DeepQA: improving the estimation of single protein model quality with deep belief networks

Renzhi Cao[1], Debswapna Bhattacharya[2], Jie Hou[3] and Jianlin Cheng[3,4*]

Abstract

Background: Protein quality assessment (QA) useful for ranking and selecting protein models has long been viewed as one of the major challenges for protein tertiary structure prediction. Especially, estimating the quality of a single protein model, which is important for selecting a few good models out of a large model pool consisting of mostly low-quality models, is still a largely unsolved problem.

Results: We introduce a novel single-model quality assessment method DeepQA based on deep belief network that utilizes a number of selected features describing the quality of a model from different perspectives, such as energy, physio-chemical characteristics, and structural information. The deep belief network is trained on several large datasets consisting of models from the Critical Assessment of Protein Structure Prediction (CASP) experiments, several publicly available datasets, and models generated by our in-house *ab initio* method. Our experiments demonstrate that deep belief network has better performance compared to Support Vector Machines and Neural Networks on the protein model quality assessment problem, and our method DeepQA achieves the state-of-the-art performance on CASP11 dataset. It also outperformed two well-established methods in selecting good outlier models from a large set of models of mostly low quality generated by *ab initio* modeling methods.

Conclusion: DeepQA is a useful deep learning tool for protein single model quality assessment and protein structure prediction. The source code, executable, document and training/test datasets of DeepQA for Linux is freely available to non-commercial users at http://cactus.rnet.missouri.edu/DeepQA/.

Keywords: Protein model quality assessment, Protein structure prediction, Machine learning, Deep belief network

Background

The tertiary structures of proteins are important for understanding their functions, and have a lot of biomedical applications, such as the drug discovery [1]. With the wide application of next generation sequencing technologies, millions of protein sequences have been generated, which create a huge gap between the number of protein sequences and the number of protein structures [2, 3]. The computational structure prediction methods have the potential to fill the gap, since it is much faster and cheaper than experimental techniques, and also can

be used for proteins whose structures are hard to be determined by experimental techniques, such as X-ray crystallography [1].

There are generally two major challenges in protein structure prediction [4]. The first challenge is how to sample the protein structural model from the protein sequences, the so-called structure sampling problem. Two different kinds of methods have been used to do the model sampling. The first is template-based modeling method [5–11] which uses the known structure information of homologous proteins as templates to build protein structure model, such as I-TASSER [12], FALCON [10, 11], MUFOLD [13], RaptorX [14], and MTMG [15]. The second is *ab initio* modeling method [16–21], which builds the structure from scratch, without using existing template structure information. The second challenge is

* Correspondence: chengji@missouri.edu
[3]Department of Computer Science, University of Missouri, Columbia, MO 65211, USA
[4]Informatics Institute, University of Missouri, Columbia, MO 65211, USA
Full list of author information is available at the end of the article

how to select good models from generated models pool, the so-called model ranking problem. It is essential for protein structure prediction, such as selecting models generated by *ab initio* modeling methods. There are mainly two different types of methods for the model ranking. The first is consensus methods [22–25], which calculate the average structural similarity score of a model against other models as its model quality, such as Modfoldclust2 [24] which compares 3D models of proteins by the Q measure. This method assumes the models in a model pool that are more similar to other models have better quality. It shows good performance in previous Critical Assessment of Techniques for Protein Structure Prediction (CASP) experiments [26] (during previous CASP, the consensus QA methods that evaluate protein model quality assessment by pairwise comparison usually performs better than single-model QA methods that evaluate protein model's quality without using other model's information), which is a world-wide experiment for blindly testing protein structure prediction methods every 2 year. However, the accuracy of this method depends on input data, such as the proportion of good models in a model pool and the similarity between low quality models. It has been shown that this kind of method is not working well when a large portion of models are of low quality [27]. The time complexity of most consensus methods is $O(n^2)$ time complexity (n: the total number of models), making it too slow to assess the quality of a large number of models. These problems with consensus methods highlight the importance of developing another kind of protein model quality assessment (QA) method - single-model QA method [5, 18, 27–33] that predicts the model quality based on the information from a single model itself. Single-model quality assessment methods only require the information of a single model as input, and therefore its performance does not depend on the distribution of high and low quality models in a model pool. In this paper, we focus on develop a new single-model quality assessment method that uses deep learning in conjunction with a number of useful features relevant to protein model quality.

Currently, most single-model QA methods predict model quality from sequence evolutionary information [34], residue environment compatibility [35], structural features and physics-based knowledge [29–32, 36–39]. One such single-model QA method - ProQ2 [40] has relatively good performance in the CASP11 experiment, which uses Support Vector Machines with a number of features from a model and its sequence to predict its quality. ProQ3 [41] is updated version of ProQ2 by exchanging features with energy terms calculated from Rosetta and shows superior performance over ProQ2. Another single-model quality assessment method

- RFMQA [39] applies Random Forest on structural features and knowledge-based potential energy terms, which achieves good performance on CASP10 targets. In addition, ResQ [42] is a new protein model quality assessment method for estimating B-factor and residue-level quality in protein structure prediction, based on local variations of modelling simulations and the uncertainty of homologous alignments.

