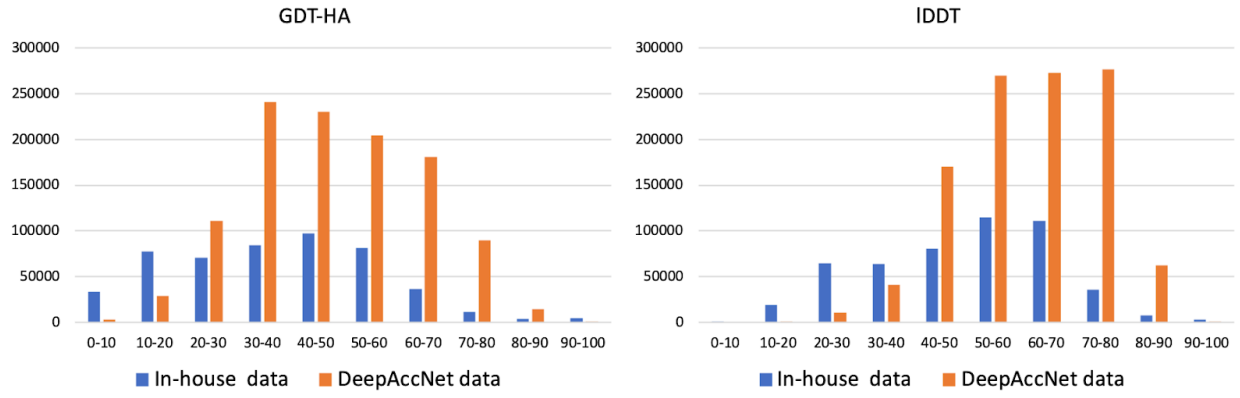
10. Dataset description

10.1 The number of protein models used for training and test

Supplementary Table 22. The number of protein targets and models of different datasets

Dataset	Source	#Targets		#Decoy or starting models	
In-house training data	CASP 7-12	592	29455	115857	500255
	CATH	28863		384398	
DeepAccNet training data	PISCES	7992		1104080	
Test	CASP13	28		28	
	CASP14	37		37	
	CAMEO	208		208	
	CASP13 FM	28		4193	

10.2 The distribution of the training protein models



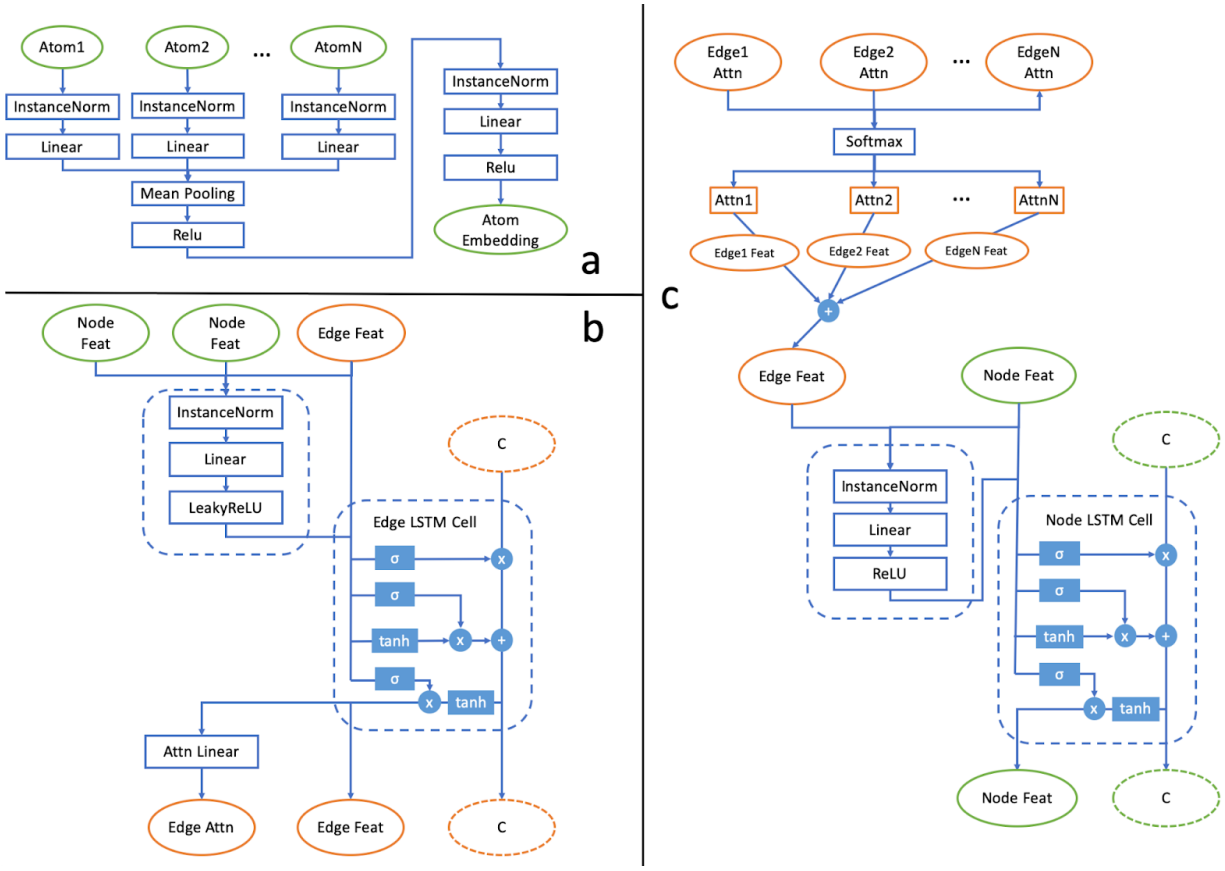
Supplementary Figure 5. The decoy quality distribution of the training data.

