Bioinformatics, 37(21), 2021, 3752–3759
doi: 10.1093/bioinformatics/btab632
Advance Access Publication Date: 2 September 2021
Original Paper

Structural bioinformatics

Improved estimation of model quality using predicted inter-residue distance

Lisha Ye[1], Peikun Wu[1], Zhenling Peng [2], Jianzhao Gao [1], Jian Liu [3] and Jianyi Yang [1,*]

[1]School of Mathematical Sciences, Nankai University, Tianjin 300071, China, [2]Research Center for Mathematics and Interdisciplinary Sciences, Shandong University, Qingdao 266237, China and [3]College of Computer Science, Nankai University, Tianjin 300071, China

*To whom correspondence should be addressed.

Associate Editor: Jan Gorodkin

Received on February 27, 2021; revised on August 27, 2021; editorial decision on August 30, 2021; accepted on August 31, 2021

Abstract

Motivation: Protein model quality assessment (QA) is an essential component in protein structure prediction, which aims to estimate the quality of a structure model and/or select the most accurate model out from a pool of structure models, without knowing the native structure. QA remains a challenging task in protein structure prediction.

Results: Based on the inter-residue distance predicted by the recent deep learning-based structure prediction algorithm trRosetta, we developed QDistance, a new approach to the estimation of both global and local qualities. QDistance works for both single- and multi-models inputs. We designed several distance-based features to assess the agreement between the predicted and model-derived inter-residue distances. Together with a few widely used features, they are fed into a simple yet powerful linear regression model to infer the global QA scores. The local QA scores for each structure model are predicted based on a comparative analysis with a set of selected reference models. For multi-models input, the reference models are selected from the input based on the predicted global QA scores. For single-model input, the reference models are predicted by trRosetta. With the informative distance-based features, QDistance can predict the global quality with satisfactory accuracy. Benchmark tests on the CASP13 and the CAMEO structure models suggested that QDistance was competitive with other methods. Blind tests in the CASP14 experiments showed that QDistance was robust and ranked among the top predictors. Especially, QDistance was the top 3 local QA method and made the most accurate local QA prediction for unreliable local region. Analysis showed that this superior performance can be attributed to the inclusion of the predicted inter-residue distance.

Availability and implementation: http://yanglab.nankai.edu.cn/QDistance.

Contact: yangjy@nankai.edu.cn

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Quality assessment (QA) of protein structure models is an essential component in protein structure prediction (Uziela *et al.*, 2017; Won *et al.*, 2019) and refinement (Hiranuma *et al.*, 2021). Due to its importance, QA has been one of the prediction categories in the Critical Assessment of Techniques for Protein Structure Prediction (CASP) since 2006 (Cozzetto *et al.*, 2007, 2009; Kryshtafovych *et al.*, 2011, 2014, 2016, 2018). There are two kinds of QA: global QA and local QA. Global QA aims to estimate the quality of a structure model or select the most accurate model out from a pool of structure models. Local QA is to predict the residue-specific distance deviation in a structure model. Significant progress has been made in the field of protein structure prediction, mostly due to the rapid development of deep learning, such as in RaptorX-Contact (Xu, 2019), AlphaFold1(Senior *et al.*, 2020), trRosetta (Yang *et al.*, 2020) and AlphaFold2 (Callaway, 2020). In contrast, though many QA methods have been developed, the progress is less significant and more efforts are required (Won *et al.*, 2019).

Current QA methods can be divided into two groups depending on the input number of models: single-model-based or clustering-based methods. The single-model-based methods require only one model as input, such as the ProQ series (Uziela *et al.*, 2016, 2017; Uziela and Wallner, 2016; Wallner and Elofsson, 2003), VoroMQA (Olechnovic and Venclovas, 2017), QAcon (Cao *et al.*, 2017), SVMQA (Manavalan and Lee, 2017), Qprob (Cao and Cheng, 2016), DeepQA (Cao *et al.*, 2016), CNNQA (Hou, 2019), Ornate (Pages *et al.*, 2019), GraphQA (Baldassarre *et al.*, 2021) and so on.

The main differences between these methods are in two folds: different feature representations (e.g. potential-based features and model geometry-based features) and prediction engines (e.g. support-vector machines and deep learning). In addition, there are also a few quasi single-model methods that support single-model input, but using reference models created by other tools (Maghrabi and McGuffin, 2017). Clustering-based methods take a set of models as input, such as Pcons (Lundstrom *et al.*, 2001), APOLLO (Wang *et al.*, 2011), ModFOLDclust (McGuffin, 2008) and ResQ (Yang *et al.*, 2016), etc. In general, these methods require that there should be a subset of models with apparent pairwise similarity, which largely limits their application in reality. By contrast, the single-model-based methods do not have such limitation but their accuracy is relatively low in general.

In recent years, *de novo* protein structure prediction becomes much more accurate than before, mostly due to the accurate inter-residue contact/distance prediction by deep learning (Greener *et al.*, 2019; Senior *et al.*, 2020; Xu, 2019; Yang *et al.*, 2020). It is thus of great interest to test if the predicted inter-residue contact/distance can be used to improve QA. In fact, a few papers were published in the course of this work. The first was QDeep (Shuvo *et al.*, 2020), which performed dynamic programing to calculate the alignment score of the predicted and the model's distance histogram as a part of the network's input features. The second was ResNetQA (Jing and Xu, 2020), in which model-derived distance map and predicted distance potential were fed into a 2D residual neural network to make global and local QA prediction simultaneously. Recently, predicted local quality score (plDDT) is provided along with the predicted structure model by the amazing system AlphaFold2 (Jumper *et al.*, 2021).

In this work, we introduce QDistance, a new global and local QA prediction method using predicted inter-residue distance. Benchmark tests on the CASP13 and the CAMEO datasets suggested that QDistance had competitive accuracy with other methods. Blind tests in the CASP14 experiments indicated that QDistance was one of the top performers.

2 Materials and methods

2.1 Overview of the proposed method

The flowchart of QDistance is shown in Figure 1. The input to QDistance can be either single model (Fig. 1A) or a set of models (Fig. 1B) for a target protein. For a target, a multiple sequence alignment (MSA) is generated by searching the query sequence against the sequence database Uniclust30 (Version 2018_08) by the software HHblits (Version 3.0.3) with an *e*-value cutoff 0.001. trRosetta is used to predict inter-residue distance and structure models for a query target. To predict the global QA score for each model, three groups of features are designed, including features based on: 2D distance matrix comparison, potential scores and other single QA methods and 1D structural feature comparison (Fig. 1C). These features are fed into a simple yet powerful linear regression model to predict the GDT_TS score (Zemla, 2003). For multi-models input, an additional step of comparative analysis [i.e. Equation (5)] with the top-scoring models (ranked based on the linear regression score) is performed to improve the global QA prediction. To make local QA prediction for multi-models input, the top models (according to the predicted GDT_TS score) are first collected. A consensus analysis is then used to infer the local QA score for each model [with Equation (6)]. We can see that the multi-models local QA prediction procedure can be applied to inputs with multiple models only. For single-model input, to make local QA prediction with a similar idea, we apply trRosetta to generate five reference models.

