Briefings in Bioinformatics, 00(0), 2021, 1–10

https://doi.org/10.1093/bib/bbab384
Problem Solving Protocol

ZoomQA: residue-level protein model accuracy estimation with machine learning on sequential and 3D structural features

Kyle Hippe, Cade Lilley, Joshua William Berkenpas, Ciri Chandana Pocha, Kiyomi Kishaba, Hui Ding, Jie Hou, Dong Si and Renzhi Cao

Corresponding author. Renzhi Cao, Department of Computer Science, Pacific Lutheran University, Tacoma, WA 98447, USA. Tel: 2535357409; E-mail: caora@plu.edu

Abstract

Motivation: The Estimation of Model Accuracy problem is a cornerstone problem in the field of Bioinformatics. As of CASP14, there are 79 global QA methods, and a minority of 39 residue-level QA methods with very few of them working on protein complexes. Here, we introduce ZoomQA, a novel, single-model method for assessing the accuracy of a tertiary protein structure/complex prediction at residue level, which have many applications such as drug discovery. ZoomQA differs from others by considering the change in chemical and physical features of a fragment structure (a portion of a protein within a radius *r* of the target amino acid) as the radius of contact increases. Fourteen physical and chemical properties of amino acids are used to build a comprehensive representation of every residue within a protein and grade their placement within the protein as a whole. Moreover, we have shown the potential of ZoomQA to identify problematic regions of the SARS-CoV-2 protein complex.

Results: We benchmark ZoomQA on CASP14, and it outperforms other state-of-the-art local QA methods and rivals state of the art QA methods in global prediction metrics. Our experiment shows the efficacy of these new features and shows that our method is able to match the performance of other state-of-the-art methods without the use of homology searching against databases or PSSM matrices.

Availability: http://zoomQA.renzhitech.com

Kyle Hippe is an undergraduate student at Pacific Lutheran University, his research interest includes Bioinformatics.
Cade Lilley is an undergraduate student at Pacific Lutheran University, his research interest includes Bioinformatics.
Joshua William Berkenpas is an undergraduate student at Pacific Lutheran University, his research interest includes Bioinformatics.
Ciri Chandana Pocha is an undergraduate student at Saint Louis University, her research interest includes Bioinformatics.
Kiyomi Kishaba is an undergraduate student at Pacific Lutheran University, her research interest includes Bioinformatics.
Hui Ding is an associate professor of Center for Informational Biology at University of Electronic Science and Technology of China. Her research is in the areas of computational biology and system biology.
Jie Hou is an Assistant Professor at Saint Louis University, USA. His research is mainly focused on developing data-driven computational methods (particularly machine learning, deep learning, and computational optimization methods) for protein structure and function prediction.
Dong Si is an Assistant Professor at University of Washington Bothell, USA. His research interest is mainly focused on developing and applying machine learning and data mining techniques to solve biomedical, health, and social science problems, such as biomedical image data analysis, protein structure predictions, Parkinson's, Major Depression and social networking data analysis.
Renzhi Cao is an Assistant Professor at Pacific Lutheran University, USA. His research interest is mainly focused on developing and applying machine learning and data mining techniques to solve biomedical problems, such as protein structure and function predictions.
Submitted: 17 May 2021; **Received (in revised form):** 2 August 2021

Introduction

Proteins are the drivers of biological action. They are responsible for everything from locomotion to digestion to the creation of energy. The ability for proteins to complete these functions is largely dependent on their tertiary structure: the 3D arrangement of amino acids, the primary building blocks of proteins. Understanding, predicting and analyzing protein tertiary structure is therefore a large key to breakthroughs in many different areas of biology, such as drug discovery [1, 2].

Advancements in next-generation sequencing technologies allow for efficient and accurate generation of protein sequences. However, methods for determining their structure, such as X-ray crystallography and Nuclear Magnetic Resonance, are time-consuming, costly and in some cases not possible. To address this, researchers in the bioinformatics field have developed numerous computational methods for tertiary structure prediction [3–12]. Computational methods allow for an abundance of structure predictions, leading to an ever increasing need for developing methods to evaluate the quality of these models. The Critical Assessment of Techniques for Protein Structure Prediction (CASP) is designed to benchmark progress of computational protein structure prediction methods and attracts hundreds of research groups around the world every other year. As CASP is gaining more exposure, private companies have begun participating in the recent CASP competitions. In 2020, companies such as Google put their resources toward protein tertiary structure prediction. Google's latest AI algorithm, AlphaFold 2, has revolutionized the field with their performance in CASP14 [5]. Their method has the capability of making high-accuracy protein structure predictions comparable to the expensive and time-consuming lab experiments. This vast improvement in accuracy highlights the importance of evaluating the predicted protein decoy, especially the residue-level accuracy. This is referred to as Estimation of Model Accuracy problem or the protein Quality Assessment (QA) problem.

Computational methods for the QA problem aim to quantify the accuracy of a protein decoy in reference to its native structure but without the knowledge of its ground truth. There are two metrics that address this: GDT-TS (global distance test tertiary structure), which refers to the global structure accuracy, and the LDDT (local distance difference test) for the accuracy of individual amino acids in the prediction[13, 17]. Methods for protein quality assessment are improving, as noted in the CASP 13 experiment [18]. Current methods can be further split into two general approaches: single-model methods which only have access to a single prediction and must estimate its quality [19–25] and consensus models that look at many models at a time to evaluate conserved structural patterns in order to infer quality [26, 27]. Single model methods utilize a range of different input features and methods in order to predict local or global quality. For example, DeepQA, a single-model global method utilizing deep belief networks, predicts global quality using structural, chemical and knowledge-based energy scores achieved noteworthy performance in the CASP12 experiment [28]. Similar to this, SMOQ predicts absolute local qualities of tertiary structure models based on structural features (e.g. secondary structures and chemical properties such as solvent accessibility or hydrophobicity) [29]. Another method, VoroMQA, considers a protein's atoms and uses Voronoi tessellation of those atoms to calculate the knowledge-based potential, that is using contacting surfaces instead of distances, to predict the local quality of amino acids [30]. The features used in most of those methods are similar and need input from both the protein sequence and features from the protein structure, or databases for secondary structure predictions. Lastly, there are other structural analysis-based methods for local quality assessment. For example, GMQ [31] utilizes Conditional Random Fields [32] rather than chemical and physical properties of local environments. In addition, only a few methods work on protein complexes (e.g. ProQ2 [33] used a local linear window in addition to a radius to all other residues interacting with that local linear window).

Here, we propose ZoomQA, a novel single-model quality assessment method based on sequential and 3D structural and chemical features. We benchmarked this tool on the CASP14 released targets and compared its performance to other state-of-the-art methods. The accuracy achieved in local quality metrics without the use of database homology searching indicates the value of the novel features and furthers the validity of using these features to perform protein quality assessment.