Here, we propose to develop a novel single-model quality assessment method based on deep belief network - a kind of deep learning methods that show a lot of promises in image processing [43–45] and bioinformatics [46]. We benchmark the performance of this method on large QA datasets, including the CASP datasets, four datasets from the recently 3DRobot decoys [47], and a dataset generated by our in-house *ab initio* modeling method UniCon3D. The good performance of our method - DeepQA on these datasets demonstrate the potential of applying deep learning techniques for protein model quality assessment.

The paper is organized as follows. In the Methods Section, we describe the datasets and features that are used for deep learning method, and how we implement, train, and evaluate the performance of our method. In the Result Section, we compare the performance of deep learning technique with two other QA methods based on support vector machines and neural networks. In the Results and Discussion Section, we summarize the results. In the Conclusion Section, we conclude the paper with our findings and future works.

Methods
Datasets
We collect three previous CASP models (CASP8, CASP9, and CASP10) from the CASP website http://prediction-center.org/download_area/, 3DRobot decoys [47], and 3113 native protein structure from PISCES database [48] as the training datasets. We use CASP11 models that were not used in training as testing dataset, and UniCon3D *ab initio* CASP11 decoys as the validation datasets.

The 3DRobot decoys have four sets: 200 non-homologous (48 α, 40 β, and 112 α/β) single domain proteins each having 300 structural decoys; 58 proteins used in a Rosetta benchmark [49] each having 100 structural decoys; 20 proteins in a Modeller benchmark [50] each having 200 structural decoys; and 56 proteins in a I-TASSER benchmark each having 400 structural decoys. Two sets (stage1 and stage2) of CASP11 targets are used to test the performance of DeepQA. Each target at stage one contains 20 server models spanning the whole range of structural quality and each target at stage two contains 150 top server models selected by Davis-QAconsensus method. In total, 803 proteins with 216,875 structural decoys covering wide range of qualities are collected for

training and testing DeepQA. All of these data and calculated quality scores are available at: http://cactus.rnet.missouri.edu/DeepQA/. The quality score of a model is the GDT-TS score [51] in the range [0, 1] that measures the similarity between the model and its corresponding native structure. The LGA package [52] is used to calculate GDT-TS score and the official CASP website is used to download models and native structure based on domains. In addition, we validate performance of our QA methods in a dataset produced by our *ab initio* modeling tool Uni-Con3D, which in total includes 24 targets and 20,030 models. The average of first ranked GDT_TS scores (GDT_TS1) for 84 models of Stage one and Stage two is 0.54 and 0.58 respectively. For the *ab initio* dataset, the average of first ranked GDT_TS score is 0.20.

Input features for DeepQA

In total, 16 features are used as input for our method DeepQA, which describe the structural, physio-chemical and energy properties of a protein model. These features include nine available top-performing energy and knowledge-based potentials scores, including ModelEvaluator score [31], Dope score [32], RWplus score [30], RF_CB_SRS_OD score [29], Qprob scores [33], GOAP score [53], OPUS score [54], ProQ2 score [40], DFIRE2 score [55]. All of these scores are converted into the range of zero and one as the input features for training the deep leaning networks. Occasionally, if a feature cannot be calculated for a model due to the failure of a tool, its value is set to 0.5.

The remaining seven input features are generated from the physio-chemical properties of a protein model. These features are calculated from a structural model and its protein sequence [37], which include: secondary structure similarity (SS) score, solvent accessibility similarity (SA) score, secondary structure penalty (SP) score, Euclidean compact (EC) score, Surface (SU) score, exposed mass (EM) score, exposed surface (ES) score. All of these 16 scores are converted into the range between zero and one for training the deep learning networks, and the following formula is used for normalizing DFIRE2, RWplus, and RF_CB_SRS_OD scores:

$$
\begin{cases}
Norm_{S_{Dfire}} & = \dfrac{-P_{Dfire\ score}}{1{,}971 * L} \\[2ex]
Norm_{S_{RWplus}} & = \dfrac{-P_{RWplus\ score}}{232.6 * L} \\[2ex]
Norm_{S_{RF_C B_S RS_O D}} & = \dfrac{700 - P_{RF_C B_S RS_O D\ score}}{1000 + 0.4823 * L}
\end{cases}
$$

L is the sequence length, $P_{Dfire\ score}$ is the predicted DFIRE2 score, $P_{RWplus\ score}$ is the predicted RWplus score, and $P_{RF\_CB\_SRS\_OD\ score}$ is the predicted RF_CB_SRS_OD score. The score is set to zero when the calculated result is less than zero, and one when the calculated result is larger than one. Occasionally, if a feature cannot be calculated for a model due to the failure of a tool, its value is set to 0.5.

A summary table of all features and their descriptions is given in Table 1.