2.2 Feature design

2.2.1 Features based on 2D distance matrix comparison

The inter-residue distance prediction becomes more and more accurate due to the rapid development and application of deep learning algorithms. In this work, the distance matrix for a target protein is predicted by trRosetta (Yang *et al.*, 2020), one of the state-of-the-art methods for protein structure prediction. The distance predicted by trRosetta is a histogram represented by a 3D matrix of size $L \times L \times 37$, where L is the length of the target protein and 37 represents the 37 distance bins. Thus, the distance for each residue pair is represented by a probability distribution. Contact- and distance-based features are designed from this distribution detailed below.

Three ways are used to define a contact matrix [denote by CT = (ct_{ij})] for the target protein, starting from the prediction by trRosetta. The first is based on the summed probability of the distance bins between 0 and 8 Å. A pair of residues are defined to be in contact if the probability is higher than 0.5. The second way is defined based on probability weighted sum of the distances in all bins, which results into a distance matrix [denote by DT = (dt_{ij})]. A pair of residues is defined to be in contact if the distance is <8 Å. The third way is similar to the second one but with a different distance matrix; i.e. the distance corresponds to the bin with the maximum probability.

For each structure model, based on the coordinates of its C-β (C-α for glycine) atoms, a distance matrix can be obtained [denote by DM = (dm_{ij})]. Two residues are in contact if the distance dm_{ij} is <8 Å, resulting into a contact matrix [denote by CM = (cm_{ij})].

For each definition of *CT*, two contact-based features are defined as the number of common contact pairs in CT and CM, divided by the total number of contact pairs in CT and CM, respectively. Thus, a total of six contact-based features are obtained.

For each of the two predicted distance matrices (i.e. from weighted sum and maximum probability), four distance-based features are defined to measure the agreement between the predicted and model-derived distance matrices. The first is defined by the following equation.

$$S_1 = \frac{1}{4L} \sum_{t \in \{1,2,4,8\}} \sum_{i=1}^{L} \frac{\sum_{j,|i-j| \geq 12, dt_{ij} \leq 20} I(|dt_{ij} - dm_{ij}| \leq t)}{\sum_{j,|i-j| \geq 12, dt_{ij} \leq 20} 1}, \quad (1)$$

where $I()$ is an indicator function which equals to 1 if the condition is satisfied and 0 otherwise. The remaining three are defined as follows:

$$S_2 = 1 - \frac{|\sum dt_{ij} \cdot dm_{ij}|}{\sqrt{(\sum dt_{ij}^2)}\sqrt{(\sum dm_{ij}^2)}}, \quad (2)$$

$$S_3 = 1 - \frac{\sum_{i,j}(dt_{ij} - dm_{ij})^2}{\max(\sum_{i,j} dt_{ij}^2, \sum_{i,j} dm_{ij}^2)}, \quad (3)$$

$$S_4 = 1 - \frac{\sqrt{\sum_{i,j}(dt_{ij} - dm_{ij})^2}}{\max(\sqrt{\sum_{i,j} dt_{ij}^2}, \sqrt{\sum_{i,j} dm_{ij}^2})}. \quad (4)$$

In summary, for each structure model, 14 features can be obtained from the above calculations (6 from contact and 8 from distance comparisons).

2.2.2 Features based on potentials scores and other single QA methods

A total of 15 features are designed, including 12 potential scores and predictions from 3 single QA methods. The potential scores include Dope (Shen and Sali, 2006), OPUS (Wu *et al.*, 2007), RWplus (Zhang and Zhang, 2010), RF_CB_SRS_OD (Rykunov and Fiser, 2007), DFIRE2 (Zhou and Zhou, 2002) and 7 Rosetta's energy terms used in ProQ3 (Uziela *et al.*, 2016): radius of gyration (rg), statistical potentials for secondary structure information (hs_pair, ss_pair, sheet, rsigma), contact order (co) and centroid hydrogen bonding (cen_hb). Similar to the procedure adopted in ProQ3, side-chain rebuilding and energy minimization were performed before evaluating the Rosetta energy on each structure model. The single QA methods include DeepQA (Cao *et al.*, 2016), ModelEvaluator