The paper is organized as follows. In the Method section, we describe the data acquisition, feature generation for the tool and address the model architecture and training details. In the Results section, we analyze the performance of ZoomQA in comparison to other methods. In the Discussion section, we provide a summary and interpretation of results. In the Conclusion section, we address significant findings from this work and address future directions.

Methods

ZoomQA uses a novel representation of amino acids in the protein structure and addresses the residue level protein quality assessment problem with help of machine learning techniques.

PDB data analysis

The basis of this work stems from analysis of 55 000 crystal PDB's retrieved from http://www.rcsb.org/. The first area of exploration regards something we refer to as a 'fragment structure' which describes a portion of a protein centered around a target amino acid and includes all residues within a radius of consideration *r*, measured in angstroms, ranging from 5 to *r* angstroms. The second area of exploration is an analysis of the occurrence of torsion angle combinations for four categories: the occurrence of angles regardless of secondary structure and three values representing the occurrence of the given torsion angles when considering the secondary structure categories of alpha-helix, beta-sheet and coil.

When analyzing the fragment structures, the first action is extracting the contact map from the PDB. Once the contact map is extracted, we need each residue's amino acid letter code, hydrophobicity, monoisotopic mass, solvent accessibility and isoelectric point. From this, we were able to generate what we call 'Zoom Features', a measure of a certain metric of a fragment structure as the radius of consideration increases. Examples of the relative amino acid density graphs can be found in Supplementary Figures 1 and 2, see Supplementary Data available online at http://bib.oxfordjournals.org/, which demonstrate the typical environment around the amino acid Alanine and Tryptophan. Since we can see different trends for the amino acid environment surrounding a specific amino acid, we assumed that there were trends in these data that could allow us to infer the quality of an amino acid based off of not only the relative change in fragment amino acid composition but other chemical

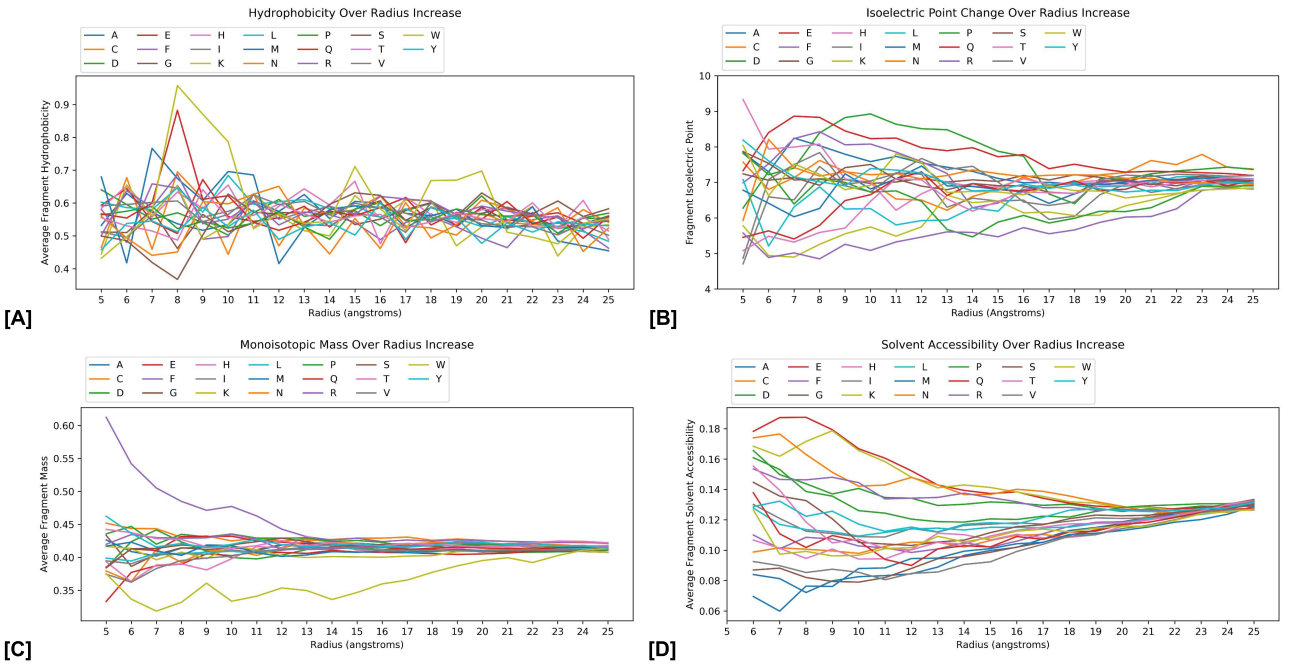

Figure 1. (A) The average hydrophobicity of the fragment structure at a radius of 5 to *r* centered around the target amino acid with letter codes of amino acids represented in the figure legend. (B) The average monoisotopic mass of the fragment structure at radius 5 to *r* centered around the target amino acid with letter codes of amino acids represented in the figure legend. (C) The average solvent accessibility of the fragment structure at radius 5 to *r* centered around the target amino acid with letter codes represented in the figure legend. (D) The fragment structure at radius 5 to *r* centered around the target amino acid with letter codes represented in the figure legend. This is generated by the IsoelectricPoint module in Python's package BioPython [34].

and physical features as well. Figure 1 shows the same change of fragment structure composition as the radius of consideration increases for hydrophobicity, monoisotopic mass, solvent accessibility and isoelectric point, respectively. Once this information is acquired, we can compile the data for the change over the radius expansion. If we consider the amino acid at index 0 in the sequence, most methods consider the sliding window of neighbors in reference to that amino acid in the sequence (i.e. the residue at index 1, 2, 3, etc.) [33]. However, since proteins are not strictly linear, other amino acids that may not be directly next to the target amino acid in the sequence play a role in determining the target amino acid's placement. More importantly, it will be difficult for protein complexes where several chains are not directly connected. To address this, we analyze the 3D radius around an amino acid as the target's environment. We can extract relevant data from this environment which is a more representative feature of how amino acids are placed within proteins.

The second area of exploration was an analysis of dihedral angles that occur in crystal PDBs. In theory, both phi and psi can be in a range (-180, 180). However, in practice, the torsion angles cannot reach that full range. This is due to the amine group of each amino acid. The size, mass, hydrophobicity and other factors all play into the placement of the amine group, and thus the placement of the amine group can sterically prohibit certain angles from existing in nature. Furthermore, torsion angles correlate with the secondary structure of the amino acid and can be used to infer secondary structure. This feature is often used in quality assessment tools, such as DeepQA, AngularQA and SMOQ, but is oftentimes encoded as a 0 or 1 describing whether or not the torsion angles represent the predicted secondary structure [28, 29, 37]. We can get a more representative feature of torsion angles and secondary structure by analyzing the occurrence of these torsion angles regardless of secondary structure and the torsion angle occurrences when the amino acid is within secondary structures (alpha-helices, beta-sheets and coils). Supplementary Figures 3 and 4, see Supplementary Data available online at http://bib.oxfordjournals.org/, show the Ramachandran plots that represent the stability scores for two different amino acids: alanine and proline. Both amino acids exhibit different common torsion angles dependent on the secondary structure they are in. These different values led us to believe that this information could help our model differentiate the quality of an amino acid based off of its predicted secondary structure and its predicted stability within that structure.

