Article

P3CMQA: Single-Model Quality Assessment Using 3DCNN with Profile-Based Features

Yuma Takei [1,2] and Takashi Ishida [1,*]

1. Department of Computer Science, School of Computing, Tokyo Institute of Technology, Ookayama, Meguro-ku, Tokyo 152-8550, Japan; takei@cb.cs.titech.ac.jp
2. Real World Big-Data Computation Open Innovation Laboratory (RWBC-OIL), National Institute of Advanced Industrial Science and Technology (AIST), Aomi, Koto-ku, Tokyo 135-0064, Japan
* Correspondence: ishida@c.titech.ac.jp

Abstract: Model quality assessment (MQA), which selects near-native structures from structure models, is an important process in protein tertiary structure prediction. The three-dimensional convolution neural network (3DCNN) was applied to the task, but the performance was comparable to existing methods because it used only atom-type features as the input. Thus, we added sequence profile-based features, which are also used in other methods, to improve the performance. We developed a single-model MQA method for protein structures based on 3DCNN using sequence profile-based features, namely, P3CMQA. Performance evaluation using a CASP13 dataset showed that profile-based features improved the assessment performance, and the proposed method was better than currently available single-model MQA methods, including the previous 3DCNN-based method. We also implemented a web-interface of the method to make it more user-friendly.

Keywords: model quality assessment (MQA); estimation of model accuracy (EMA); protein structure prediction; machine learning; deep learning; 3DCNN; CASP

1. Introduction

Protein structure prediction is one of the most important issues in bioinformatics, and various prediction methods have been developed such as Alphafold [1]. Currently, there is no best structure prediction method for any target, and thus, the user practically uses multiple methods to generate structure models. In addition, each method often generates multiple structure models. Therefore, model quality assessment (MQA), which estimates the quality of the model structures, is required for selecting a final structure model. There are two kinds of MQA, single-model and consensus method. The single-model methods use only one model structure as the input, whereas the consensus methods need to take a consensus of multiple model structures. The performance of consensus methods is often higher in a case that many high-quality model structures are available, such as CASP [2]. However, consensus methods do not perform well with fewer model structures. Furthermore, the consensus methods use single-model MQA scores for the features [3]. Therefore, the development of high-performance single-model MQA methods is important.

To improve the single-model MQA, various methods have been developed. Currently, one of the best methods is based on the three-dimensional convolutional neural network (3DCNN) [4,5]. 3DCNN is a derivative of deep neural networks and can handle 3D-coordinates of protein atoms directly. However, these methods show only comparable performance to the other existing methods. One of the major problems in the previous 3DCNN-based methods was with their features, which were based on only atom types. They did not use any features related to evolution, such as sequence profile information. Various studies have shown the importance of sequence profile information, and the addition of such information could significantly improve the performance [6,7].

In this study, we developed a single-model MQA method for protein structures based on 3DCNN using sequence profile-based features. The proposed method, profile-based three-dimensional neural network for protein model quality assessment (P3CMQA), has basically the same neural network architecture as that of a previous 3DCNN-based method [5] but uses additional features, i.e., sequence profile information, and predicted local structures from the sequence profiles. Comparison with state-of-the-art methods showed that P3CMQA performed considerably better than them. P3CMQA is available both as a web application and a stand-alone application at http://www.cb.cs.titech.ac.jp/p3cmqa and https://github.com/yutake27/P3CMQA, respectively.

2. Materials and Methods

A 3DCNN-based method by Sato and Ishida (Sato-3DCNN) used only 14 atom types as the features [5]. In addition to the atom-type features, here, we tried to improve the performance by adding sequence profile-based features. We used a position-specific scoring matrix (PSSM) of a residue obtained by PSI-BLAST [8] to incorporate evolutionary information. We also added predicted local structures from the sequence profiles: predicted secondary structure (predicted SS) and predicted relative solvent accessibility (predicted RSA). SS and RSA are predicted by SSpro/ACCpro [9]. The overall workflow is shown in Figure 1.
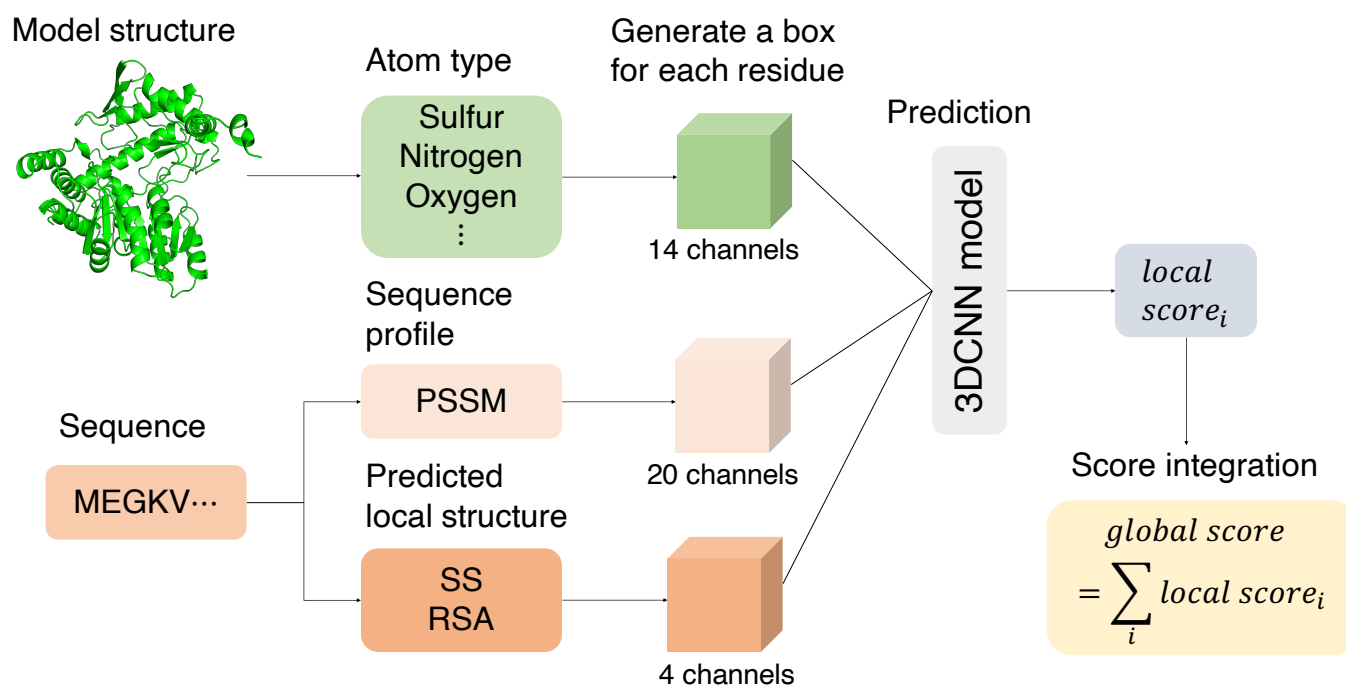