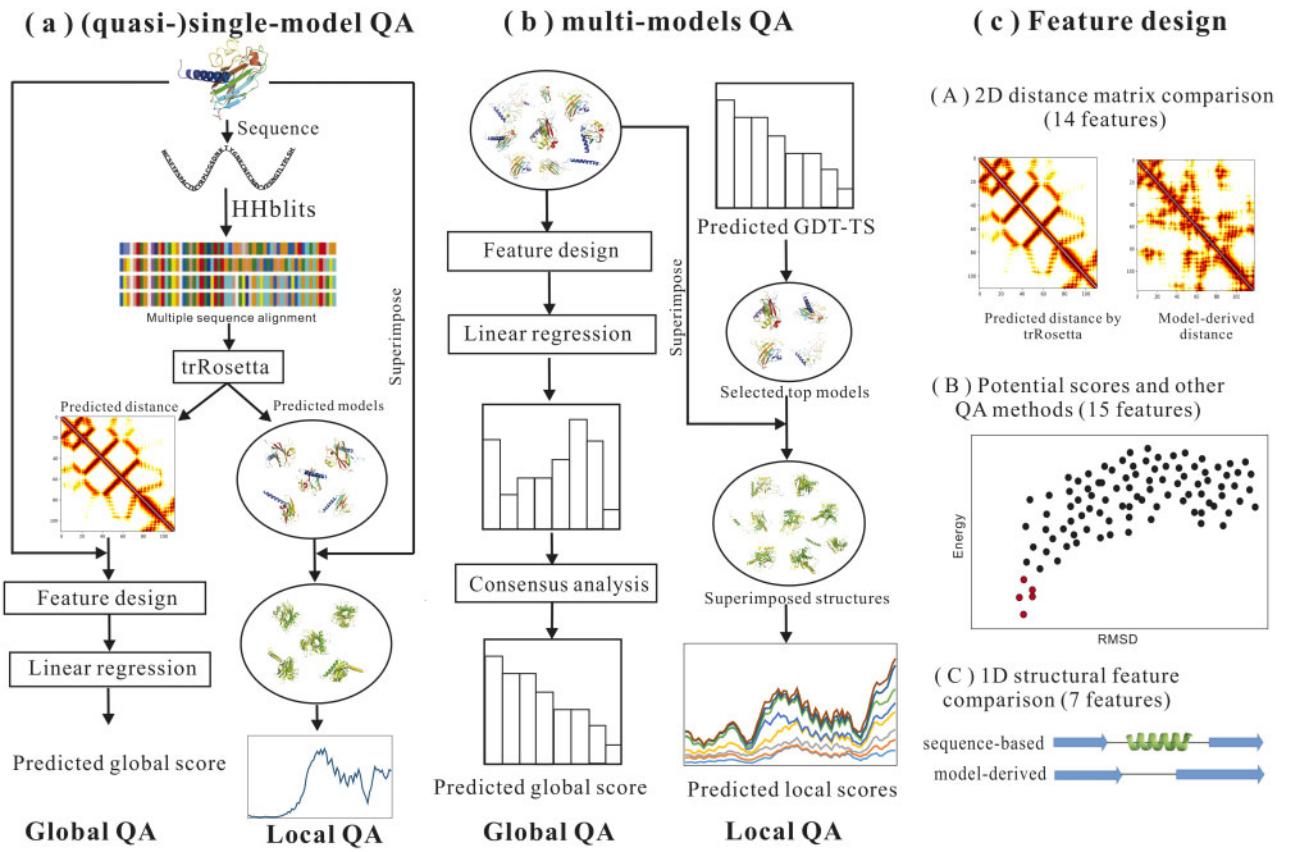

Fig. 1. Flowchart of the proposed method QDistance. The supported input includes single model and multiple models. (**A**) Single-model QA prediction procedure. A linear regression model is used for global QA prediction. And for local QA prediction, we used the structure models generated by trRosetta as the reference models to infer the local quality scores using Equation (7). (**B**) Multi-models QA prediction procedure. For global QA, an additional refinement step based on consensus analysis [i.e. Equation (5)] is performed after linear regression. Local QA prediction for each structure model is based on consensus analysis through comparison [i.e. Equation (6)] with a set of reference models, selected based on the global QA scores. (**C**) Three groups of features are designed to encode each structure model

(Wang *et al.*, 2009) and Qprob (Cao and Cheng, 2016). All scores are converted into the range of [0, 1] using the logistic function.

2.2.3 Features based on 1D structural feature comparison

Similar to other QA methods, we generate seven features based on 1D structural feature comparison as in the DeepQA (Cao *et al.*, 2016). The considered 1D structural features include surface area, secondary structure, exposed mass, exposed surface, solvent accessibility and Euclidean compact. The hypothesis is that the 1D structural features derived from a good structure model should match well with sequence-based 1D structural feature prediction, given that the sequence-based prediction is very accurate nowadays. More details about these features are available in the Supplementary Table S1.

2.3 Global and local QA prediction

To predict the global QA score for each model, the features generated above are fed into a linear regression model. The linear regression scores for the input with multiple models are then refined by a comparative analysis with the following equation. For single-model input, no refinement is performed.

$$G_i = \frac{\sum_{j=1}^{N_1} GDT - TS_{ji} \times L_j}{\sum_{j=1}^{N_1} L_j}, i = 1, \ldots, T, \quad (5)$$

where G_i is the predicted GDT_TS score for the ith model, T is the total number of input models, L_j is the linear regression score for the jth reference model, GDT_TS_{ji} is the GDT_TS score between the ith and the jth reference model and N_1 is the number of top models ranked based on the linear regression score. In this work, it is set to 20% of the total number of input models.

The local QA prediction for input with multiple models is based on a similar idea of comparative analysis. The top 10% models (ranked by the predicted GDT_TS score) are selected as reference models. The local scores for each model are then predicted based on comparison with the reference models.

$$d_{ik} = \frac{\sum_{j=1,j\neq i}^{N_2} d_{ijk} \times G_j}{\sum_{j=1,j\neq i}^{N_2} G_j}, i = 1, \ldots, T; k = 1, \ldots, L, \quad (6)$$

where d_{ik} is the predicted distance deviation for the kth residue in the ith model, d_{ijk} is the distance deviation between the ith model and the jth reference model for the kth residue after superimposition, G_j is the same as in Equation (5), N_2 is the number of top models ranked based on G_j and L is length of the target.

To predict the local QA scores for single-model input, we use the models generated by trRosetta as reference models. The local QA scores are then predicted with the following formula.

$$d_k = \frac{1}{5}\sum_{j=1}^{5} d_{jk}, k = 1, \ldots, L, \quad (7)$$

where d_k is the predicted distance deviation for the kth residue in the input model, d_{jk} is the distance deviation between the input model and the jth reference model for the kth residue after superimposition.

2.4 Performance evaluation

In CASP, a few major metrics are used to evaluate the global QA prediction: Loss, Pearson's correlation coefficient (PCC) (denote by Pearson) and Best_difference (denote by Diff).

$$\text{Loss} = 100 \times \sum_{i=1}^{N} |x_i - y_i|, \tag{8}$$

$$\text{Diff} = 100 \times |y_{pred} - y_0|, \tag{9}$$

where x_i *and* y_i are the predicted and real GDT_TS score of the ith model, respectively; y_{pred} is the real GDT_TS score of the model with the highest predicted score, y_0 is the real GDT_TS score of the best model. From the definition, we can see that lower values of the Loss and Best_difference, and higher value of Pearson reflect more accurate global QA prediction. Note that the global quality GDT_TS is superimposition-dependent and thus domain orientations may affect the evaluation results. In contrast, the lDDT score is superimposition-independent and thus it is also considered in our evaluation. By default, GDT_TS is adopted unless declared in the remaining of this article.

For local QA evaluation, two metrics are used to evaluate the performance. The first is ASE:

$$ASE = 100 \times (1 - \frac{1}{L}\sum_{i=1}^{L} |S(e_i) - S(r_i)|), \tag{10}$$

where L is the length of the target; e_i and r_i are the estimated and real distance deviation of the model's ith residue, respectively; $S(d) = 1/(1+(d/d_0)^2)$ is a S-function; the value of d_0 is set to 5 Å in the CASP official evaluation (Won *et al.*, 2019). However, this cutoff might be too high. The average ASE over four lower distance thresholds of 0.5, 1, 2 and 4 Å is calculated as well, inspired by the cutoffs used in GDT_TS (Zemla, 2003). The average ASE over all models for all targets in a dataset is calculated for comparison. The closer ASE is to 100, the more accurate the local quality prediction is. In the remaining of this article, d_0 is set to 5 Å for ASE by default unless declared. The second measure is the PCC between the predicted local scores and the ground truth.