In order to generate this stability score, we explored various machine learning techniques before settling on Random Forest regressors. For each amino acid, a Random Forest was trained on 97 200 instances of angles with their normalized relative occurrence and tested on the remaining 32 400 instances of angles that can come from the combination of angles in the range (-180, 180). After exploring hyperparameters of the number of trees and maximum tree depth, the optimal parameters were found by minimizing the testing average error + maximum error + the difference between the testing and training values for maximum and average error.

Data preparation

The feature input for ZoomQA is protein decoys generated from the CASP competitions (CASP6 through CASP13) and obtained from the CASP website http://predictioncenter.org/download_area/ [35]. All PDB's are then filtered and structurally aligned with their respective targets. From there, we processed all the PDB's into matrices containing raw data for each individual amino acid. This resulted in over 26 000 000 examples of residue data.

Next, we filtered out outliers by removing training examples with a mean absolute distance error (in angstroms) of more than 25 angstroms. In order to balance the representation of different qualities, we randomly shuffled all of the data into 60 batches of 430 000 training examples with a 61st batch containing the overflow. We then establish 100 bins with labels ranging from 0.0–0.01 all the way to 0.99–1.00 and select 600 targets for each bin and batch. Since the data are normalized when selecting bins, the bin width is 0.25 angstroms. The number of targets (600) was selected arbitrarily as analysis was performed showing that there were a minimum of 1100 targets in each bin from each batch, guaranteed to fill the quota for each batch. It allowed us to generate a sufficiently large dataset for training while still balancing the data appropriately. This process results in 61 smaller batches of length 60 000 that are balanced in their representation of labels. These are then used as a sample space for training data. One of the 61 batches generated is withheld from training and used as testing data. Validation data come from 33 targets of the CASP14 competition, representing 7736 unique PDBs, which are not used in the training data set.

Features

ZoomQA utilizes 14 properties regarding the chemical and physical properties of the target amino acid and its environment. We will use the term 'fragment' to describe a region of a protein that can be generated by including all amino acids within a radius of consideration *r* of a target amino acid where *r* represents a distance measured in angstroms. Two datasets were generated: one where *r* was set to a minimum of 5 angstroms and a maximum of 25 angstroms, and another where *r* was set to a minimum of 5 angstroms and a maximum of 55 angstroms. For each dataset generated, the step for considering a new 'fragment' was 1 angstrom. The average proportion of protein residues included at the radius of 25 angstroms was 0.5265 and the average proportion of protein residues included at the radius of 55 angstroms was 0.9393. Unless the description states otherwise, all described features are normalized by the following equation:

$$Z_i = (x_i - \min(x))/(\max(x) - \min(x)), \quad (1)$$

where Z_i is the normalized value, x_i is the value we are trying to normalize, min(*x*) represents the minimum value from the set *x* and max(*x*) represents the maximum value from the set *x*. Since the maximum and minimum values come from the set of numbers generated for each residue, the normalized value is guaranteed to be in the range of 0 and 1.

The first property is the average amino acid density of a fragment as the radius of consideration increases from 5 to *r* angstroms. This is a *r* x 20 matrix. The columns of this matrix correspond to the letter codes for all twenty amino acids in alphabetical order. The rows of this matrix represent the radius of consideration in the set {5, 6,...,*r*} (e.g. row 0 represents a radius of consideration of 5 angstroms, row *r* represents a radius of consideration of *r* angstroms). Each element of this matrix represents the relative density of the column amino acid in the fragment with a radius equal to the row + 5 angstroms from the center amino acid. This property adds a total of 1020 features to the total feature set.

The second property is the average hydrophobicity of the fragment of protein in contact with the target amino acid as the radius of consideration increases from 5 to *r* angstroms. This is a vector with length *r*, where index 0 represents the average hydrophobicity of all amino acids within the fragment if the radius of consideration is 5 angstroms. This includes the hydrophobicity of the target amino acid because that influences the overall hydrophobicity of the structure. Since the final model considers a largest radius of 55 angstroms, this property adds 51 features to the total feature set.

The third property is the average monoisotopic mass of the fragment of protein in contact with the target amino acid because the radius of consideration increases from 5 to *r* angstroms. This is a vector with length *r*, where index 0 represents the average mass of all amino acids within the fragment if the radius of consideration is 5 angstroms. This includes the mass of the target amino acid because it influences the overall mass of the structure. This vector adds a total of 51 features to the overall feature set.

The fourth property is the average solvent accessibility of the fragment of protein in contact with the target amino acid as the radius of consideration increases from 5 to *r* angstroms and adds a total of 51 features to the overall feature set. Similarly, this is a vector with length *r*, where index 0 represents the average solvent accessibility of all amino acids within the fragment if the radius of consideration is 5 angstroms. This includes the solvent accessibility of the target amino acid as that influences the overall solvent accessibility of the structure.

The fifth property is the isoelectric point of the fragment of protein in contact with the target amino acid as the radius of consideration increases from 5 to *r* angstroms and once again adds a total of 51 features to the overall feature set. This is a vector with length *r*, where index 0 represents the isoelectric point of the fragment generated by a radius of consideration of 5 angstroms. In order to generate this, we consider the fragment's amino acid sequence as a unique protein and then use the IsoelectricPoint module in Python's package BioPython [34] to calculate the isoelectric point for that fragment.

The sixth property is a length *r* vector that represents the average distance of all amino acids to the target amino acid as the radius of consideration increases from 5 to *r*, adding 51 features to the total feature set. This value is normalized by the radius of gyration, defined as the maximal distance between the target amino acid and any amino acid within the radius of consideration. This is accompanied by the seventh property which is a length *r* vector that represents the standard deviation in the distance between the target acid and its set of amino acids in contact as the radius of consideration increases. This adds a total of 51 features to the input. The input data are normalized before calculating the standard deviation; thus, the feature is normalized upon creation. The input data are normalized by dividing all distances between the target amino acid and its contacts by the radius of gyration.

The eighth property is a vector with length *r* that represents the percentage of the protein as a whole that is within the radius of consideration as the radius increases from 5 to *r* angstroms. This adds a total of 51 features to the input.