Deep belief network architectures and training procedure

Our in-house deep belief network framework [46] is used to train deep learning models for protein model

Table 1 16 features for benchmarking DeepQA

Feature Name	Feature descriptions
(1). Surface score (SU)	The total area of exposed nonpolar residues divided byc the total area of all residues
(2). Exposed mass score (EM)	The percentage of mass for exposed area, equal to the total mass of exposed area divided by the total mass of all area
(3). Exposed surface score (ES)	The total exposed area divided by the total area
(4). Solvent accessibility score (SA)	The difference of solvent accessibility predicted by SSpro4 [1] from the protein sequence and those of a model parsed by DSSP [2]
(5). RF_CB_SRS_OD score [3]	A novel distance dependent residue-level potential energy score.
(6). DFIRE2 score [4]	A distance-scaled all atom energy score.
(7). Dope score [5]	A new statistical potential discrete optimized protein energy score.
(8). GOAP score [6]	A generalized orientation-dependent, all-atom statistical potential score.
(9). OPUS score [7]	A knowledge-based potential score.
(10). ProQ2 score [8]	A single-model quality assessment method by machine learning techniques.
(11). RWplus score [9]	A new energy score using pairwise distance-dependent atomic statistical potential function and side-chain orientation-dependent energy term
(12). ModelEvaluator score [10]	A single-model quality assessment score based on structural features using support vector machine.
(13). Secondary structure similarity score (SS)	The difference of secondary structure information predicted by Spine X [11] from a protein sequence and those of a model parsed by DSSP [2]
(14). Secondary structure penalty score (SP)	Calculated from the predicted secondary structure alpha-helix and beta-sheet matching with the one parsed by DSSP.
(15). Euclidean compact score (EC)	The pairwise Euclidean distance of all residues divided by the maximum Euclidean distance (3.8) of all residues.
(16). Qprob [12]	A single-model quality assessment score that utilizes 11 structural and physicochemical features by feature-based probability density functions.

quality assessment. As is shown in Fig. 1, in this framework, a two-layer Restricted Boltzmann Machines (RBMs) form the hidden layers of the deep learning networks, and one layer of logistic regression node is added at the top to output a real value between 0 and 1 as predicted quality score. The weights of RBMs are initialized by unsupervised learning called pre-training. The pre-train process is carried out by the 'contrastive divergence' algorithm to adjust the weight in the RBM networks [56]. The mean square error is considered as cost function in the process of standard error backward propagation. The final deep belief architecture is fine-tuned and optimized based on Broyden-Fletcher-Goldfarh-Shanno(BFGS) optimization [57]. We divide the training data equally into five sets, and a five-fold cross validation is used to train and validate DeepQA. Five parameters of DeepQA are adjusted during the training procedure. The five parameters are total number of nodes at the first hidden layer (N1), total number of nodes at the second hidden layer (N2), learning rate ε (default 0.001), weight cost ω (default 0.07), and momentum ν (default from 0.5 to 0.9). The last three parameters are used for training the RBMs. The average of Mean Absolute Error (MAE) is calculated for each round of five-fold cross validation to estimate the model accuracy. MAE is the absolute difference of predicted value and real value.

Model accuracy evaluation metrics

We evaluate the accuracy of DeepQA on 84 protein targets on both stage one and stage two models of the 11th community-wide experiment on the Critical Assessment of Techniques for Protein Structure Prediction (CASP11), which are available in the CASP official website (http://www.predictioncenter.org/casp11/index.cgi).

The real GDT-TS score of each protein model is calculated against the native structure by TM-score [51]. Second, all feature scores are calculated for each protein model. The trained DeepQA is used to predict the quality score of a model based on its input feature scores.

To evaluate the performance of QA method, we use the following metrics: average per-target loss which is the difference of GDT-TS score of the top one model selected by a QA method and that of the best model in the model pool, average per-target correlation which is the Pearson's correlation between all models' real GDT-TS scores and its predicted scores, the summation of real TM-score and RMSD scores of the top models selected by a QA method, and the summation of real TM-score and RMSD scores of the best of top five models selected by QA methods.

To evaluate the performance of QA methods on *ab initio* models, we calculated the average per-target TM-score and RMSD for the selected top one model, and also for the best of selected top five models by QA methods.

Results and discussion

Comparison of Deep learning with support vector machines and neural networks

We train the deep learning and two other widely used machine learning techniques (Support Vector Machine and Neural Network) separately on our training datasets and compare their performance using five-fold cross-validation protocol. SVMlight [7] is used to train the support vector machine, and the tool Weka [58] is used to train the neural networks. The RBF
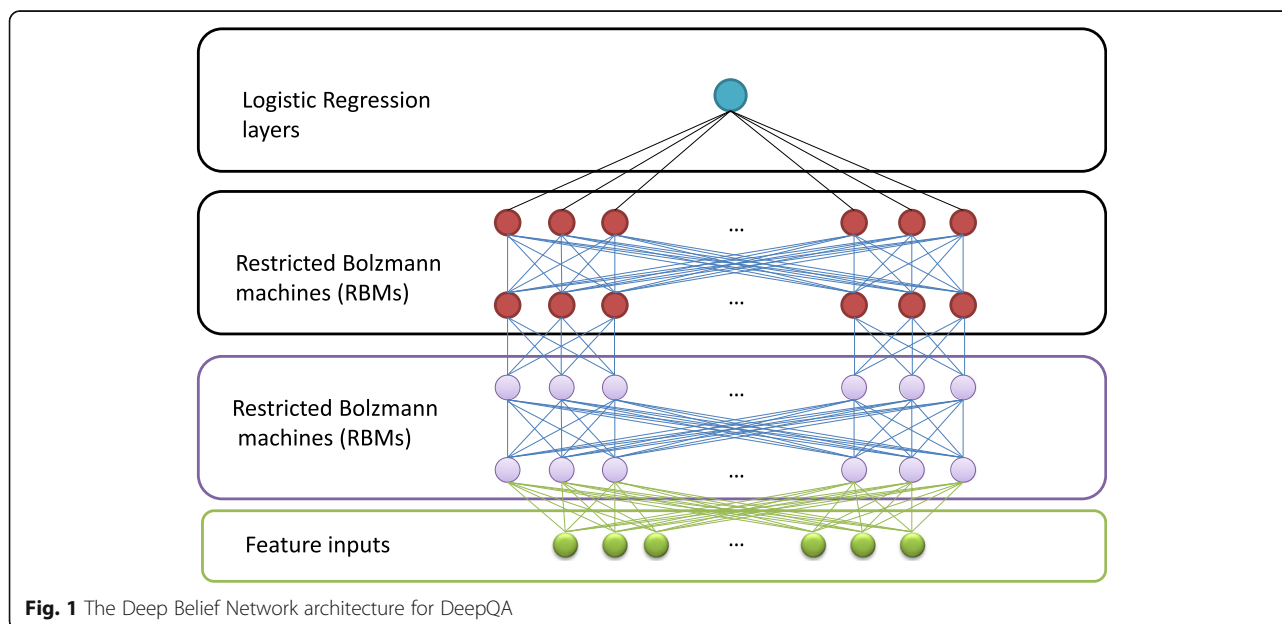