3 Results and discussion

We trained our method on the protein structure models from CASP8–12 and tested it on the structure models from CASP13 and CAMEO (3 months between June 19, 2020 and September 12, 2020). The models were downloaded from the respective official websites. Besides these benchmark tests, we also participated in the blind tests in the QA category of the CASP14 experiment.

3.1 Contribution of different features

As shown in Figure 1C, the features used by QDistance are divided into three groups: A, B and C. Table 1 lists the performance of QDistance with individual feature groups and combinations to investigate their contributions. In order to check the contribution directly, no refinement was performed [i.e. with Equation (5)]. The global QA scores were from linear regression directly. Among the three feature groups, distance-based features (A) lead to the most accurate prediction for all three metrics. The combined feature group (i.e. B + C) outperforms each individual feature group (B) and (C) in terms of Loss and Best_difference but with lower PCC. Additional inclusion of the distance-based features significantly improves this measure from 0.585 to 0.763. The ability in selecting the best model and reproducing GDT_TS is also enhanced. For example, the Best_difference improves from 8.36 to 6.322, and the Loss improves from 11.839 to 9.13.

In addition, the contribution of the features is quantitatively measured by the average weight of the linear regression coefficients. Supplementary Figure S1A suggests that distance-based features contribute the most to the prediction. The distance-based features are further divided into five sub-groups: contact-based features (6) and S1–S4 [i.e. Equations (1)–(4)]. Supplementary Figure S1B suggests that the contact-based features have lower contribution than other features. This is probably because the distance contains much richer information than the binary contact.

Table 1. The performance of QDistance on the 73 targets of CASP13's dataset with different features

Feature	Diff	Loss	Pearson
A	8.362	11.457	0.729
B	11.27	12.008	0.612
C	11.615	12.805	0.588
B + C	8.36	11.839	0.585
A + B+C	6.322	9.13	0.763

Note: The feature groups A–C are the same as those in Figure 1. A: distance-based features, B: potential scores and single QA methods and C: features based on 1D structural feature comparison. Note that the global QA scores were from linear regression directly without refinement in Equation (5) to quantify the contribution of different features directly.

3.2 Performance on the CASP13 dataset

There are two stages in the CASP13 QA experiments. The main purpose of Stage 1 is to decide if a QA predictor is a single- or multi-models method. Thus, we only present the results on the Stage 2 models, which are summarized in Table 2.

Table 2 presents the results of our method and the top-performing QA groups in CASP13 (Won *et al.*, 2019). Our single-model-based method is more accurate than the single-model methods ModFoLD7_rank, ProQ3D and FaeNNz, with lower Diff and Loss and higher Pearson. However, it is less accurate than the top-performing groups MULTICOM_CLUSTER, UOSHAN and ModFOLDclust2, which are multi-models-based methods. After the refinement based on comparisons with the reference models using Equation (5), a significant improvement was achieved in QDistance, which outperforms all other methods. For example, QDistance's Best_difference and Loss are 5.324 and 4.875, respectively, which are all lower than MULTICOM_CLUSTER (5.406 and 7.676). The PCC by multi-models-based QDistance is 0.906, also higher than the best method UOSHAN (0.895).

When the ground truth is replaced by the lDDT score (Mariani *et al.*, 2013), our method has similar accuracy with UOSHAN but is less accurate than MULTICOM_CLUSTER (Supplementary Table S2). This may be because our method was trained to reproduce the GDT_TS score rather than the lDDT score.

Table 2. Comparison with other top global QA methods on models from CASP13

Method	Diff	Loss	Pearson
QDistance[b]	5.324	4.875	0.906
MULTICOM_CLUSTER	5.406	7.676	0.86
UOSHAN	5.523	5.5	0.895
ModFOLDclust2	6.688	7.414	0.852
QDistance[a]	6.322	9.13	0.763
ModFoLD7_rank	6.612	11.883	0.761
ProQ3D	8.306	10.632	0.65
FaeNNz	8.773	11.277	0.698

Note: The global accuracy is measured based on GDT_TS. The multi-models-based prediction by QDistance was based on the refinement with Equation (5). Evaluation on the common submission results of the above methods, including 73 CASP13 targets.

[a]Single model-based (the same as A + B + C in Table 1).

[b]Multi-models-based.

For input with multiple models, the top models (ranked based on the predicted global QA scores) are selected as reference models to predict the local QA scores for each input model. A detailed relation between ASE and the number of used reference models is available in Supplementary Figure S2, which suggests that 15 seems to be an optimal number. The local QA for each model is then obtained based on pairwise comparisons with the reference models with Equation (6). For input with a single model, the predicted structure models by trRosetta are used as reference models to infer the local QA scores for the input model using Equation (7).

The PCC and ASE of the predicted local quality on the CASP13 models by QDistance and other top-performing groups in CASP13 are shown in Figure 2. Both versions of QDistance outperform other groups according to PCC (Fig. 2A). The multi-models-based version of QDistance achieves a PCC above 0.7, higher than the quasi single-model-based version of QDistance. When it comes to ASE, the results depend on the value of d_0. When it is set to 5 Å as done in CASP, the multi-models-based QDistance has the highest ASE (87.883) (Fig. 2B); while the quasi single-model-based QDistance has comparable ASE with the method UOSHAN. When lower cutoff values of d_0 are used, the ASEs for all methods increased (see Supplementary Table S3). For example, the ASE for QDistance (with multiple models) increases from 87.646 to 92.116 when the cutoff decreases from 4 to 0.5 Å. The control method UOSHAN seems to be more sensitive at a lower cutoff, which achieves the highest ASE (94.134) when d_0 is 0.5 Å. This is probably because this method sets a maximum cutoff of 15 Å for predicted distance deviation. Averaging the ASEs over the four cutoffs leads to the highest ASE by UOSHAN, followed by QDistance (Fig. 2C). Note that when d_0 is 0.5 or 1 Å, all other methods do not have distinguishable ASE values, suggesting that these cutoffs may be too stringent to be used for local QA assessment.