The ninth property is very similar to the first feature where we find the relative density of each amino acid for fragments as the radius of consideration increases. For this feature, we weight the occurrence of the amino acids, adding an increased weight if the amino acid in contact is not sequentially in contact with the target amino acid. This generates a contact matrix that emphasizes the contacts resulting from the folding of a protein. This property has dimensions 20 x 51 resulting in 1020 total features being added to the feature set.

The tenth property is the stability score of the target amino acid's torsion angles as generated by the Random Forest models described in the previous section. We generate four stability scores. The first score is the stability score of the angles regardless of the secondary structure. The next three scores are the stability scores of the torsion angles if the secondary structure of this amino acid was an alpha-helix, beta-sheet or coil, respectively. We do not consider the secondary structure of this amino acid coming from the PDB, however. This property adds a total of four values to our input set.

The rest of the properties all pertain to the center amino acid, the target. We include the amino acids' monoisotopic mass, the hydrophobicity, the solvent accessibility, isoelectric point and the torsion angles (two values), all normalized to values between 0 and 1. We also include the amino acid letter code and the secondary structure extracted from protein structure as one-hot encoded vectors. This results in a total of 31 features constituting the final row, and we pad with zeroes to make a square matrix for the model.

Method architecture and prediction process

The final model was trained on 60 000 vectors of the top 100 features for a set of samples of data generated from a maximum radius of consideration of 55 angstroms. This led to a total of 2397 features being generated for each amino acid in the training set. The final model selects the top 100 performing features based on their Pearson correlation to their labels. The Support Vector Machine is trained as the final model with the RBF kernel, a C value of 1.0, an epsilon value of 0.1 and a gamma value of 1.0. The loss function used was absolute distance error per amino acid as calculated by the following equation:

$$\text{Loss} = abs(\hat{y} - y). \tag{2}$$

For the distance calculations, we use the alpha carbon as reference. Supplementary Table 1, see Supplementary Data available online at http://bib.oxfordjournals.org/, describes the hyperparameter search to achieve the final parameters.

Figure 2 represents the overall process of creating local quality predictions based off of a single PDB input. In order to use ZoomQA, PDB format input is required (the input could be single chain protein or complex). From there, each PDB is loaded and features are extracted and compiled into a 47 x 51 dimension matrix. Once this is created, we select the top 100 correlated features as we found from our experiments. The list of these features can be found in Supplementary Table 2, see Supplementary Data available online at http://bib.oxfordjournals.org/. Once the feature set is established, each amino acid is fed through the support vector machine producing a local quality score. This is repeated for all amino acids in the structure. Once all local quality scores are calculated, the global quality score score is calculated using the following equation:

$$\text{Global}\sim\text{Qaulity} = \frac{\sum_{i=1}^{n} \frac{1}{(1+(j_i * (j_i/12))}}{n}, \tag{3}$$

where n is the length of the protein sequence and j is the set of local distance error predictions for each amino acid [13, 38] where x_i is a local error prediction, and 12 is decided by an experiment of evaluating the performance of different values from 5 to 20 on the training dataset (data are not shown).

Results

ZoomQA was benchmarked on the latest CASP 14 dataset and compared with other top-performing single model QA methods. In total, 33 targets are used from CASP 14 dataset and all predictions are filtered so they have the same sequence in each target. In this work, we primarily analyze local quality metrics; global quality analysis appears in Supplementary Figures 5 and 6, see Supplementary Data available online at http://bib.oxfordjournals.org/. For local quality metrics, we evaluate the minimum, average and maximum distance error, LDDT error, and the absolute distance and LDDT SD in error for targets over a distribution of bins.

For performance in local quality, Table 1 demonstrates minimum, average and maximum distance error for the other local QA methods tested, while Table 2 represents the minimum, average and maximum LDDT error for the overall CASP 14 benchmark set including further segregation into the two different stages (20 out of all server models are selected by organizers for stage 1, and stage 2 comes from the top 150 models selected using the Davis-EMA consensus method by the organizer). When measuring distance error, ZoomQA falls short of other methods but manages to be within 10% of each method for minimum, average and maximum error. VoroMQA significantly outperforms ZoomQA in the overall metric with a P-value of 0.008 originating from a T-Test. Similarly, SMOQ outperforms ZoomQA in this metric with a P-value of 0.015, also from a T-Test. When looking at overall LDDT performance, however, ZoomQA manages to beat both VoroMQA and SMOQ in all overall metrics. These values are significant with a P-value of 0.0 as reported by a T-Test. ZoomQA is beaten in Stage 1 performance only by the method SMOQ on the average and maximum LDDT error with a P-value of 6.99e-14 but still outperforms VoroMQA here with a P-value of 3.19e-08. Finally, in stage 2 performance, ZoomQA outperforms both SMOQ and VoroMQA significantly with both with P-values of 0.0. Figure 3 demonstrates the SD of distance error when the target value is in the range 0–19 angstroms with a bin width of 1 angstrom. ZoomQA outperforms SMOQ and VoroMQA at ranges 1–7. It is worth noting that all distances greater than 19 angstroms were grouped into the final bin, causing a large spike in the deviation of distance error. Figure 4 shows the SD of LDDT error when the target value is in 20 bins, 0–0.05, 0.05–0.010 and so on. ZoomQA outperforms VoroMQA for all bins and outperforms SMOQ at bins 0–14.

Discussion

Model

As stated earlier, the final model used in ZoomQA is a Support Vector Machine; this was concluded after rigorous testing using a radius of consideration of 25 angstroms and evaluating Support Vector Machines, Random Forests, Multilayer Perceptrons and Convolutional Neural Networks. Due to time and resource constraints, this extensive testing was not performed on the data generated with a radius of consideration of 55 angstroms, but testing showed that the 55 angstrom data outperformed the experiments on data generated with a radius of consideration of 25 angstroms. Results for these experiments can be found in Supplementary Table 1, see Supplementary Data available online at http://bib.oxfordjournals.org/. Hyperparameter optimization was performed for Support Vector Machines on the radius 55 data with the same parameters described in Supplementary Table 1, see Supplementary Data available online at http://bib.oxfordjournals.org/.