Figure 1. Overall workflow of this work. First, a bounding box is generated for each residue from the coordinate information of the model structure. Then, 14-dimensional atom-type features are obtained from the model structure. In addition, 20-dimensional sequence profile features and 4-dimensional local structure features are generated from the sequences. These features are then input to the three-dimensional convolutional neural network to predict a local score for each residue. Finally, the local scores are averaged to obtain a global score for the entire model.

2.1. Featurization

2.1.1. Making Residue-Level Bounding Box

We made a residue-level bounding box in the same way as in Sato-3DCNN. The length of one side of the bounding box is 28Å and the box is divided into 1Å voxels. The axis of the bounding box is defined by the orthogonal basis calculated from the C-CA vector and N-CA vector and the cross product of C-CA and N-CA. By fixing the axis, the problem of rotation need not be considered.

2.1.2. Atom-Type Features

In Sato-3DCNN, 14 features corresponding to combinations of atoms and residues based on Derevyanko-3DCNN [4] were used. The detail of the 14 atom-type features is shown in Table A1. In this work, we used all 14 features as atom-type features and all these features are binary indicators for each voxel.

2.1.3. Evolutionary Information

We used the position-specific scoring matrix (PSSM) as evolutionary information. We generated PSSM using PSI-BLAST [8] against the Uniref90 database (downloaded April 2019) with two iterations. The maximum and minimum values of PSSM in the training dataset were 13, -13; thus, we normalized PSSM using the following formula.

$$Normalized\ PSSM = \frac{PSSM + 13}{26}$$

PSSM is a residue-level feature, but we assigned PSSM to all the atoms that made up the residue.

2.1.4. Predicted Local Structure

We used predicted local structure as a feature, and the actual local structure of the model structure was not used because it is considered to be observable using 3DCNN. Predicted secondary structure (SS) and predicted relative solvent accessibility (RSA) were used as the predicted local structure. SS was predicted from the sequence profile using SSpro [9]. SSpro predicts SS into three classes; therefore, we use predicted SS in the form of a three-dimensional one-hot vector. RSA was predicted from the sequence profile using ACCpro20 [9]. ACCpro20 predicts RSA from 5% to 95%, as a 5% increase; thus, we divided the predicted RSA by 100 and scaled it from 0 to 0.95. Like PSSM, predicted SS and RSA are residue-level features, but we assigned them to all atoms.

2.2. 3DCNN Training

2.2.1. Network Architecture

The same network architecture used in Sato-3DCNN, which consists of six convolutional layers and three fully connected layers, was used for the neural network architecture. To avoid overfitting, the batch normalization [10] was applied after each convolutional layer, and PReLU [11] was applied as an activation function after the batch normalization. We show the detail of the network architecture in Table A2. We examined other network architectures such as residual network [12], but they did not improve performance significantly.

2.2.2. Label and Score Integration

We trained our models by supervised learning. We generated a bounding box for each residue and trained the model; hence, a label is required, which represents the local structure quality of each residue. Thus, we used lDDT [13] as a label for each residue, as with Sato-3DCNN. lDDT is a local structural similarity score that is superposition-free. It evaluates whether the local atomic environment of the reference structure is preserved in a protein model.

We trained models by binary classification, as in Sato-3DCNN. Therefore, we set the threshold to 0.5 and converted lDDT to positive or negative. Sigmoid cross entropy was used as a loss function. We also tried regression learning, but the predictive performance decreased.