Since our multi-models-based local QA relies on a set of reference models, we investigate on the performance difference on the reference models and other models (see Supplementary Fig. S3). As expected, the average GDT_TS score of the selected reference models is higher than the remaining models (0.542 versus 0.39, Supplementary Fig. S3A). When measured by PCC and ASE ($d0 = 5$ Å), there is no notable difference between the reference models and other models (Supplementary Fig. S3B and C). However, when measured by the average ASE at four distance cutoffs ($d0 = 0.5$, 1, 2 and 4 Å), the reference models show less accurate ASE than other models (84.781 versus 89.519, Supplementary Fig. S3D). This might be because the reference models are similar to each other, making the predicted local distance deviation artificially small. The *S*-function difference for the predicted and the real local distance deviation may be further enlarged at a more stringent cutoff (i.e. $d0 = 0.5$ or 1 Å), making the ASE values become smaller for the reference models.

3.3 Performance on the CAMEO dataset

To further validate the performance of QDistance (quasi single-model-based), we collected ~2000 models from CAMEO (12 weeks between June 19, 2020 and September 12, 2020). CAMEO is a weekly assessment of protein structure prediction methods (Haas *et al.*, 2018), in which only local QA is evaluated. Given that the participating groups in CAMEO only have access to single model during the prediction period, we present the results for the quasi single-model-based QDistance. The raw structure models and the QA predictions by other methods were downloaded directly from the CAMEO official website. There are 182 targets for this dataset and about 10 models on average for each target. QDistance's local prediction is similar to the previously described procedure, but with the pairwise local score being lDDT (Mariani *et al.*, 2013) rather than distance deviation.

There are a few differences between the CAMEO and the CASP's local QA assessment. First, the predicted score in CAMEO is the local lDDT rather than distance deviation. Second, the evaluation in CAMEO is based on AUC rather than the ASE. Three different ways can be used to calculate the AUC scores: residue-based (AUC_r), model-based (AUC_m) and target-based (AUC_t). In the residue-based calculation, a single AUC is calculated by comparing the predicted lDDT scores for all residues from all models and targets with the native ones. While in the model-based version, an AUC is calculated for each model and the average AUC over all models is used as the final AUC. For the target-based evaluation, the AUC for all residues from all models of the same target is first computed and the average AUC over all targets is calculated.

Figure 3 presents the comparison between QDistance and other QA methods on the CAMEO dataset. The official evaluation in CAMEO is based on AUC_r (Fig. 3A). QDistance's AUC_r is 0.883, lower than the top method QMEANDisco 3 (0.937) but with similar to other methods. Note that the AUC_r may not reflect the exact performance of a QA method because the residues from all models and targets are treated equally, though the length and difficulty of different targets are usually different. To partly address this issue, the other two metrics, AUC_m and AUC_t are introduced above, which are presented in Figure 3B and C, respectively. The absolute values of AUC_m and AUC_t become lower than AUC_r for almost all methods, suggesting that the evaluation based on AUC_r may overestimate the performance of QA methods. The AUC_m and AUC_t values for our method are 0.834 and 0.887, respectively, both higher than the method QMEANDisco 3 (0.833 and 0.884).

Similar to AUC, we also calculate three forms of PCCs: PCC_r, PCC_m and PCC_t. Supplementary Table S4 shows that our method is less accurate than QMEANDisco 3, probably because our method has been trained to predict local distance deviation rather than local lDDT. Nevertheless, QDistance outperforms or is competitive with other methods.

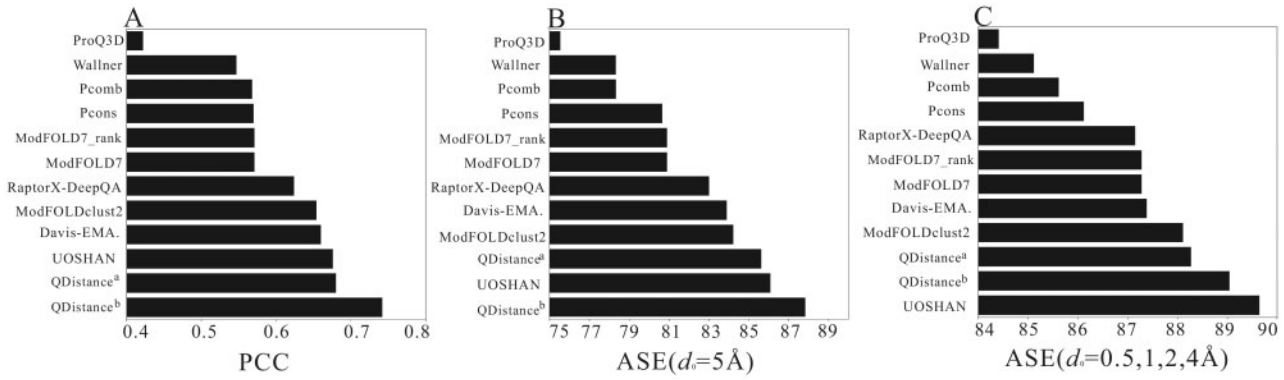

Fig. 2. Comparison of QDistance with other local QA methods on the models for 72 CASP13 targets. (a) Quasi single-model-based; (b) multi-models-based. (**A**) PCC. (**B**) ASE at $d0 = 5$ Å. (**C**) Average ASE at $d0 = 0.5$, 1, 2, 4 Å