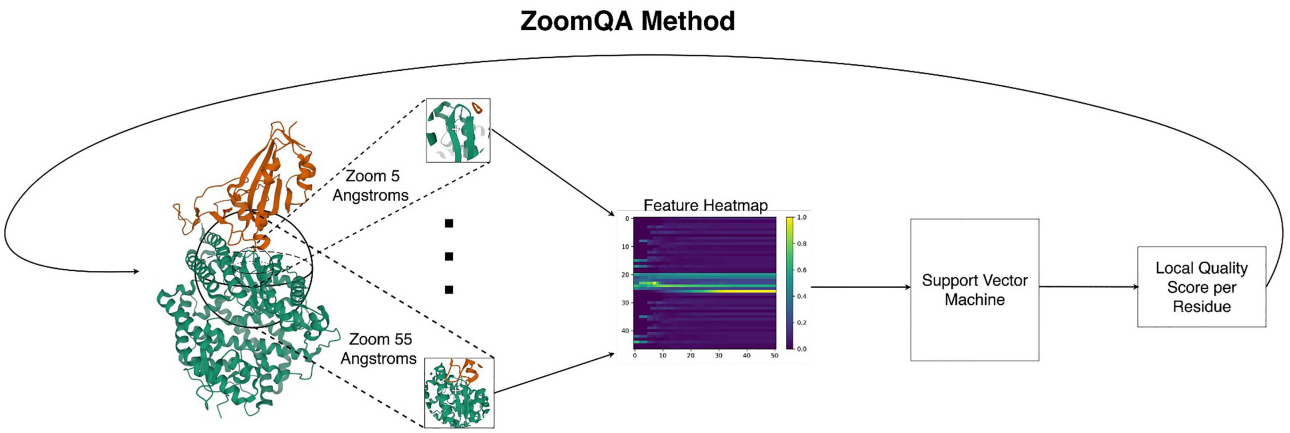

Figure 2. Flowchart of ZoomQA method as a whole. PDB data are taken in and features are generated. Each feature set is then fed into a support vector machine that predicts the local quality of that amino acid. Repeat this process for all amino acids in the PDB.

Table 1. Local QA absolute distance error of ZoomQA versus other tools

Method	Stage	Min. error	Ave. error	Max. error
ZoomQA	Overall	1.156	12.668	248.751
	Stage 1	1.466	22.32	248.751
	Stage 2	1.156	8.53	111.494
SMOQ	Overall	**1.353**	**12.016**	**242.104**
	Stage 1	**1.551**	**20.375**	**242.104**
	Stage 2	1.353	8.307	104.962
VoroMQA	Overall	**1.094**	**11.798**	**234.13**
	Stage 1	**1.381**	**19.844**	**234.13**
	Stage 2	1.094	8.290	100.98

Table 1 represents the performance of ZoomQA versus other tools on mean absolute distance error. Values in bold represent significant values, with *P*-values less than 0.05. Normality of values proven with Chi-Square test for normality.

Table 2. Local QA LDDT error of ZoomQA versus other tools

Method	Stage	Min. error	Ave. error	Max. error
ZoomQA	Overall	**0.063**	**0.159**	**0.61**
	Stage 1	**0.069**	0.18	0.43
	Stage 2	**0.0635**	**0.151**	**0.402**
SMOQ	Overall	0.064	0.189	0.718
	Stage 1	0.075	**0.159**	**0.355**
	Stage 2	0.095	0.199	0.415
VoroMQA	Overall	0.079	0.218	0.824
	Stage 1	0.082	0.12	0.435
	Stage 2	0.099	0.228	0.546

Table 2 represents the performance of ZoomQA versus other tools on LDDT error. Values in bold represent significant values, with *P*-values less than 0.05. Normality of values proven with Chi-Square test for normality.

In addition to these models described above, we attempted to train multiple different ResNet's [14]. Due to time constraints, the model was not adapted from the architecture described by He *et al.* ResNet input shape is a matrix with dimensions 224 x 224 x 3. Since our feature set is much smaller than the required input size, we duplicated the matrix until we reached the necessary size. This was done as to maintain the architecture of the residual blocks, as changing the structure of the blocks lead to extremely poor performance. Testing was done on three standard models, one with 18 layers, one with 34 layers, one with 50 layers. These methods achieved an average error on the CASP 14 validation set of 27.34 angstroms of error, 15.22 angstroms of error and 17.71 angstroms of error, respectively. Further experiments will have to be done to adapt the ResNet architecture to such a small input size, but initial experiments indicate that the conventional architecture does not perform well on the given dataset. More experiments will have to be completed for this technique as well as exploring newer deep-learning techniques such as Transformers [15], which have shown excellent results in many natural language processing tasks.

Feature selection and model optimization

With the generation of all of the features described in Section 2.3, there are a number of data points which are not correlated to the outputs and impede the convergence of a model regardless of the architecture. To get around this, we performed many

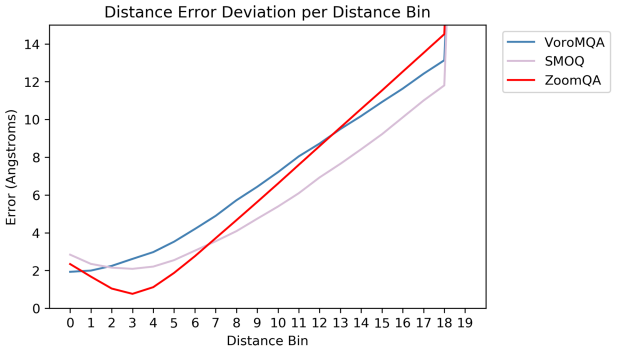

Figure 3. Performance of ZoomQA versus other local QA methods. Each line represents the SD of the absolute distance error when the ground truth value is in each bin.

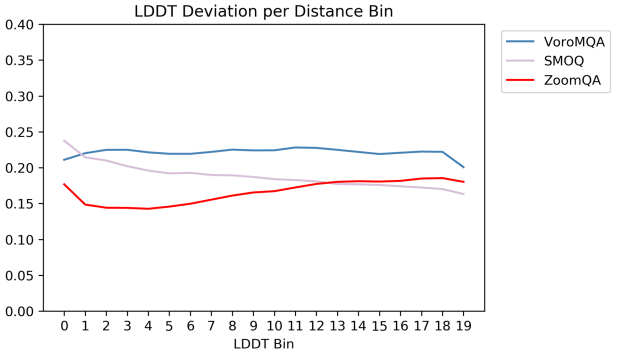

Figure 4. Performance of ZoomQA versus other local QA methods. Each line represents the SD of the LDDT error when the ground truth value is in each bin.

different feature selection methods before finally deciding to select features based off of their raw Pearson Correlation. The feature selection experiments are described below. The final model used the top 100 features. This was concluded after experiments suggesting diminishing returns including more than 100 features. When including 10 features, the model was roughly 10% worse than including 100 features, and adding 10 features incrementally to the model improved performance by roughly 1%. After 100 features were included, adding 10 more features resulted in increases in performance of roughly 0.01%, and with the top 1000 features included, the model only performed 1% better than the top 100 features. The features included can be found in Supplementary Table 2, see Supplementary Data available online at http://bib.oxfordjournals.org/.