As mentioned above, we trained a 3DCNN-model for each residue, and the 3DCNN-model returned a score for a residue in the prediction. Thus, we needed to integrate the predicted score of each residue into the score of the entire protein structure model to predict its quality. The score of the entire protein structure model was calculated as the mean of the score for each residue, as with Sato-3DCNN.

In this study, we compared our method to other methods to predict the quality of the entire protein structure model. We used GDT_TS [14] as a label that represents the quality of the entire structure model in the evaluation.

2.2.3. Parameters

We used AMSGrad [15] as an optimizer and set the learning rate to 0.001. We used 32 Nvidia Tesla P100 GPUs and performed distributed learning. We set the batch size of each GPU to 32, and the total batch size was 1024.

2.2.4. Training Process

We trained the model using the architecture, labels, loss function, and optimization function described above. To avoid overfitting, the average Pearson correlation for each target of the validation dataset was calculated for each epoch, and the model of the epoch with the best Pearson correlation was selected. The training of the model took about two hours for each epoch and was completed within 5 epochs.

2.3. Dataset

For the training dataset, we used the predicted protein model structures and native structures from CASP7 to CASP10 [16–19]. The total number of protein targets was 434, and the total number of protein structure models was 116,227. We excluded the protein targets with fewer than 50 models. Finally, the training dataset included 421 protein targets and 116,096 protein structure models. We randomly split each protein target into the training dataset and the validation dataset (8 to 2). In the training dataset, due to a large number of structure models, we randomly selected 25% models for each protein target. As a result, we used 23,405 structure models and 4,666,494 residues for training.

For the test dataset, we used CASP12 and CASP13 datasets [2,20]. Each dataset was divided into stage 1 and stage 2. However, only stage 2, which has 150 structure models per protein target, was used. The GDT_TS for each structure model was obtained from the CASP website. In CASP, protein targets for which the best-predicted model had GDT_TS less than 40 were not considered in the evaluation. Thus, we excluded such protein targets, and finally, the number of protein targets in CASP12 and CASP13 was 51 and 66, respectively.

In both the training dataset and the test dataset, we used SCWRL4 [21] to optimize the side-chain conformation. By using SCWRL4, it is possible to evaluate the quality of model structures that contain only main-chain. The details of both datasets are shown in Table 1.

Table 1. Details of the training dataset and the test dataset.

	Dataset	Number of Targets	Number of Model Structures per Target
Train	Train	337	69.5
	Validation	85	271.7
Test	CASP12	51	149.9
	CASP13	66	149.9

2.4. Performance Evaluation

We used the following four measures to evaluate the performance of our method.

1. The average Pearson correlation coefficient for each target
2. The average Spearman correlation coefficient for each target
3. The average GDT_TS loss for each target
4. The average Z-score for each target

Pearson correlation coefficient is calculated from the correlation between the predicted quality score and the GDT_TS. Spearman correlation coefficient is calculated from the same data as the Pearson correlation coefficient. GDT_TS loss is the difference between the GDT_TS of the best model and the GDT_TS of the model with the highest prediction

score. Therefore, a lower GDT_TS loss represents better performance. Z-score is a standard score of a selected model, which is the difference between the GDT_TS of a selected model and the population mean in units of standard deviation. A higher Z-score indicates a better performance.

3. Results and Discussion

3.1. Training Result for Each Feature

We added evolutionary information and predicted local structural features to improve the performance of the model. Thus, we evaluated the contribution of each feature using the validation dataset. To compare the prediction performance of trained 3DCNN-models using each feature, the average of the Pearson correlation between the predicted global score and GDT_TS for each target was used because we selected the model with the best Pearson correlation on the validation dataset during training. We show the results in Table 2. The model using only atom-type features was the same as Sato-3DCNN except for the split between the training and validation sets and the optimizer. This result shows that the prediction performance was greatly improved by adding profile-based features. Both evolutionary information and predicted local structural features contributed to improving the performance, and the latter had a slightly larger contribution. When we used all these features, the best performance was achieved. The results for metrics in addition to the Pearson correlation on the validation dataset are shown in Table 3. For the other metrics, the best performance was obtained when all features were used.

Table 2. Prediction performance for the combination of each feature.

Atom-Type Features	Evolutionary Information	Predicted Local Structure	Pearson (Validation)
✓	✗	✗	0.757
✓	✓	✗	0.834
✓	✗	✓	0.847
✗	✓	✓	0.858
✓	✓	✓	**0.865**

Columns 1–3 represent combinations of features referenced in Section 2.1. The fourth column represents the average Pearson correlation coefficient for each target in the validation dataset. The best performance in the fourth column is in boldface.

Table 3. Performance in the CASP12 stage 2 test dataset.