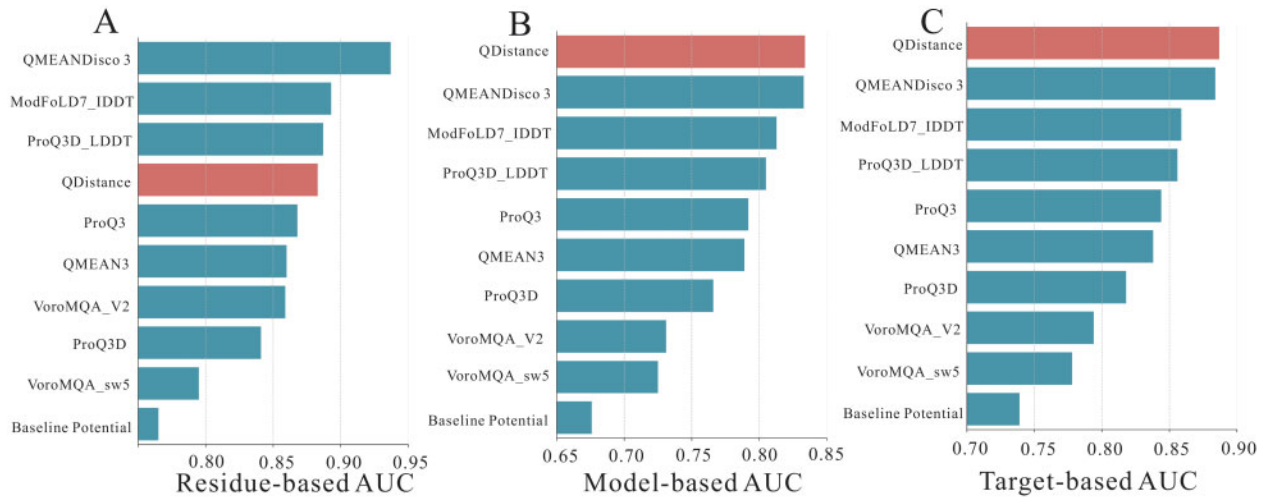

Fig. 3. Comparison of QDistance with other methods on the CAMEO dataset. The dataset consists of 158 targets that all assessed methods have prediction results. (**A–C**) are the residue-based, model-based and target-based AUCs, respectively

3.4 Blind tests in the CASP14 experiment

We participated in the QA category in the bind tests of the CASP14 experiment with two groups Yang_TBM and Yang-Server. Both groups were built based on the QDistance algorithm but with different ways of selecting the top reference models [i.e. in Equation (5)]. The former is based on predicted global QA score (the same as QDistance) while the latter is based on the average pairwise TM-score (Zhang and Skolnick, 2004) to other models.

Figure 4 shows the results of the global and local QA prediction for ~10 000 models (Stage 2) of 68 CASP14 targets. The Best_difference, Loss and ASE were directly from the official website while the PCC for each group was the average of the individual coefficients over all 68 targets. A total of 72 groups participated in the global QA prediction but four groups were excluded as they submitted predictions for only a few targets. Figure 4(A–C) shows the Loss, Best_difference and Pearson, respectively. For global QA prediction, our group, Yang_TBM (highlighted as filled bars), ranks at the 7th, 12th and 10th according to Loss (Fig. 4A), Best_difference (Fig. 4B) and Pearson (Fig. 4C), respectively. In fact, it is difficult to maintain a top position for all metrics. For example, the group MULTICOM-CONSTRUCT is the top predictor according to Best_difference; however, it is ranked at the 11th and 12th by Pearson and Loss, respectively.

There are 39 groups participating in the local QA prediction. Figure 4D shows that Yang_TBM ranks at the third with very similar ASE (~85) to the top two groups (DAVIS-EMAconsensus and ModFOLDclust2). In addition, according to the official assessment in the CASP14 meeting, Yang_TBM's local QA prediction had the highest accuracy for unreliable local regions, which are defined as regions of sequential residues with distance deviation >3.8 Å. The top method in CAMEO, QMEANDisCo, also participated in the test. Table 3 shows the rankings of our method and other selected methods of interest. It shows that QMEANDisCo were ranked lower compared with our method. This might be because different metrics are used in CAMEO and CASP for evaluating the performance.

As mentioned in Section 1, two recently published QA methods, QDeep and ResNetQA, also made use of predicted distance. According to the description in the CASP14 abstracts, these two methods also participated in the CASP14 experiments with group names Bhattacharya-QDeep and RaptorX-QA, respectively. The rankings of these two groups and QDistance-based Yang_TBM are summarized in Table 3. Though predicted distance is employed in all three methods, the table shows that Yang_TBM has much higher ranking than these two methods. This difference may be attributed to three aspects. The first is different methods are used to predict the distance: DMPfold in Bhattacharya-QDeep, RaptorX-Contact in RaptorX-QA and trRosetta in Yang_TBM. The second is different ways of using predicted distance. In Yang_TBM, a set of 14 features are carefully designed to reflect difference between the model-derived distance and the predicted distance. For Bhattacharya-QDeep, it used a few distance map similarity scores based on dynamic programing alignments as input features. And for RaptorX-QA, deep residual neural networks were applied with direct input of distance matrices. The third is these two methods do not make use of reference models.

In addition, the QA prediction by the group BAKER-ROSETTASERVER in CASP14 was based on the method DeepAccNet-MSA, which made use of the trRosetta output as input to its network (Hiranuma *et al.*, 2021). For global QA, Table 3 shows that BAKER-ROSETTASERVER's ranking is slightly higher than Yang_TBM based on Diff (7 versus 12), but lower than Yang_TBM based on the metrics Loss and Pearson. For local QA, BAKER-ROSETTASERVER has a lower ranking than Yang_TBM probably because it is a single-model-based method while Yang_TBM is based on multiple models.

3.5 Comparison with other methods with the same MSA

As Bhattacharya-QDeep (QDeep in brief), ResNetQA (used by RaptorX-QA) and our method rely on MSA, it is necessary to compare their performance with an identical set of input MSAs. We try to run both methods locally with the same MSAs used by QDistance. QDeep was downloaded and ran locally with default options. ResNetQA was also ran locally but its input was obtained by submitting the MSAs to the RaptorX-Contact server. As both QDeep and ResNetQA are single-model-based method, we ran QDistance based on single model (or quasi-single model for local QA) for a fair comparison.

Table 4 shows that when the same set of MSAs is used, QDistance outperforms both methods, for both global and local QA predictions. We noticed that both methods provided their raw predictions for 20 CASP13 targets at Github, which are also included in Table 4. It suggests that the downloaded predictions are more accurate than those obtained by running the respective methods locally. It might be because different MSAs were used. For ResNetQA, another possible reason is the RaptorX-Contact server was not updated and incompatible with ResNetQA (private communication).

3.6 Strength and weaknesses of QDistance

QDistance has been shown to be competitive with other QA methods in the above benchmark and blind tests. QDistance relies on the comparisons with a set of selected reference models. Thus, it performs well when multiple accurate models are successfully identified from the assessed set of models. One of the known weaknesses of