Method 1: Pearson correlation

When selecting the features based on feature correlation, we calculate Pearson correlation on each feature and select the top n (in the final model $n = 100$) features based on the absolute value of their feature correlation. Once the features are selected, the features are arranged either into a vector with the top correlated feature at the beginning of the vector and the lowest at the end, or into an n x m matrix where coordinate (0,0) represents the highest correlated feature and the lowest correlation is at coordinate (n, m). The distinction between creating a vector or a matrix depends on the method being tested. Vectors are created when training Random Forests, Support Vector Machines and Multilayer Perceptrons, whereas matrices are used when training Convolutional Neural Networks. This method was used in the final model and achieved all the metrics stated in Section 3.

Method 2: Pearson correlation and clustering

When selecting features based on correlation and clustering, we repeat the process described above and rank each feature based on their Pearson correlation. Next, we perform K-Mean clustering on each feature and generate n number of clusters of features, while n is the number of features you want to select. In our case, n is 100 since we selected the top 100 features. Each feature is selected from each cluster prioritizing features near the centroid until we have n features. These features are then arranged into a vector with the highest correlation feature in the first index and the lowest correlation feature in the last index, or into an n x m matrix where coordinate (0,0) represents the highest correlated feature and the lowest correlation is at coordinate (n, m). The distinction between creating a vector or a matrix depends on the method being tested. Vectors are created when training Random Forests, Support Vector Machines and Multilayer Perceptrons, whereas matrices are used when training Convolutional Ceural Networks. Ultimately, selecting features based on Pearson correlation and clustering resulted in very few effective features being chosen. When the clusters were generated, a select few of the clusters held the majority of the high correlation features. When choosing features from each cluster, we very quickly exhausted the pool of effective features. Using this technique for feature selection resulted in an average absolute distance error of 14.03 angstroms, slightly worse performing than selecting solely based on Pearson correlation.

Method 3: Maximum relevance minimum redundancy

Maximum relevance minimum redundancy [16] is a feature selection technique that aims to maximize the correlation between selected features and their labels while minimizing the correlation between chosen features. In order to select the top n features, we followed the FCQ protocol which specifies that features be scored according to the quotient of their F Statistic when compared with their labels, and the correlation between the feature and the already chosen features. According to the authors, this method lead to the most stable feature selection regardless of downstream machine learning technique used.

Utilizing this method, we selected the top 100 performing feature and came up with 23 features that did not appear in the feature set generated by the raw feature correlation (method 1). Of these features, 6 came from the relative amino acid density change, 6 came from the structure contact matrix, 3 came from the hydrophobicity change feature, 3 from the isoelectric point change and the final 5 came from the vector representing the average distance of amino acids to their center as the radius of consideration increases.

Overall, this led to a decrease in performance of many machine learning models on the dataset. The same hyperparameter optimization was run on these data as with the Pearson correlation feature selection method, and a minimum absolute distance error of 15.24 angstroms was achieved, whereas the original model used in ZoomQA achieved an average mean absolute distance error of 12.67. This could be due to the relatively low correlation of some of the features chosen by this method. In particular, 5 of the new features chosen had a Pearson correlation to their labels of less than 0.05. While these features were unique, they did not contribute any discriminatory ability to the model. This feature selection technique has been demonstrated to be successful in many bioinformatics task, but unfortunately is not a good fit for this method.

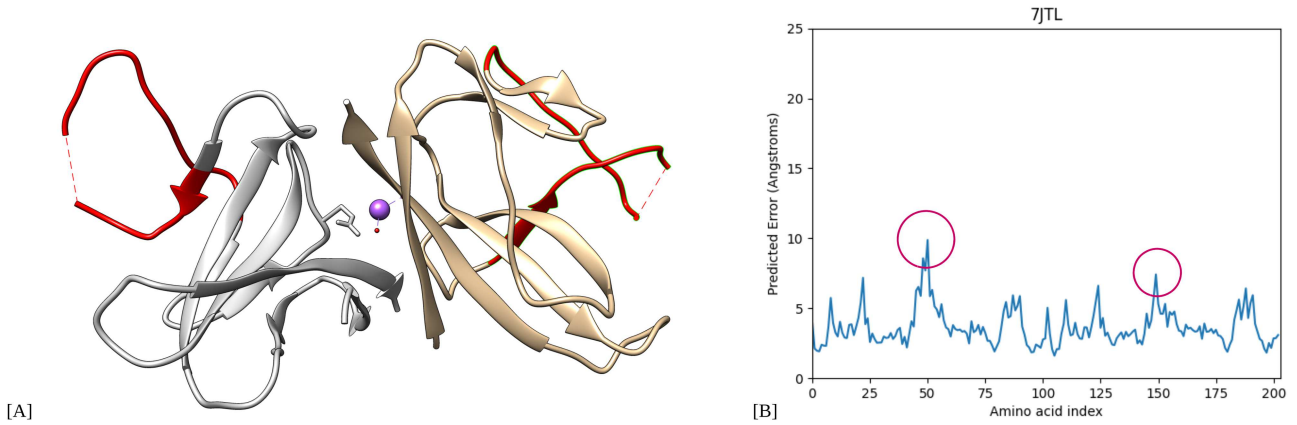

Figure 5. Case study of ZoomQA on PDB 7JTL. (A) Modeled PDB with highlights showing the large error regions identified by the tool. (B) Error graph in absolute distance error for PDB 7JTL.

Efficacy of features

When using all of the features for training, we achieved a mean absolute distance error of 16.23 angstroms. During efforts to improve these results, we found a large number of features that exhibited a very low Pearson correlation with their labels. Upon this realization, we began filtering our feature set based on the features' Pearson correlation with the labels. When evaluating all features, there are 28 features that have a correlation with the targets that is over 0.20. Once we go down to the top 100 features, this correlation only drops to 0.12. Of the top 100 correlated features, 51 come from the change in solvent accessibility as the radius of consideration increases, 28 come from the average distance of amino acids in a fragment to their center, 13 come from the change in hydrophobicity of protein fragments as the radius of consideration increases, all of the secondary structure stability scores are included, as are the hydrophobicity, mass and isoelectric point values for the target amino acid. Supplementary Figure 7, see Supplementary Data available online at http://bib.oxfordjournals.org/, helps visualize the distribution of included features versus the original set of properties. The performance of the raw features on the validation set can be found in Supplementary Table 3, see Supplementary Data available online at http://bib.oxfordjournals.org/, where the Solvent Accesibility Change at radius 46 and 51 performed best with a Pearson Correlation value of 0.32. It is interesting to note that while the isoelectric point values for the target amino acid are included, data regarding the change in isoelectric point of the fragment structures around the target amino acid are not included. In fact, the change in isoelectric point of protein fragments ranks around the 1700 most influential features with an average correlation of around 0.02. This indicates that this feature is not indicative of the quality of an amino acid and was part of the initial troubles when training a model. This is an interesting conclusion since all other ZoomQA features were, at a minimum, in the top third of features based off their correlation. Other features that rank below 0.05 correlation include certain values coming from the change in amino acid density of protein fragments and portions of the structure contact matrix. This was expected as a considerable number of targets do not contain certain amino acids, meaning that there are large portions of the training set that contain a large portion of zeros. The usage of these features improved our mean absolute distance error by roughly 4 angstroms to a value of 12.67 angstroms of error.