11. Summary of Input Features

Supplementary Table 23. Summary of input features

Type	Feature	Dimension
Residue	One-hot encoding of residue	21
	rPosition	1
	Dihedral, SS3 and RSA calculated by DSSP	6
	One-hot encoding and relative coordinate of heavy atoms	7
Residue pair	Distance (CαCα, $C_\beta C_\beta$ and NO)	3
	Orientation (ω, θ and φ)	3
	Sequential separation	9

12. Detailed architectures of the atom embedding module and message passing blocks for edges and nodes



Supplementary Figure. 6. Detailed architectures of the atom embedding module and message passing blocks for edges and nodes. A. The atom embedding module; B. The message block for edges; C. The reduce block for nodes

References

1. Zhang, Y. & Skolnick, J. SPICKER: A clustering approach to identify near-native protein folds. *Journal of Computational Chemistry* **25**, 865–871 (2004).
2. Hiranuma, N. *et al.* Improved protein structure refinement guided by deep learning based accuracy estimation. *Nature Communications* **12**, 1340 (2021).
3. Igashov, I., Olechnovič, liment, Kadukova, M., Venclovas, Č. & Grudinin, S. VoroCNN: Deep convolutional neural network built on 3D Voronoi tessellation of protein structures. *Bioinformatics* (2021) doi:10.1093/bioinformatics/btab118.
4. Zhang, J. & Zhang, Y. A Novel Side-Chain Orientation Dependent Potential Derived from Random-Walk Reference State for Protein Fold Selection and Structure Prediction. *PLOS ONE* **5**, e15386 (2010).
5. Won, J., Baek, M., Monastyrskyy, B., Kryshtafovych, A. & Seok, C. Assessment of protein model structure accuracy estimation in CASP13: Challenges in the era of deep learning. *Proteins: Structure, Function, and Bioinformatics* **87**, 1351–1360 (2019).
6. Haas, J. *et al.* The Protein Model Portal--a comprehensive resource for protein structure and model information. *Database (Oxford)* **2013**, bat031 (2013).
7. Kinch, L. N., Kryshtafovych, A., Monastyrskyy, B. & Grishin, N. V. CASP13 target classification into tertiary structure prediction categories. *Proteins: Structure, Function, and Bioinformatics* **87**, 1021–1036 (2019).
8. Xu, J. Distance-based protein folding powered by deep learning. *PNAS* **116**, 16856–16865 (2019).