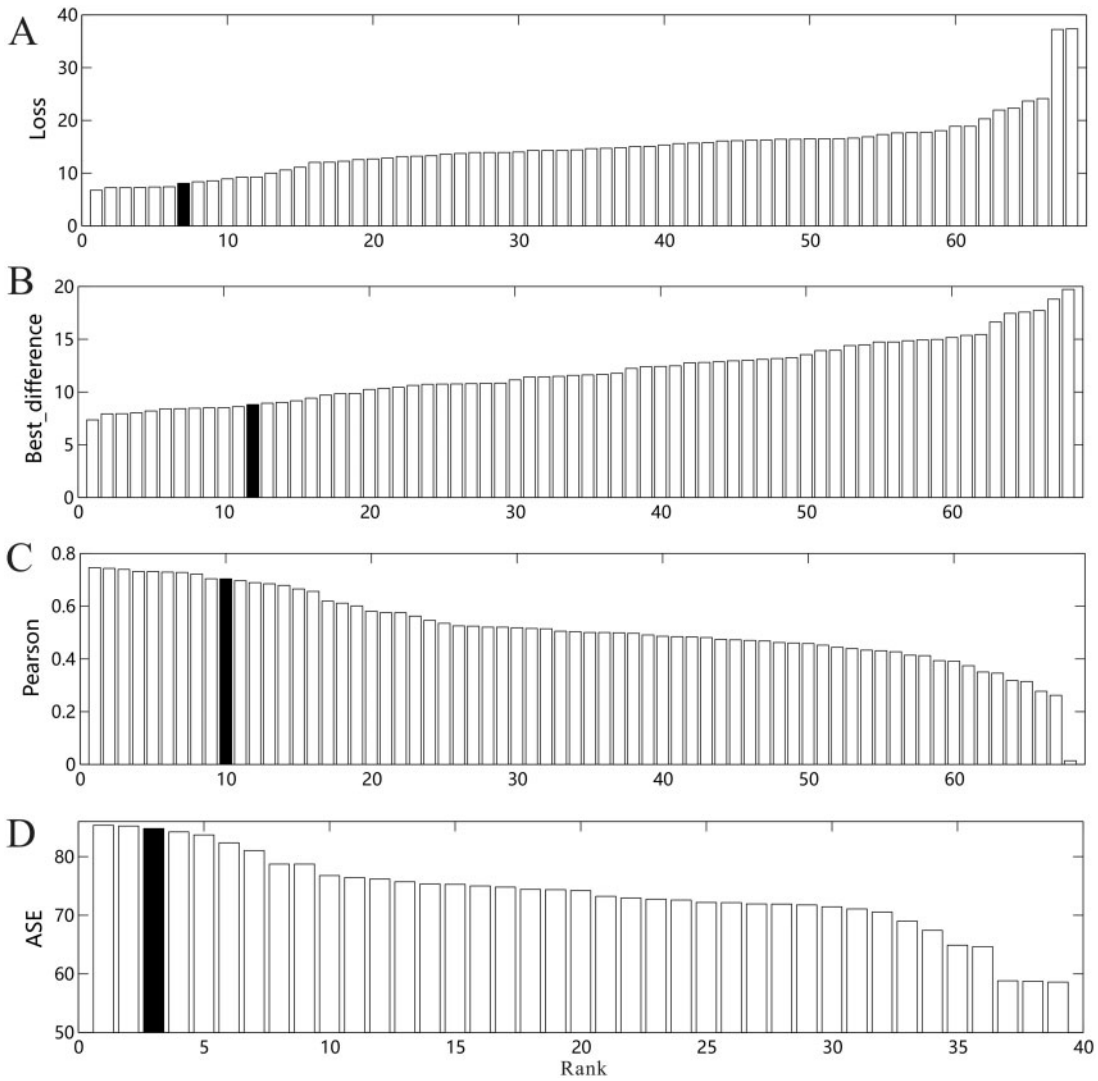

Fig. 4. The performance of QDistance-based method Yang_TBM in CASP14's blind test. The *x*-axis is the ranking of the participating groups in CASP14. (**A–C**) are for global QA prediction and (**D**) is for local QA prediction. Yang_TBM is highlighted as black bar while other groups are shown as white bars

Table 3. Ranking of the QDistance-based Yang_TBM and other selected methods of interest

Group	Loss	Diff	Pearson	ASE
Yang_TBM	7	12	10	3
BAKER-ROSETTASERVER	20	7	18	11
Bhattacharya-QDeep	39	41	53	16
RaptorX-QA	17	62	51	30
QMEANDisCo	28	49	42	39

Note: Bhattacharya-QDeep and RaptorX-QA used predicted distance. QMEANDisCo was the top method according to the CAMEO experiment. BAKER-ROSETTASERVER used trRosetta outputs as input features.

Table 4. Comparison of QDistance with QDeep and ResNetQA based on an identical set of MSAs

Method	Diff	Loss	Pearson	PCC	ASE
QDistance	4.438	8.865	0.793	0.661	85.328
QDeep[a]	10.334	15.284	0.563	N/A	N/A
QDeep[b]	8.88	10.123	0.751	N/A	N/A
ResNetQA[a]	9.521	14.569	0.667	0.384	66.646
ResNetQA[b]	8.57	7.89	0.823	0.54	84.5

Note: The data are on 20 CASP13 targets that were used by both methods.
[a]The raw predictions were obtained by running the methods locally.
[b]The raw predictions were downloaded from Github.

the consensus-based method is that it cannot identify an outstanding but minority model. This issue has been partly addressed by introducing the new distance-based features. When the predicted distance is accurate, such model is possible to be recognized with the newly designed distance features [Equations (1)–(4)]. However, QDistance may not work well when the predicted distance is not accurate. Because the accuracy of the predicted distance depends on the availability of enough homologous sequences in the MSA, we check the relationship between the MSA depth and the QA accuracy. Supplementary Table S5 shows that QDistance is on average more accurate when the MSA is deeper.

4 Conclusions

In this work, by introducing predicted inter-residue distance-based features, we developed QDistance, a new protein model QA

method. Both single-model- and multi-models-based inputs are supported in QDistance. Even with simple linear regression, QDistance showed competitive performance with other state-of-the-art methods, as benchmarked on the CASP13 and the CAMEO datasets. In addition, blind test in CASP14 indicated that QDistance was robust and ranked as one of the top performers. The outstanding performance of QDistance can be attributed to the utilization of the distance predicted by deep learning.

Funding

This work was supported by the National Natural Science Foundation of China (11871290, 61873185), KLMDASR; and National Key R&D Program of China (2018YFC1603800, 2018YFC1603802).

Conflict of Interest: none declared.

References

Baldassarre,F. *et al.* (2021) GraphQA: protein model quality assessment using graph convolutional networks. *Bioinformatics*, **37**, 360–366.

Callaway,E. (2020) 'It will change everything': deepMind's AI makes gigantic leap in solving protein structures. *Nature*, **588**, 203–204.

Cao,R. and Cheng,J. (2016) Protein single-model quality assessment by feature-based probability density functions. *Sci. Rep.*, **6**, 23990.

Cao,R. *et al.* (2016) DeepQA: improving the estimation of single protein model quality with deep belief networks. *BMC Bioinformatics*, **17**, 495.