Performance

In regards to performance, ZoomQA manages to outperform other well-performing models on the CASP14 benchmark dataset on the LDDT metric and rivals their performance on the local distance metric. This performance is further highlighted in Figure 3 where ZoomQA achieves lower distance error deviation across many of the true values below 7 angstroms, and in Figure 4 where it outperforms other methods in LDDT deviation when below 0.60. This can impact the reliability of our predictions, as if the quality of the model overall is around 2–3 angstroms, our model tends to perform better, but the model does not tend to work as well as other methods on decoys with lower overall accuracy. This is a reflection of the current trends in CASP structure predictions. We achieve these results without the use of a PSSM and any alignment data obtained from BLAST/PSI-BLAST. This highlights the efficacy of the features obtained from ZoomQA and also allows for the tool to be easier to use than other methods that would require the use of a large database or the time-consuming process of performing sequence alignment. For reference, we benchmarked the runtime of the local quality tools SMOQ and VoroMQA on the 214 examples of CASP target T1042, a structure with 276 residues, and found that VoroQA completed the quickest in 635 s (10 min 35 s), ZoomQA finished next in 3031 s (50 min 31 s) and SMOQ, the only tool requiring homology search and PSSM, took 44 752 s (12 h 25 min). It also takes out some of the variation in results as sequences without homology in the provided database could harm the performance of methods that require them.

Case study on SARS-CoV-2 protein complex

We used our new ZoomQA tool to validate the quality of experimental complex PDB 7JTL, which is the structure of SARS-CoV-2 ORF8 accessory protein and has two chains. Figure 5A shows the structure of the experimental complex, and Figure 5B demonstrates the prediction from our ZoomQA. As we can see in Figure 5B, there are a few peaks in our prediction (see the two cases circled in red), which indicates those regions may have serious flaws. We identified those two regions in the experimental complex and highlighted them in red (see Figure 5A), and indeed, we found out that there are gaps in the experimental complex. This case shows the potential of using ZoomQA to identify serious flaws in the experimental or predicted protein structure.

We also evaluated the capability of ZoomQA in scoring protein–protein docked poses, though ZoomQA was not originally trained for ranking protein complexes. The global quality score of the docked pose is estimated by the set of local distance error predictions for the selected residues around the interface surface using equation (2). The interface residues are determined using a distance cut-off of 8 angstroms between any two CB atoms (or CA for Gly) in the pairs of interacting chains. The neighboring amino acids (i.e. window size 24) surrounding the selected interface residues are included for the global quality estimation. The performance was compared to two other scoring algorithms that were developed for ranking protein docked decoys: DockScore [39] and ZRANK [40]. Targets were collected from the CAPRI Docking competition [41]. The dataset contained a total of 15 targets, each with protein decoys predicted by diverse docking algorithms. These decoys were collected and re-ranked using ZoomQA, DockScore and ZRANK. The quality of the top-ranked decoys (top1 and top5) selected by each method was compared using similarity scores (i.e. lrmsd, irmsdbb, irmsdsc) [42]. These metrics are also used in the official evaluation results and provided in the CAPRI scoring decoys database. The results of model scoring are provided in Supplementary Tables 3 and 4, see Supplementary Data available online at http://bib.oxfordjournals.org/. ZoomQA is comparable with the other two methods on average according to the accuracy of top 1 and top 5 selected decoys. Particularly, ZoomQA picked a near-native decoy for target T47 that has an lrmsd 1.92, which is significantly better than the model chosen by DockScore.

Conclusion

In this paper, we purpose a new residue-level protein model quality assessment tool, ZoomQA, which utilizes novel sequential and 3D structural features to grade the local quality of a tertiary structure prediction. It outperforms state-of-the-art methods in local quality, particularly when measuring LDDT, and although our method is not trained using GDT-TS as a metric, the conversion of local quality assessment scores to global quality assessment score rivals other methods when compared using the GDT-TS metric. Our method only needs protein structure as input and points out a new direction for evaluating the quality of predicted protein decoy and protein complex.

In the future, we plan on fine-tuning the ZoomQA features to minimize the production of low-correlated features. This would eliminate the need for feature selection based on Pearson correlation. Additionally, we plan on incorporating more ZoomQA features to better describe the 3D structure of proteins. We would also like to explore different forms of deep learning to gain insights into our data that may not be possible with conventional machine learning techniques. Finally, we plan to analyze more details of each feature and also explore the deep learning techniques, such as the DeepAccNet utilization of deep learning techniques for protein structure refinement [43].

Key Points

- The change of chemical and physical features in 3D environment with different radius is important feature for protein structure prediction.
- The ZoomQA works well in identifying regions in protein complexes that potentially have serious flaws, and demonstrated good performance compared with other methods in the CASP dataset.
- All new features proposed by ZoomQA except for the change in isoelectric point show great performance compared with traditional features for validating the quality of protein structures.
- Feature selection based on Pearson correlation is effective to improve the performance of our Support Vector Machine model to evaluate the local quality of protein structures.

Authors' contributions

R.C. and K.H. conceived the experiment(s), K.H. and R.C. conducted the experiment(s), K.H., C.L., J.B., J.H., C.P. and R.C. analyzed the results and built the website. K.H., C.L., J.B., K.K., H.D., J.H. and D.S. wrote and reviewed the manuscript.

Funding

Natural Sciences Undergraduate Research Program at Pacific Lutheran University. This material is based upon work supported by Google Cloud.