Method	Pearson	Spearman	Loss	Z-Score
Proposed	**0.856** $(-)$	**0.782** $(-)$	**4.319** $(-)$	**1.240** $(-)$
Sato-3DCNN (AMSGrad)	0.746 (4.67×10^{-9})	0.675 (5.31×10^{-7})	5.530 (4.89×10^{-1})	1.139 (4.99×10^{-1})
ProQ3D	0.750 (8.18×10^{-9})	0.672 (3.41×10^{-7})	7.989 (4.82×10^{-3})	0.922 (7.38×10^{-3})
SBROD	0.682 (9.87×10^{-10})	0.612 (1.87×10^{-7})	7.063 (3.47×10^{-2})	0.967 (4.23×10^{-2})
VoroMQA	0.671 (1.11×10^{-9})	0.592 (1.77×10^{-9})	7.649 (4.30×10^{-2})	0.963 (4.30×10^{-2})

The first column represents the method name. The second and third columns show the average Pearson and Spearman correlation coefficients per target. The fourth and fifth columns show the average GDT_TS loss and average Z-score of selected models for each target. The values in parentheses are the *p*-values calculated by the Wilcoxon signed-rank test for the difference between the proposed method and the comparison method. The best values and *p*-values smaller than 0.01 are shown in bold.

3.2. Comparison with Other Methods on CASP Datasets

We compared the proposed method with major single-model MQA methods on the CASP12, CASP13 datasets. We used Sato-3DCNN [5], ProQ3D [22], SBROD [23], and VoroMQA [24] as comparison methods. Sato-3DCNN is the direct predecessor of this research. ProQ3D is a deep neural network-based method using profile-based features. SBROD is a method using ridge regression with various geometric structural features. VoroMQA is a method using a statistical potential that depends on interatomic contacts by Voronoi tessellation.

The results of all comparison methods were executed by us. Sato-3DCNN is a retrained model using the same training data as that of this method with an optimizer changed from SMORMS3 [25] to AMSGrad. ProQ3D, which was last updated on 8 October 2017, was downloaded, and we used the S-score version model. The version of VoroMQA was 1.19.2352. SBROD, which was last updated on 14 August 2019, was downloaded, and we used the model that was trained using CASP5-10 for the training dataset.

The result for the CASP12 stage 2 dataset is shown in Table 3. The proposed method showed better performances for each metric. To check whether there is a statistically significant difference between the performance of the proposed method and the comparison method, we conducted the Wilcoxon signed-rank test at a significance level of 0.01. As a result, it is shown that there are statistically significant differences between the proposed method and the comparison methods in terms of Pearson and Spearman correlation. For GDT_TS loss and Z-score, there is no significant difference, but the proposed method performed better than the other methods. Also, the results for the CASP13 stage 2 dataset is shown in Table 4. Similarly, the results for CASP13 stage 2 showed that the proposed method was better than the other existing methods.

Table 4. Performance in the CASP13 stage 2 test dataset.

Method	Pearson	Spearman	Loss	Z-Score
Proposed	**0.797** $(-)$	**0.757** $(-)$	**5.708** $(-)$	**1.264** $(-)$
Sato-3DCNN (AMSGrad)	0.748 $(\mathbf{1.09 \times 10^{-5}})$	0.703 $(\mathbf{1.84 \times 10^{-5}})$	6.527 (4.44×10^{-1})	1.167 (3.93×10^{-1})
ProQ3D	0.686 $(\mathbf{1.42 \times 10^{-9}})$	0.638 $(\mathbf{2.03 \times 10^{-10}})$	9.482 (2.16×10^{-2})	0.990 (2.29×10^{-2})
SBROD	0.674 $(\mathbf{1.95 \times 10^{-9}})$	0.637 $(\mathbf{3.38 \times 10^{-9}})$	10.014 $(\mathbf{2.29 \times 10^{-4}})$	0.930 $(\mathbf{5.99 \times 10^{-4}})$
VoroMQA	0.676 $(\mathbf{2.38 \times 10^{-9}})$	0.624 $(\mathbf{2.52 \times 10^{-11}})$	12.105 $(\mathbf{1.73 \times 10^{-3}})$	0.786 $(\mathbf{1.15 \times 10^{-3}})$

The legends are the same as those in Table 3.

We also compared the performance for each category of targets that represent the difficulty of the prediction released by CASP. We used three categories: Free Modeling (FM), Free Modeling/Template-Based Modeling (FM/TBM), and Templated Based Modeling (TBM). The average Pearson correlation coefficient for each category in CASP13 are shown in Table A4 and Figure A1. The proposed method is the best for each category. For FM/TBM and TBM categories, there are significant differences between the proposed method and the other methods. For the FM category, there is no significant difference due to the small number of targets (12 targets).

4. Web Tool

The proposed method showed the best performance as a single-model MQA method. However, it is impractical if it is not readily available to the user. For example, ProQ3D [22] provides a web interface, but it simply outputs the global score for the entire model

structure and the local score for each residue as text. Thus, we implemented a web-based tool to make it even more user-friendly to check the results.

Figure 2 shows the input page of the web tool. The required inputs are an email address and a model structure in PDB format or mmCIF format. A target sequence in FASTA format can be entered as an option. If the sequence is not entered, a sequence generated from the model structure file is used to construct the profile-based features. When the prediction is finished, users will receive an email with an URL of the prediction result.

Figure 2. The input page of the web tool. The email address and the model structure in PDB format or mmCIF format are required inputs, and the sequence in FASTA format is an optional input. You can check the number of running jobs and the number of waiting jobs.

The execution time depends on the size of a protein. For a new protein, it takes a little longer because it is necessary to generate a sequence profile. If the sequence length is about 500, it will take about 30 min to complete. However, for proteins that have been processed before, the profile generation can be skipped, and the execution time is about one minute.