Fig. 1 The Deep Belief Network architecture for DeepQA

kernel function is used for support vector machine, and the following three parameters are adjusted: C for the trade-off between training error and margin, ε for the epsilon width of tube for regression, and parameter gamma for RBF kernel. We randomly select 7, 500 data points from the whole datasets to form a small dataset to estimate these parameters of support vector machine to speed up the training process. Based on the cross validation result on this selected small dataset, C is set to 60, ε to 0.19, gamma to 0.95. For the neural network, we adjust the following three parameters: the number of hidden nodes in the first layer (from 5 to 40), the number of hidden nodes in the second layer (from 5 to 40), and the learning rate (from 0.01 to 0.4). Based on the cross validation result on the entire datasets, we set the number of hidden nodes as 40 and 30 for the first and second layer respectively, and the learning rate is set to be 0.3. For the deep belief network, we test the number of hidden nodes in the first and second layer of RBMs from 5 to 40 respectively, learning rate ε from 0.0001 to 0.01, weight cost ω from 0.001 to 0.7, and momentum ν from 0.5 to 0.9. Based on the MAE of cross validation result, we find the following parameters with good performance: the number of hidden nodes in the first and second layer of RBMs is set to 20 and 10 respectively, learning rate to 0.0001, weight cost to 0.007, and momentum from 0.5 to 0.9. After these three machine learning methods are trained, they are evaluated on the test datasets.

The correlation and loss on both stage one and stage two models of CASP11 datasets are calculated for these three methods, and the results are shown in Table 2. Deep belief network has the best average per-target correlation on both stage one and stage two. The loss of DeepQA is also lower than or equal to the other two methods. The result of Wilcoxon signed ranked sum test between deep belief network and other two methods is also added in Table 2. The results suggest that deep belief network is a good choice for protein quality assessment problem.

Comparison of DeepQA with other single-model QA methods on CASP11

In order to reduce the model complexity and improve accuracy, we do a further analysis by selecting good features out of all these 16 features for our method DeepQA. First of all, we fix a set of parameters with good performance on all 16 features (e.g., the number of nodes in the first and second hidden layer is set to 20 and 10 respectively), and then train the Deep Belief Network for different combination of all these 16 features. Based on the MAE of these models in the training datasets, we use the following features which has relatively good performance and also low model complexity as the final features of DeepQA: Surface score, Dope score, GOAP score, OPUS score, RWplus score, Modelevaluator score, Secondary structure penalty score, Euclidean compact score, and Qprob score. After DeepQA with these sub set of features is trained on the training data, it is blindly tested on the test datasets.

We evaluate the DeepQA on CASP11 datasets, and compare it with other single-model QA methods participating in CASP11. We use the standard evaluation metrics - average per-target correlation and average per-target loss based on GDT-TS score to evaluate the performance of each method (see the results in Table 3). On stage one of CASP11, the average per-target correlation of DeepQA is 0.64, which is the same as the ProQ2 - the top single-model quality assessment method in the CASP11 experiment - and better than Qprob. The average per-target loss of DeepQA is 0.09, same as ProQ2 and ProQ2-refine, and better than other single-model QA methods. On stage two models of CASP11, DeepQA has the highest per-target average correlation. Its per-target average loss is the same as ProQ2, and better than all other QA methods. The result of Wilcoxon signed ranked sum test between DeepQA and other methods is also added in Table 3. Overall, the results demonstrate that DeepQA has achieved the state-of-the-art performance.