Cao,R. *et al.* (2017) QAcon: single model quality assessment using protein structural and contact information with machine learning techniques. *Bioinformatics*, **33**, 586–588.

Cozzetto,D. *et al.* (2007) Assessment of predictions in the model quality assessment category. *Proteins*, **69** (**Suppl. 8**), 175–183.

Cozzetto,D. *et al.* (2009) Evaluation of CASP8 model quality predictions. *Proteins*, **77** (**Suppl. 9**), 157–166.

Greener,J.G. *et al.* (2019) Deep learning extends de novo protein modelling coverage of genomes using iteratively predicted structural constraints. *Nat. Commun.*, **10**, 3977.

Haas,J. *et al.* (2018) Continuous Automated Model EvaluatiOn (CAMEO) complementing the critical assessment of structure prediction in CASP12. *Proteins*, **86** (**Suppl. 1**), 387–398.

Hiranuma,N. *et al.* (2021) Improved protein structure refinement guided by deep learning based accuracy estimation. *Nat. Commun.*, **12**, 1340.

Hou,J. *et al.* (2019) Deep convolutional neural networks for predicting the quality of single protein structural models. *bioRxiv*, 590620.

Jing,X. and Xu,J. (2020) Improved protein model quality assessment by integrating sequential and pairwise features using deep learning. *Bioinformatics*, **36**, 5361–5367.

Jumper,J. *et al.* (2021) Highly accurate protein structure prediction with AlphaFold. *Nature*, **596**, 583–589.

Kryshtafovych,A. *et al.* (2011) Evaluation of model quality predictions in CASP9. *Proteins*, **79** (**Suppl. 10**), 91–106.

Kryshtafovych,A. *et al.* (2014) Assessment of the assessment: evaluation of the model quality estimates in CASP10. *Proteins*, **82** (**Suppl. 2**), 112–126.

Kryshtafovych,A. *et al.* (2016) Methods of model accuracy estimation can help selecting the best models from decoy sets: assessment of model accuracy estimations in CASP11. *Proteins*, **84** (**Suppl. 1**), 349–369.

Kryshtafovych,A. *et al.* (2018) Assessment of model accuracy estimations in CASP12. *Proteins*, **86** (**Suppl. 1**), 345–360.

Lundstrom,J. *et al.* (2001) Pcons: a neural-network-based consensus predictor that improves fold recognition. *Protein Sci.*, **10**, 2354–2362.

Maghrabi,A.H.A. and McGuffin,L.J. (2017) ModFOLD6: an accurate web server for the global and local quality estimation of 3D protein models. *Nucleic Acids Res.*, **45**, W416–W421.

Manavalan,B. and Lee,J. (2017) SVMQA: support-vector-machine-based protein single-model quality assessment. *Bioinformatics*, **33**, 2496–2503.

Mariani,V. *et al.* (2013) lDDT: a local superposition-free score for comparing protein structures and models using distance difference tests. *Bioinformatics*, **29**, 2722–2728.

McGuffin,L.J. (2008) The ModFOLD server for the quality assessment of protein structural models. *Bioinformatics*, **24**, 586–587.

Olechnovic,K. and Venclovas,C. (2017) VoroMQA: assessment of protein structure quality using interatomic contact areas. *Proteins*, **85**, 1131–1145.

Pages,G. *et al.* (2019) Protein model quality assessment using 3D oriented convolutional neural networks. *Bioinformatics*, **35**, 3313–3319.

Rykunov,D. and Fiser,A. (2007) Effects of amino acid composition, finite size of proteins, and sparse statistics on distance-dependent statistical pair potentials. *Proteins*, **67**, 559–568.

Senior,A.W. *et al.* (2020) Improved protein structure prediction using potentials from deep learning. *Nature*, **577**, 706–710.

Shen,M.Y. and Sali,A. (2006) Statistical potential for assessment and prediction of protein structures. *Protein Sci.*, **15**, 2507–2524.

Shuvo,M.H. *et al.* (2020) QDeep: distance-based protein model quality estimation by residue-level ensemble error classifications using stacked deep residual neural networks. *Bioinformatics*, **36**, i285–i291.

Uziela,K. and Wallner,B. (2016) ProQ2: estimation of model accuracy implemented in Rosetta. *Bioinformatics*, **32**, 1411–1413.

Uziela,K. *et al.* (2016) ProQ3: improved model quality assessments using Rosetta energy terms. *Sci. Rep.*, **6**, 33509.

Uziela,K. *et al.* (2017) ProQ3D: improved model quality assessments using deep learning. *Bioinformatics*, **33**, 1578–1580.

Wallner,B. and Elofsson,A. (2003) Can correct protein models be identified? *Protein Sci.*, **12**, 1073–1086.

Wang,Z. *et al.* (2009) Evaluating the absolute quality of a single protein model using structural features and support vector machines. *Proteins*, **75**, 638–647.

Wang,Z. *et al.* (2011) APOLLO: a quality assessment service for single and multiple protein models. *Bioinformatics*, **27**, 1715–1716.

Won,J. *et al.* (2019) Assessment of protein model structure accuracy estimation in CASP13: challenges in the era of deep learning. *Proteins*, **87**, 1351–1360.

Wu,Y. *et al.* (2007) OPUS-Ca: a knowledge-based potential function requiring only Calpha positions. *Protein Sci.*, **16**, 1449–1463.

Xu,J. (2019) Distance-based protein folding powered by deep learning. *Proc. Natl. Acad. Sci. USA*, **116**, 16856–16865.

Yang,J. *et al.* (2016) ResQ: an approach to unified estimation of B-factor and residue-specific error in protein structure prediction. *J. Mol. Biol.*, **428**, 693–701.

Yang,J. *et al.* (2020) Improved protein structure prediction using predicted interresidue orientations. *Proc. Natl. Acad. Sci. USA*, **117**, 1496–1503.

Zemla,A. (2003) LGA: a method for finding 3D similarities in protein structures. *Nucleic Acids Res.*, **31**, 3370–3374.

Zhang,J. and Zhang,Y. (2010) A novel side-chain orientation dependent potential derived from random-walk reference state for protein fold selection and structure prediction. *PLoS One*, **5**, e15386.

Zhang,Y. and Skolnick,J. (2004) Scoring function for automated assessment of protein structure template quality. *Proteins*, **57**, 702–710.

Zhou,H. and Zhou,Y. (2002) Distance-scaled, finite ideal-gas reference state improves structure-derived potentials of mean force for structure selection and stability prediction. *Protein Sci.*, **11**, 2714–2726.