Figure 3 shows an example of the output of the web tool. It outputs a global score and a bar chart of residue-wise local scores. A local score is an estimation of lDDT and it takes values between 0 and 1. A higher value shows a better model structure. It also provides a 3D view of a model structure colored according to the prediction score for each residue. The model structure is colored in rainbow colors, with red areas representing low local scores and blue areas representing high local scores. It uses NGL viewer [26], and thus, the user can move and rotate the model structure to visually check the quality.

Besides, the predictions can be downloaded in several formats. It is possible to download a PDB file with prediction scores set to the b-factor of each residue. This makes it possible to check the detailed structure and prediction scores in your local environment.
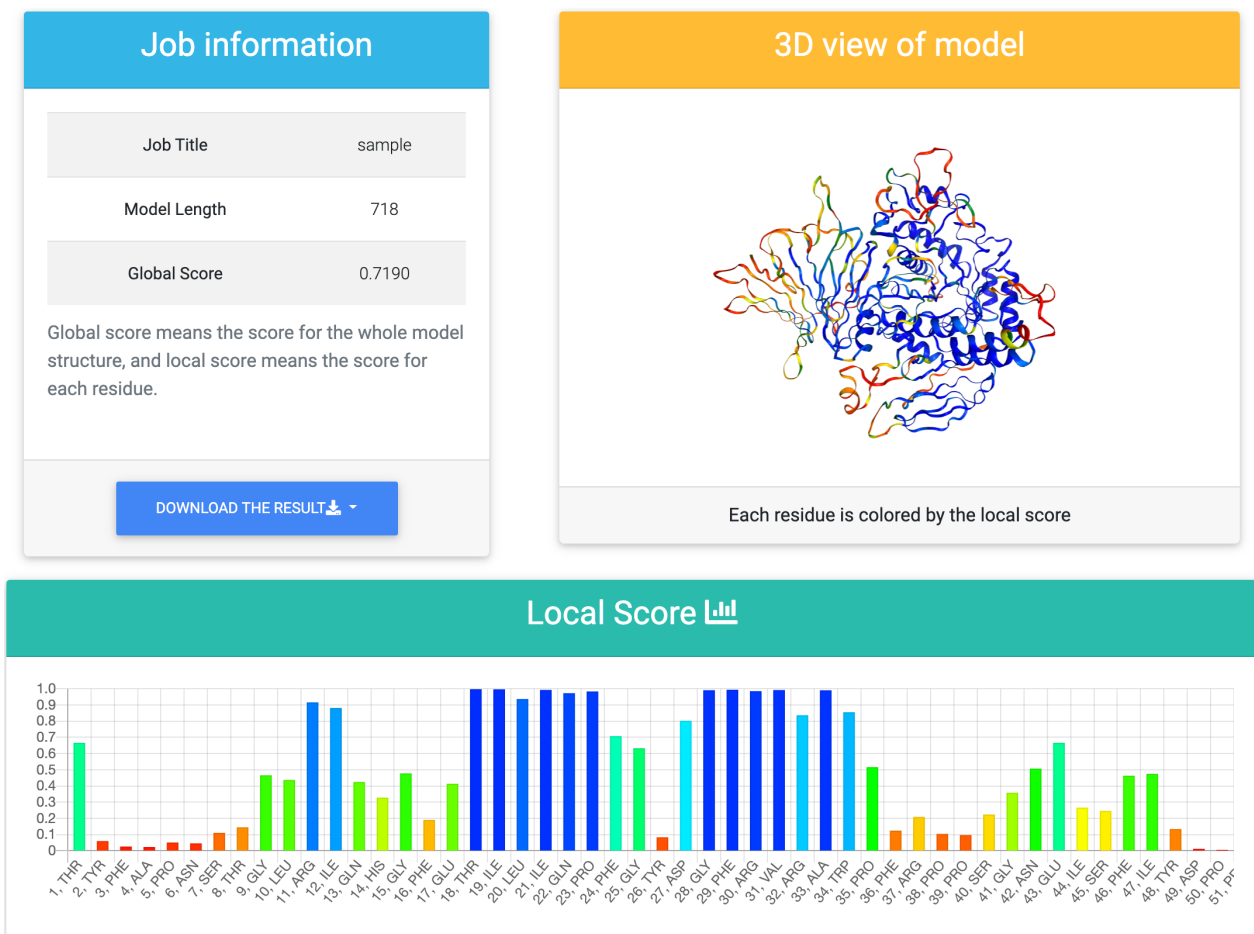
Figure 3. The output page of the prediction results. The predicted score for the whole model, the predicted score for each residue, and the three-dimensional structure colored by the local score is shown. The parts colored in blue represent high local scores, and the parts colored in red represent low local scores. The results can be downloaded in multiple formats.

5. Conclusions

We developed a single-model MQA method based on 3DCNN, called P3CMQA. It used sequence profile-based features and performed better quality estimation than that of existing single-model MQA methods. We have also developed a web tool of the proposed method, which provides prediction results in a user-friendly format.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

MQA	Model Quality Assessment
P3CMQA	Profile-based three-dimensional Convolutional neural network for protein structure Model Quality Assessment
GDT_TS	Global Distance Test Total Score
lDDT	Local Distance Diferrence Test
PSSM	Position Specific Scoring Matrix
SS	Secondary Structure
RSA	Relative Solvent Accessibility

Appendix A

Appendix A.1

14 atom-type features is shown in Table A1.