In order to evaluate how DeepQA aids the protein tertiary structure prediction methods in model selection,

Table 2 The accuracy of Deep Belief Network, Support Vector Machines, and Neural Networks in terms of Mean Absolute Error (MAE) based on cross validation of training datasets with 16 features, the average per-target correlation, and loss on stage 1 and stage 2 of CASP11 datasets for all three difference techniques. *P*-value is calculated for the significance of DBN compared to other two methods

	MAE based on cross validation	Corr. on stage 1/ significance of *P*-value	Loss on stage 1/ significance of *P*-value	Corr. on stage 2/ significance of *P*-value	Loss on stage 2/ significance of *P*-value
Deep Belief Network	0.08	0.63/-	0.09/-	0.34/-	0.06/-
Support Vector Machine	0.12	0.58/1.97E-01	0.10/6.17E-01	0.32/4.45E-04	0.07/7.41E-01
Neural Network	0.08	0.51/9.74E-04	0.12/8.35E-02	0.25/1.05E-05	0.07/1.19E-01
Mean	0.09	0.57/9.88E-02	0.10/3.50E-01	0.30/2.28E-04	0.07/4.30E-01

Table 3 Average per-target correlation and loss for DeepQA and other top performing single-model QA methods on CASP11. The table is ranked based on the average per-target loss on stage two of CASP11. *P*-value of Wilcoxon signed ranked sum test* between DeepQA and other methods is also included in the table

QA methods	Corr. on stage 1 /*P*-Value	Loss on stage 1 /*P*-Value	Corr. on stage 2 /*P*-Value	Loss on stage 2 /*P*-Value
DeepQA	0.64/-	0.09/-	0.42/-	0.06/-
ProQ2	0.64/4.80E-01	0.09/8.32E-01	0.37/2.84E-03	0.06/9.95E-01
Qprob	0.63/8.08E-01	0.10/9.38E-01	0.38/8.63E-03	0.07/7.12E-01
VoroMQA	0.56/1.60E-04	0.11/2.73E-01	0.40/2.57E-01	0.07/9.14E-01
ProQ2-refine	0.65/6.08E-02	0.09/9.17E-01	0.37/4.71E-03	0.07/4.86E-01
Wang_SVM	0.66/5.49E-02	0.11/7.98E-02	0.36/1.54E-02	0.09/4.91E-02
raghavagps-qaspro	0.35/3.79E-13	0.16/1.87E-04	0.22/1.92E-10	0.09/1.02E-03
Wang_deep_2	0.63/9.98E-01	0.12/7.18E-02	0.31/2.16E-06	0.09/8.22E-03
Wang_deep_1	0.61/3.06E-01	0.13/1.64E-03	0.30/5.93E-06	0.09/5.00E-03
Wang_deep_3	0.63/7.18E-02	0.12/3.15E-02	0.30/8.22E-03	0.09/8.22E-03
FUSION	0.10/8.43E-14	0.15/9.78E-04	0.05/1.81E-13	0.11/2.83E-07
RFMQA	0.54/1.61E-01	0.12/8.74E-01	0.29/3.80E-03	0.08/3.80E-03
ProQ3	0.65/1.62E-01	0.07/3.60E-02	0.38/4.44E-01	0.06/4.09E-01
ResQ*	0.67/-	0.05/-	0.58/-	0.09/-
ModFOLDclust2	0.74/3.96E-05	0.05/6.34E-04	0.56/1.80E-03	0.07/1.41E-01
Mean	0.57	0.11	0.33	0.08

* The Wilcoxon signed ranked sum test was performed on the correlation and loss of targets between each method against DeepQA
* ResQ was evaluated on 54 targets in CASP11, the local quality scores were converted into global quality score by equation $Global = \frac{1}{L}\sum_{i=1}^{L}\frac{1}{1+\left(\frac{Local_i}{5}\right)^2}$. More detailed results can be found in Additional file 1: Table S4

we apply DeepQA to select models in the stage two dataset of CASP11 submitted by top performing protein tertiary structure prediction methods. For most cases, DeepQA helps the protein tertiary structure prediction methods to improve the quality of the top selected model. For example, DeepQA improves overall Z-score for Zhang-Server by 6.39, BAKER-ROSETTASERVER by 16.34, and RaptorX by 6.66. The result of applying DeepQA on 10 top performing protein tertiary structure prediction methods is shown at Additional file 1: Table S1.

Case study of DeepQA on *ab initio* datasets

In order to assess the ability of DeepQA in evaluating *ab initio* models, we evaluate it on 24 *ab initio* targets with more than 20,000 models generated by UniCon3D. Table 4 shows the average per-target TM-score and RMSD for the top one model and best of top 5 models

selected by DeepQA, ProQ2, and two energy scores (i.e., Dope and RWplus), respectively. The result shows DeepQA achieves good performance in terms of TM-score and RMSD compared with ProQ2 and two top-performing energy scores. The TM-score difference of best of top 5 models between DeepQA and ProQ2 is significant. In most cases, Z-score is also widely used to highlight the significance of QA methods for model selection. The summation of Z-score based on TM-score and RMSD for each QA method is also included in Table 4. The results demonstrate that DeepQA achieves the best performance compared to other methods based on Z-score. Additional files 2 and 3: Tables S2 and S3 show the per-target TM-score and RMSD of DeepQA and ProQ2 on this *ab initio* datasets, along with Z-score of top 1 model and best of top 5 models for DeepQA.