References

1. Jacobson M, Sali A. Comparative protein structure modeling and its applications to drug discovery. *Annu Rep Med Chem* 2004; **39**:259–74.
2. Stephenson N, Shane E, Chase J. J., Ries, D., Justice, N., Zhang, J., Chan, L. and Cao, R. Survey of machine learning techniques in drug discovery. *Curr Drug Metab* 2019; **20**: 185–93.
3. Ma J, Wang S, Zhao F, *et al*. Protein threading using context-specific alignment potential. *Bioinformatics* 2013; **29**:i257–65.
4. Yang J, Anishchenko I, Park H, *et al*. Improved protein structure prediction using predicted interresidue orientations. *Proc Natl Acad Sci* 2020; **117**:1496–503.
5. Jumper J, Evans R, Pritzel A, *et al*. A. High accuracy protein structure prediction using deep learning. *Fourteenth Critical Assessment Of Techniques For Protein Structure Prediction (abstract Book)* **22**:24 (202).
6. Si D, Moritz S, Pfab J, *et al*. Deep learning to predict protein backbone structure from high-resolution cryo-EM density maps. *Sci Rep* 2020; **10**:1–22.
7. Hou J, Wu T, Cao R, *et al*. Protein tertiary structure modeling driven by deep learning and contact distance prediction in CASP13. *Proteins: Structure, Function, And Bioinformatics* 2019; **87**:1165–78.
8. Hou J, Wu T, Guo Z, *et al*. *The MULTICOM Protein Structure Prediction Server Empowered by Deep Learning and Contact Distance Prediction*. Springe, 2020.
9. Rohl C, Strauss C, Misura K, *et al*. Protein structure prediction using Rosetta. *Methods Enzymol* 2004; **383**:66–93.
10. Zhang Y. I-TASSER server for protein 3D structure prediction. *Bmc Bioinformatics*. 2008; **9**:1–8.
11. Wei G. Protein structure prediction beyond AlphaFold. *Nature Machine Intelligence* 2019; **1**:336–7.
12. Källberg M, Wang H, Wang S, *et al*. Template-based protein structure modeling using the RaptorX web server. *Nat Protoc* **7**:1511–152 (Jul).
13. Zemla A. LGA: a method for finding 3D similarities in protein structures. *Nucleic Acids Res* 2003; **31**:3370–4.

14. He K, Zhang X, Ren S, *et al*. *Deep residual learning for image recognition*, 2015.
15. Vaswani A, Shazeer N, Parmar N, *et al*. Attention is all you need. *CoRR*, abs/1706.03762, 2017. *URL*. http://arxiv.org/abs/1706.03762.
16. Zhao Z, Anand R, Wang M. *Maximum relevance and minimum redundancy feature selection methods for a marketing machine learning platform*, 2019.
17. Mariani V, Biasini M, Barbato A, *et al*. lDDT: a local superposition-free score for comparing protein structures and models using distance difference tests. *Bioinformatics* 2013; **29**:2722–8.
18. Cheng J, Choe M, Elofsson A, *et al*. Estimation of model accuracy in CASP13. *Proteins: Structure, Function, And*. *Bioinformatics* 2019; **87**:1361–77.
19. Cao R, Bhattacharya D, Adhikari B, *et al*. Large-scale model quality assessment for improving protein tertiary structure prediction. *Bioinformatics* 2015; **31**:i116–23.
20. Wallner B, Elofsson A. Can correct protein models be identified? *Protein Sci* 2003; **12**:1073–86.
21. Manavalan B, Lee J. SVMQA: support–vector-machine-based protein single-model quality assessment. *Bioinformatics* 2017; **33**:2496–503.
22. Cao R, Adhikari B, Bhattacharya D, *et al*. QAcon: single model quality assessment using protein structural and contact information with machine learning techniques. *Bioinformatics* 2017; **33**:586–8.
23. Shen M, Sali A. Statistical potential for assessment and prediction of protein structures. *Protein Sci* 2006; **15**: 2507–24.
24. Cao R, Cheng J. Protein single-model quality assessment by feature-based probability density functions. *Sci Rep* 2016; **6**:1–8.
25. Uziela K, Shu N, Wallner B, *et al*. ProQ3: Improved model quality assessments using Rosetta energy terms. *Sci Rep* 2016; **6**:1–10.
26. Lundström J, Rychlewski L, Bujnicki J, *et al*. Pcons: A neural-network–based consensus predictor that improves fold recognition. *Protein Sci* 2001; **10**:2354–62.
27. Wang Z, Eickholt J, Cheng J. APOLLO: a quality assessment service for single and multiple protein models. *Bioinformatics* 2011; **27**:1715–6.
28. Cao R, Bhattacharya D, Hou J, *et al*. DeepQA: improving the estimation of single protein model quality with deep belief networks. *Bmc Bioinformatics* 2016; **17**:1–9.
29. Cao R, Wang Z, Wang Y, *et al*. SMOQ: a tool for predicting the absolute residue-specific quality of a single protein model with support vector machines. *Bmc Bioinformatics*. 2014; **15**:1–8.
30. Venclovas K. VoroMQA: Assessment of protein structure quality usi. *Journal proteins* **8**:1131–45 (201).
31. Shin W-H, Kang X, Zhang J, *et al*. Prediction of local quality of protein structure models considering spatial neighbors in graphical models. *Sci Rep* 2017; **7**(1). 10.1038/srep40629.
32. Tang M, Tan KM, Tan XL, *et al*. Graphical models for protein function and structure prediction. *Biological Knowledge Discovery Handbook* 2013;191–222. 10.1002/9781118617151.ch09.
33. Ray A, Lindahl E, Wallner B. Improved model quality assessment using ProQ2. *Bmc Bioinformatics* 2012; **13**:1–12.
34. Cock P, Antao T, Chang J, *et al*. Biopython: freely available Python tools for computational molecular biology and bioinformatics. *Bioinformatics* 2009; **25**:1422–3.
35. Berman H, Westbrook J, Feng ZG, *et al*. The protein data bank. *Nucleic Acids Res* 2000; **28**:235–42.
36. Berman H, Westbrook J, Feng ZG, *et al*. The protein data bank, 1999. *International Tables For Crystallography* 2006.
37. Conover M, Staples M, Si D, *et al*. AngularQA: protein model quality assessment with LSTM networks. *Computational And Mathematical Biophysics* 2019; **7**:1–9.
38. Zhang Y, Skolnick J. Scoring function for automated assessment of protein structure template quality. *Proteins: Structure, Function, And Bioinformatics* 2004; **57**:702–10.
39. S. Malhotra, O. K. Mathew, and R. Sowdhamini. *Dockscore: a webserver for ranking protein-protein docked poses*, Apr 2015. URL https://bmcbioinformatics.biomedcentral.com/articles/10.1186/s12859-015-0572-6.
40. Pierce B, Weng Z. Zrank: Reranking protein docking predictions with an optimized energy function. *Proteins: Structure, Function, and Bioinformatics* 2007; **67**(4): 1078–86. 10.1002/prot.21373.
41. Lensink MF, Wodak SJ. Score_set: A capri benchmark for scoring protein complexes. *Proteins: Structure, Function, and Bioinformatics* 2014; **82**(11): 3163–9. 10.1002/prot.24678.
42. Wiehe K, Peterson MW, Pierce B, *et al*. Protein-protein docking: Overview and performance analysis. *Protein Structure Prediction* 2008;283–314. 10.1007/978-1-59745-574-9_11.
43. Hiranuma N, Park H, Baek M, *et al*. Improved protein structure refinement guided by deep learning based accuracy estimation. *Nat Commun* 2021; **12**:1–11.