Table A1. 14 atom-type features.

Type	Description	Residue:Atom
1	Sulfur/selenium	CYS:SG, MET:SD, MSE:SE
2	Nitrogen (amide)	ASN:ND2, GLN:NE2, backbone N (including N-terminal)
3	Nitrogen (aromatic)	HIS:ND1/NE1, TRP:NE1
4	Nitrogen (guanidinium)	ARG:NE/NH*
5	Nitrogen (ammonium)	LYS:NZ
6	Oxygen (carbonyl)	ASN:OD1, GLN:OE1, backbone O (except C-terminal)
7	Oxygen (hydroxyl)	SER:OG, THR:OG1, TYR:OH
8	Oxygen (carboxyl)	ASP:OD*, GLU:OE*, C-terminal O, C-terminal OXTc
9	Carbon (sp2)	ARG:CZ, ASN:CG, ASP:CG, GLN:CD, GLU:CD, backbone C
10	Carbon (aromatic)	HIS:CG/CD2/CE1, PHE:CG/CD*/CE*/CZ, TRP:CG/CD*/CE*/CZ*/CH2, TYR:CG/CD*/CE*/CZ
11	Carbon (sp3)	ALA:CB, ARG:CB/CG/CD, ASN:CB, ASP:CB, CYS:CB, GLN:CB/CG, GLU:CB/CG, HIS:CB, ILE:CB/CG*/CD1, LEU:CB/CG/CD*, LYS:CB/CG/CD/CE, MET:CB/CG/CE, MSE:CB/CG/CE, PHE:CB, PRO:CB/CG/CD, SER:CB, THR:CB/CG2, TRP:CB, TYR:CB, VAL:CB/CG*, backbone CA
12	Occupancy	*:*
13	Backbone	*:N, *:CA, *:C
14	CA	*:CA

The first to the eleventh features are the features used by Derevyanko et al. [4] and the twelfth to the fourteenth features are the features added by Sato et al. [5]. An asterisk (*) for a residue represents all residues, and an asterisk for an atom represents either 1, 2, or 3. Note that the asterisk for a atom in the Occupancy represents all atoms.

Appendix A.2

The detail of the network architecture is shown in Table A2.

Table A2. Neural network architecture of Sato-3DCNN.

Layer Name	Output Shape	Detail
Input	$14 \times 28 \times 28 \times 28$	
Conv3D	$128 \times 25 \times 25 \times 25$	Batch Normalization, PReLU
Conv3D	$256 \times 22 \times 22 \times 22$	Batch Normalization, PReLU
Conv3D	$256 \times 11 \times 11 \times 11$	Batch Normalization, PReLU
Conv3D	$512 \times 8 \times 8 \times 8$	Batch Normalization, PReLU
Conv3D	$512 \times 6 \times 6 \times 6$	Batch Normalization, PReLU
Conv3D	$1024 \times 3 \times 3 \times 3$	Batch Normalization, PReLU
Global Average Pooling	1024	
Linear	1024	Batch Normalization, PReLU
Linear	256	Batch Normalization, PReLU
Linear	1	

The first column shows the layer name; the second column shows the output shape of the layer; and the third column shows the detailed information of the layer. The total number of parameters is about 22 million.

Appendix A.3

The results against multiple metrics for each combination of features in the validation dataset are shown in Table A3. In addition to the evaluation metrics in the test dataset, we also include the area under an ROC curve (AUC score) since we trained the model by binary classification. For all metrics, the performance was the best when all features were used.

Table A3. Performance comparison in the validation dataset by feature combinations.

Atom-Type Features	Evolutionary Information	Predicted Local Structure	Pearson	Spearman	Loss	Z-Score	AUC
✓	✗	✗	0.757	0.645	8.518	4.244	0.878
✓	✓	✗	0.834	0.729	9.860	4.239	0.923
✓	✗	✓	0.847	0.724	3.883	4.742	0.944
✗	✓	✓	0.858	0.742	4.818	4.666	0.948
✓	✓	✓	**0.865**	**0.751**	**2.519**	**4.866**	**0.956**

Columns 1–3 represent combinations of features referenced in Section 2.1. The fourth and fifth columns show the average Pearson and Spearman correlation coefficient for each target in the validation dataset. The sixth and seventh show the average GDT_TS loss and average Z-score of selected models for each target. The eighth column shows the area under an ROC curve (AUC) calculated from the predicted local scores and the local labels.

Appendix A.4

In the CASP dataset, targets are divided into categories according to their prediction difficulty. In the CASP13 dataset, there are six categories; FM, FM-sp, FM/TBM, TBM-hard, TBM-easy, not evaluated. The categories can be obtained from the official CASP page (https://predictioncenter.org/casp13/domains_summary.cgi, accessed on 9 March 2021.) Because of the many categories, we treat FM and FM-sp as FM and TBM-hard and TBM-easy as TBM. Besides, these categories are assigned to each domain, and some multi-domain targets belong to multiple categories. For such targets, we take the union of the categories. For example, a multi-domain target that belongs to FM, TBM, and not evaluated categories is treated as an FM/TBM category. Finally, the number of targets in each category was 12 for FM, 15 for FM/TBM, and 37 for TBM, 1 for not evaluated.