Table 4 Model selection ability on *ab initio* datasets for DeepQA, ProQ2, Dope2, and RWplus score based on TM-score and RMSD, and their summation of Z-score

QA methods	TM-score on top 1 model/ SUM Z-score (>0.0)	RMSD on top 1 model/ SUM Z-score (<0.0)	TM-score on best of top 5/SUM Z-score (>0.0)	RMSD on best of top 5/SUM Z-score (<0.0)
DeepQA	0.23/0.86	19.01/-0.76	0.26/1.78	17.14/-1.52
ProQ2	0.22/0.40	19.73/-0.37	0.25/1.28	17.93/-1.04
Dope	0.22/0.49	19.55/-0.51	0.24/1.13	18.10/-1.00
RWplus	0.22/0.53	19.68/-0.35	0.25/1.49	17.38/-1.41
Mean	0.22/0.68	19.49/-0.64	0.25/1.46	17.64/-1.26

Table 5 Average per-target correlation and loss on Stage 1 and Stage 2 for DeepQA and its training features on CASP11. The significance between DeepQA and individual feature was assessed by Wilcoxon signed ranked sum paired *t*-test*, and its *P*-value was included to represent the improvement of DeepQA against its input features

QA methods	Corr. onstage 1/P-value	Loss on stage 1/P-value	Corr. on stage 2/P-value	Loss on stage 2/P-value
DeepQA	0.64/-	0.09/-	0.42/-	0.06/-
Dope	0.54/1.77E-06	0.11/0.0421	0.30/4.63E-10	0.08/2.76E-01
EC score	0.37/4.29E-11	0.18/5.71E-07	0.02/3.23E-14	0.14/2.08E-10
GOAP score	0.54/2.74E-05	0.13/0.0016	0.31/5.07E-07	0.07/1.06E-01
ModelEvaluator score	0.56/0.0001	0.10/0.2160	0.28/1.87E-09	0.08/1.99E-02
OPUS score	0.43/2.14E-11	0.12/0.0588	0.30/4.53E-09	0.08/3.54E-01
Qprob score	0.63/0.8080	0.09/0.9382	0.38/8.63E-03	0.06/7.12E-01
RWplus score	0.54/4.80E-06	0.14/0.0009	0.30/9.41E-09	0.08/4.49E-02
SP score	0.47/3.07E-10	0.14/0.0067	0.26/6.17E-10	0.10/1.10E-05
SU score	0.50/3.78E-09	0.18/4.94E-07	0.19/6.34E-11	0.11/3.95E-07
Mean	0.52/0.09	0.13/0.14	0.27/0.00	0.09/0.17

* The Wilcoxon signed ranked sum paired *t*-test was performed on the correlation and loss of targets between each feature against DeepQA

Comparison of DeepQA with individual features on CASP11

In order to examine the improvement that DeepQA achieved by integrating multiple features for protein quality assessment, specifically, the improvement of DeepQA compared against its nine input training features, we performed Wilcoxon signed ranked sum test on per-target correlation and loss metrics between each input feature and DeepQA predictions. The correlation, loss and significance on Stage one and Stage two for DeepQA and nine input training features are shown in Table 5. In Table 5, DeepQA achieves best correlation on Stage1 against all other nine features, and *P*-value of statistical analysis between DeepQA and most features (except Qprob) is less than 0.05. However, *P*-value of statistical analysis on Stage two in Table 5 is less than 0.05 for DeepQA against all nine input features. For the loss metric, DeepQA achieves the best performance against all nine input features, but *P*-value of statistical analysis shows that the improvement is not always significant. In summary, we compared the performance of DeepQA with all nine input features, and the result shows improvement based on both correlation and loss on CASP11 datasets. In addition, the significant improvement of DeepQA on correlation metric compared with most input features (except Qprob) has been achieved according to the statistical analysis of Wilcoxon signed ranked sum test, and the improvement of DeepQA on loss metric is not significant compared with most input features, especially on Stage two of CASP11 datasets.

Conclusions

In this paper, we develop a new single-model QA method (DeepQA) based on deep belief network. It performs better than support vector machines and neural networks, and achieve the state-of-the-art performance in comparison with other established QA methods. DeepQA is also useful for ranking *ab initio* protein models. And DeepQA could be further improved by incorporating more relevant features and training on larger datasets.

Additional file

Additional file 1: Table S1. Z-score improvement of applying DeepQA for CASP11 top performance protein tertiary structure prediction methods. **Table S2.** TM-score and RMSD score (and their Z-score) of DeepQA on *ab initio* datasets. **Table S3.** TM-score and RMSD score of ProQ2 on *ab initio* datasets. **Table S4.** Average per-target correlation and loss for DeepQA and ResQ on 54 targets of CASP11. (DOCX 34 kb)

Authors' contributions
JC and RC conceived and designed the project. RC, DB, JH implemented and tested the tool. RC, DB, JH, and JC wrote the manuscript. All the authors read and approved the manuscript.

Competing interests
The authors declare that they have no competing interests.

Consent for publication
Not applicable.

Ethics approval and consent to participate
Not applicable.

Author details
[1]Department of Computer Science, Pacific Lutheran University, Tacoma, WA 98447, USA. [2]Department of Electrical Engineering and Computer Science, Wichita State University, Wichita, KS 67260, USA. [3]Department of Computer Science, University of Missouri, Columbia, MO 65211, USA. [4]Informatics Institute, University of Missouri, Columbia, MO 65211, USA.

References
1. Jacobson M, Sali A. Comparative protein structure modeling and its applications to drug discovery. Annu Rep Med Chem. 2004;39(85):259–74.
2. Li J, Cao R, Cheng J. A large-scale conformation sampling and evaluation server for protein tertiary structure prediction and its assessment in CASP11. BMC Bioinf. 2015;16(1):337.
3. Cao R, Cheng J. Integrated protein function prediction by mining function associations, sequences, and protein–protein and gene–gene interaction networks. Methods. 2016;93:84–91.
4. Cao R, Bhattacharya D, Adhikari B, Li J, Cheng J. Large-scale model quality assessment for improving protein tertiary structure prediction. Bioinformatics. 2015;31(12):i116–23.
5. Cao R, Jo T, Cheng J. Evaluation of protein structural models using random forests. 2016. arXiv preprint arXiv:160204277.
6. Li J, Bhattacharya D, Cao R, Adhikari B, Deng X, Eickholt J, Cheng J. The MULTICOM protein tertiary structure prediction system. Protein Struct Prediction. 2014;1137:29–41.
7. Joachims T. Optimizing search engines using clickthrough data. In: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM; 2002. p. 133–42. (KDD '02). http://dx.doi.org/10.1145/775047.775067.
8. Simons KT, Kooperberg C, Huang E, Baker D. Assembly of protein tertiary structures from fragments with similar local sequences using simulated annealing and Bayesian scoring functions. J Mol Biol. 1997;268(1):209–25.
9. Page R. TreeView: an application to display phylogenetic trees on personal computer. Comp Appl Biol Sci. 1996;12:357–8.
10. Wang C, Zhang H, Zheng W-M, Xu D, Zhu J, Wang B, Ning K, Sun S, Li SC, Bu D. FALCON@ home: a high-throughput protein structure prediction server based on remote homologue recognition. Bioinformatics. 2016;32(3):462–4.
11. Li SC, Bu D, Xu J, Li M. Fragment-HMM: A new approach to protein structure prediction. Protein Sci. 2008;17(11):1925–34.
12. Zhang Y. I-TASSER server for protein 3D structure prediction. BMC Bioinf. 2008;9(1):40.
13. Zhang J, Wang Q, Barz B, He Z, Kosztin I, Shang Y, Xu D. MUFOLD: a new solution for protein 3D structure prediction. Proteins. 2010;78(5):1137–52.
14. Peng J, Xu J. RaptorX: exploiting structure information for protein alignments by statistical inference. Proteins. 2011;79(S10):161–71.
15. Li J, Cheng J. A Stochastic Point Cloud Sampling Method for Multi-Template Protein Comparative Modeling. Sci rep. 2016;6:25687.
16. Liaw A, Wiener M. Classification and regression by randomForest. R news. 2002;2(3):18–22.
17. Bhattacharya D, Cheng J. De novo protein conformational sampling using a probabilistic graphical model. Sci rep. 2015;5:16332.
18. Liu T, Wang Y, Eickholt J, Wang Z. Benchmarking deep networks for predicting residue-specific quality of individual protein models in CASP11. Sci Rep. 2016;6:19301.
19. Bhattacharya D, Cao R, Cheng J. UniCon3D: de novo protein structure prediction using united-residue conformational search via stepwise, probabilistic sampling. Bioinformatics. 2016;32(18):2791–9. doi:10.1093/bioinformatics/btw316.
20. Adhikari B, Bhattacharya D, Cao R, Cheng J. CONFOLD: residue-residue contact-guided *ab initio* protein folding. Proteins: Struct Funct Bioinf. 2015; 83(8):1436–49.
21. Simons KT, Bonneau R, Ruczinski I, Baker D. *Ab initio* protein structure prediction of CASP III targets using ROSETTA. Proteins: Struct Funct Bioinf. 1999;37(S3):171–6.
22. McGuffin L. The ModFOLD server for the quality assessment of protein structural models. Bioinformatics. 2008;24(4):586–7.
23. Wang Q, Vantasin K, Xu D, Shang Y. MUFOLD-WQA: a new selective consensus method for quality assessment in protein structure prediction. Proteins. 2011;79(SupplementS10):185–95.
24. McGuffin L, Roche D. Rapid model quality assessment for protein structure predictions using the comparison of multiple models without structural alignments. Bioinformatics. 2010;26(2):182–8.
25. Cao R, Bhattacharya D, Adhikari B, Li J, Cheng J. Massive integration of diverse protein quality assessment methods to improve template based modeling in CASP11. Proteins: Structure, Function, and Bioinformatics. 2015; 84:247–59. doi:10.1002/prot.24924.
26. Kryshtafovych A, Barbato A, Monastyrskyy B, Fidelis K, Schwede T, Tramontano A. Methods of model accuracy estimation can help selecting the best models from decoy sets: assessment of model accuracy estimations in CASP11. Proteins: Structure, Function, and Bioinformatics. 2015;84:349–69. doi:10.1002/prot.24919.
27. Cao R, Wang Z, Cheng J. Designing and evaluating the MULTICOM protein local and global model quality prediction methods in the CASP10 experiment. BMC Struct Biol. 2014;14(1):13.
28. Cao R, Wang Z, Wang Y, Cheng J. SMOQ: a tool for predicting the absolute residue-specific quality of a single protein model with support vector machines. BMC Bioinf. 2014;15(1):120.
29. Rykunov D, Fiser A. Effects of amino acid composition, finite size of proteins, and sparse statistics on distance-dependent statistical pair potentials. Proteins: Struct Funct Bioinf. 2007;67(3):559–68.
30. Zhang J, Zhang Y. A novel side-chain orientation dependent potential derived from random-walk reference state for protein fold selection and structure prediction. PLoS One. 2010;5(10):e15386.
31. Wang Z, Tegge AN, Cheng J. Evaluating the absolute quality of a single protein model using structural features and support vector machines. Proteins. 2009;75(3):638–47.
32. Shen M, Sali A. Statistical potential for assessment and prediction of protein structures. Protein Sci. 2006;15(11):2507–24.
33. Cao R, Cheng J. Protein single-model quality assessment by feature-based probability density functions. Sci Rep. 2016;6:23990.
34. Kalman M, Ben-Tal N. Quality assessment of protein model-structures using evolutionary conservation. Bioinformatics. 2010;26(10):1299–307.
35. Liithy R, Bowie J, Eisenberg D. Assessment of protein models with three-dimensional profiles. Nature. 1992;356:83–5.
36. Ray A, Lindahl E, Wallner B. Improved model quality assessment using ProQ2. BMC Bioinf. 2012;13(1):224.
37. Mishra A, Rao S, Mittal A, Jayaram B. Capturing native/native like structures with a physico-chemical metric (pcSM) in protein folding. Biochim Biophys Acta Proteins Proteomics. 2013;1834(8):1520–31.
38. Benkert P, Biasini M, Schwede T. Toward the estimation of the absolute quality of individual protein structure models. Bioinformatics. 2011;27(3):343–50.
39. Manavalan B, Lee J, Lee J. Random forest-based protein model quality assessment (RFMQA) using structural features and potential energy terms. PLoS One. 2014;9(9):e106542.
40. Uziela K, Wallner B. ProQ2: Estimation of Model Accuracy Implemented in Rosetta. Bioinformatics. 2016;32(9):1411–3.
41. Uziela K, Wallner B, Elofsson A. ProQ3: improved model quality assessments using Rosetta energy terms. 2016. arXiv preprint arXiv:160205832.
42. Yang J, Wang Y, Zhang Y. ResQ: an approach to unified estimation of B-factor and residue-specific error in protein structure prediction. J Mol Biol. 2016;428(4):693–701.
43. LeCun Y, Bengio Y, Hinton G. Deep learning. Nature. 2015;521(7553):436–44.
44. Zou WY, Wang X, Sun M, Lin Y. Generic object detection with dense neural patterns and regionlets. 2014. arXiv preprint arXiv:14044316.
45. Silver D, Huang A, Maddison CJ, Guez A, Sifre L, van den Driessche G, Schrittwieser J, Antonoglou I, Panneershelvam V, Lanctot M. Mastering the game of Go with deep neural networks and tree search. Nature. 2016; 529(7587):484–9.
46. Eickholt J, Cheng J. Predicting protein residue–residue contacts using deep networks and boosting. Bioinformatics. 2012;28(23):3066–72.