We show the average Pearson correlation coefficient for each category of targets on the CASP13 dataset in Table A4. The category "not evaluated" is excluded because there is only one target and no significant comparison can be performed.

Table A4. The average Pearson correlation coefficient for each category of targets on CASP13 dataset.

Method	FM (12 Targets)	FM/TBM (15 Targets)	TBM (37 Targets)
Proposed	**0.757** ($-$)	**0.812** ($-$)	**0.822** ($-$)
Sato-3DCNN(AMSGrad)	0.663 ($\mathbf{2.44 \times 10^{-3}}$)	0.730 ($\mathbf{4.27 \times 10^{-3}}$)	0.797 ($\mathbf{8.47 \times 10^{-3}}$)
ProQ3D	0.626 ($\mathbf{9.28 \times 10^{-3}}$)	0.689 ($\mathbf{3.05 \times 10^{-4}}$)	0.712 ($\mathbf{9.81 \times 10^{-7}}$)
SBROD	0.633 (1.61×10^{-2})	0.628 ($\mathbf{4.27 \times 10^{-4}}$)	0.720 ($\mathbf{6.23 \times 10^{-6}}$)
VoroMQA	0.579 ($\mathbf{1.46 \times 10^{-3}}$)	0.661 ($\mathbf{8.54 \times 10^{-4}}$)	0.724 ($\mathbf{2.32 \times 10^{-5}}$)

The first column represents the method name. The columns 2 show the average Pearson correlation coefficients for targets in category FM. Columns 3 and 4 similarly represent the values for targets in category FM/TBM and TBM, respectively. The values in parentheses are the *p*-values calculated by the Wilcoxon signed-rank test for the difference between the proposed method and the comparison method. The best values and *p*-values smaller than 0.01 are shown in bold.

The distribution of the Pearson correlation coefficient for each target on the CASP13 dataset is shown in Figure A1.
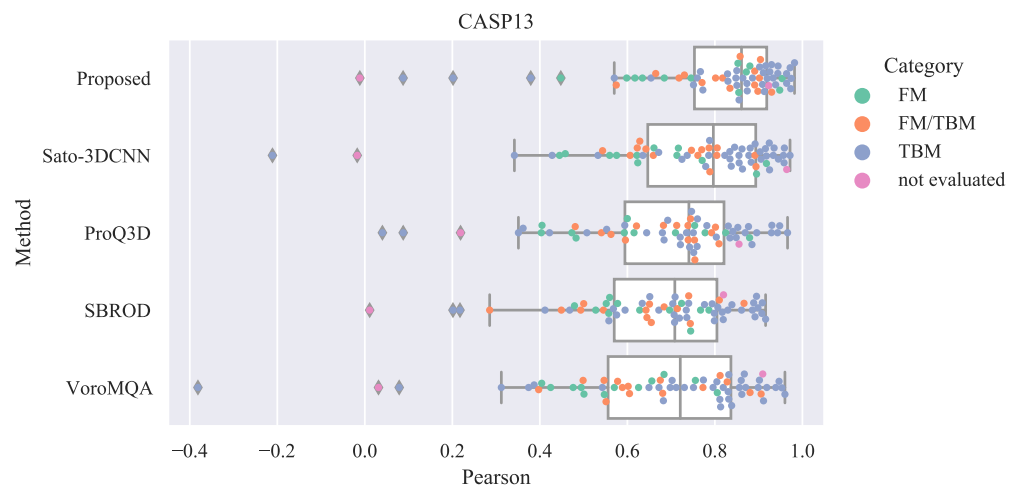

Figure A1. Swarm plot and box plot of the Pearson correlation coefficient for each target on CASP13. The x-axis represents the Pearson correlation coefficient, and the y-axis represents the method. A point represents a target, and the color of the point represents the category of the target.

References

1. Senior, A.W.; Evans, R.; Jumper, J.; Kirkpatrick, J.; Sifre, L.; Green, T.; Qin, C.; Žídek, A.; Nelson, A.W.; Bridgland, A.; et al. Protein structure prediction using multiple deep neural networks in the 13th Critical Assessment of Protein Structure Prediction (CASP13). *Proteins* **2019**, *87*, 1141–1148. [CrossRef] [PubMed]
2. Kryshtafovych, A.; Schwede, T.; Topf, M.; Fidelis, K.; Moult, J. Critical assessment of methods of protein structure prediction (CASP)—Round XIII. *Proteins Struct. Funct. Bioinform.* **2019**, *87*, 1011–1020. [CrossRef]
3. Hou, J.; Wu, T.; Cao, R.; Cheng, J. Protein tertiary structure modeling driven by deep learning and contact distance prediction in CASP13. *Proteins Struct. Funct. Bioinform.* **2019**, *87*, 1165–1178. [CrossRef] [PubMed]
4. Derevyanko, G.; Grudinin, S.; Bengio, Y.; Lamoureux, G. Deep convolutional networks for quality assessment of protein folds. *Bioinformatics* **2018**, *34*, 4046–4053. [CrossRef] [PubMed]
5. Sato, R.; Ishida, T. Protein model accuracy estimation based on local structure quality assessment using 3D convolutional neural network. *PLoS ONE* **2019**, *14*, e0221347. [CrossRef] [PubMed]
6. Ray, A.; Lindahl, E.; Wallner, B. Improved model quality assessment using ProQ2. *BMC Bioinform.* **2012**, *13*. [CrossRef]
7. Uziela, K.; Shu, N.; Wallner, B.; Elofsson, A. ProQ3: Improved model quality assessments using Rosetta energy terms. *Sci. Rep.* **2016**, *6*, 33509. [CrossRef]
8. Altschul, S.F.; Madden, T.L.; Schäffer, A.A.; Zhang, J.; Zhang, Z.; Miller, W.; Lipman, D.J. Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. *Nucleic Acids Res.* **1997**, *25*, 3389–3402. [CrossRef] [PubMed]