47. Deng H, Jia Y, Zhang Y. 3DRobot: automated generation of diverse and well-packed protein structure decoys. Bioinformatics. 2016;32(3):378-87.
48. Wang G, Dunbrack RL. PISCES: a protein sequence culling server. Bioinformatics. 2003;19(12):1589–91.
49. Simons K, Kooperberg C, Huang E, Baker D. Assembly of protein tertiary structures from fragments with similar local sequences using simulated annealing and Bayesian scoring functions. J Mol Biol. 1997;268(1):209–25.
50. John B, Sali A. Comparative protein structure modeling by iterative alignment, model building and model assessment. Nucleic Acids Res. 2003;31(14):3982–92.
51. Zhang Y, Skolnick J. Scoring function for automated assessment of protein structure template quality. Proteins: Struct Funct Bioinf. 2004;57(4):702–10.
52. Zemla A. LGA: a method for finding 3D similarities in protein structures. Nucleic Acids Res. 2003;31(13):3370–4.
53. Zhou H, Skolnick J. GOAP: a generalized orientation-dependent, all-atom statistical potential for protein structure prediction. Biophys J. 2011;101(8):2043–52.
54. Wu Y, Lu M, Chen M, Li J, Ma J. OPUS-Ca: a knowledge-based potential function requiring only Cα positions. Protein Sci. 2007;16(7):1449–63.
55. Yang Y, Zhou Y. Specific interactions for *ab initio* folding of protein terminal regions with secondary structures. Proteins: Struct Funct Bioinf. 2008;72(2):793–803.
56. Hinton GE. Training products of experts by minimizing contrastive divergence. Neural Comput. 2002;14(8):1771–800.
57. Nawi NM, Ransing MR, Ransing RS: An improved learning algorithm based on the Broyden-Fletcher-Goldfarb-Shanno (BFGS) method for back propagation neural networks. In Sixth International Conference on Intelligent Systems Design and Applications (Vol. 1, pp. 152–157). IEEE. ISBN: 0-7695-2528-8. IEEE Computer Society Washington, DC, USA.
58. Hall M, Frank E, Holmes G, Pfahringer B, Reutemann P, Witten IH. The WEKA data mining software: an update. ACM SIGKDD Explor Newsl. 2009;11(1):10–8.