9.   Magnan, C.N.; Baldi, P. SSpro/ACCpro 5: Almost perfect prediction of protein secondary structure and relative solvent accessibility using profiles, machine learning and structural similarity. *Bioinformatics* **2014**, *30*, 2592–2597. [CrossRef]

10. Ioffe, S.; Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *Proceedings of the 32nd International Conference on Machine Learning, Lille, France, 7–9 July 2015*; Proceedings of Machine Learning Research; Bach, F., Blei, D., Eds.; PMLR: Lille, France, 2015; Volume 37, pp. 448–456.

11. He, K.; Zhang, X.; Ren, S.; Sun, J. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 1026–1034. [CrossRef]

12. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778. [CrossRef]

13. Mariani, V.; Biasini, M.; Barbato, A.; Schwede, T. IDDT: A local superposition-free score for comparing protein structures and models using distance difference tests. *Bioinformatics* **2013**, *29*, 2722–2728. [CrossRef]

14. Zemla, A. LGA: A method for finding 3D similarities in protein structures. *Nucleic Acids Res.* **2003**, *31*, 3370–3374. [CrossRef]

15. Reddi, S.J.; Kale, S.; Kumar, S. On the Convergence of Adam and Beyond. In Proceedings of the International Conference on Learning Representations, Vancouver, BC, Canada, 30 April–3 May 2018.

16. Moult, J.; Fidelis, K.; Kryshtafovych, A.; Rost, B.; Hubbard, T.; Tramontano, A. Critical assessment of methods of protein structure prediction—Round VII. *Proteins: Struct. Funct. Bioinform.* **2007**, *69*, 3–9. [CrossRef]

17. Moult, J.; Fidelis, K.; Kryshtafovych, A.; Rost, B.; Tramontano, A. Critical assessment of methods of protein structure prediction-Round VIII. *Proteins Struct. Funct. Bioinform.* **2009**, *77*, 1–4. [CrossRef]

18. Moult, J.; Fidelis, K.; Kryshtafovych, A.; Tramontano, A. Critical assessment of methods of protein structure prediction (CASP)—round IX. *Proteins Struct. Funct. Bioinform.* **2011**, *79*, 1–5. [CrossRef]

19. Moult, J.; Fidelis, K.; Kryshtafovych, A.; Schwede, T.; Tramontano, A.; Topf, M.; Fidelis, K.; Moult, J.; Fidelis, K.; Kryshtafovych, A.; et al. Critical assessment of methods of protein structure prediction (CASP)—round x. *Proteins Struct. Funct. Bioinform.* **2014**, *82*, 1–6. [CrossRef] [PubMed]

20. Moult, J.; Fidelis, K.; Kryshtafovych, A.; Schwede, T.; Tramontano, A.; Topf, M.; Fidelis, K.; Moult, J.; Fidelis, K.; Kryshtafovych, A.; et al. Critical assessment of methods of protein structure prediction (CASP)—Round XII. *Proteins Struct. Funct. Bioinform.* **2018**, *86*, 7–15. [CrossRef] [PubMed]

21. Krivov, G.G.; Shapovalov, M.V.; Dunbrack, R.L. Improved prediction of protein side-chain conformations with SCWRL4. *Proteins Struct. Funct. Bioinform.* **2009**, *77*, 778–795. [CrossRef]

22. Uziela, K.; Hurtado, D.M.; Shu, N.; Wallner, B.; Elofsson, A. ProQ3D: Improved model quality assessments using deep learning. *Bioinformatics* **2017**, *33*, 1578–1580. [CrossRef] [PubMed]

23. Karasikov, M.; Pagès, G.; Grudinin, S. Smooth orientation-dependent scoring function for coarse-grained protein quality assessment. *Bioinformatics* **2019**, *35*, 2801–2808. [CrossRef]

24. Olechnovič, K.; Venclovas, Č. VoroMQA: Assessment of protein structure quality using interatomic contact areas. *Proteins Struct. Funct. Bioinform.* **2017**, *85*, 1131–1145. [CrossRef]

25. Funk, S. RMSprop loses to SMORMS3—9Beware the Epsilon! 2015. Available online: https://sifter.org/~simon/journal/20150420.html (accessed on 25 February 2021).

26. Rose, A.S.; Bradley, A.R.; Valasatava, Y.; Duarte, J.M.; Prlić, A.; Rose, P.W. NGL viewer: Web-based molecular graphics for large complexes. *Bioinformatics* **2018**, *34*, 3755–3758. [CrossRef] [PubMed]