Statistics and Its Interface Volume 5 (2012) 109–115

Protein structural model selection based on protein-dependent scoring function

Zhiquan He, Jingfen Zhang, Yang Xu, Yi Shang and Dong Xu*

Selection of good models from a structural model pool is an important and challenging step in protein structure prediction. While various score functions have been developed, their applications in protein structure predictions are unsatisfactory. In this study, we developed a novel two-stage optimization method which effectively combines a set of basic scoring functions for improving the selection performance. In the first stage of protein-dependent optimization, this method combines seven scoring functions and optimizes the weights among them on the model pool of each protein. In the second stage, the method integrates scores with optimized protein-dependent weights, and then seeks correlations among these scores and structural features using a Support Vector Machine (SVM) to predict the quality of protein structures. Test results on two benchmarks from different model generation methods showed that the sum of basic scoring functions with optimized weights achieved better model selection performance than any individual scoring function or equal-weight combination of these scoring functions. A leave-one-out test demonstrated further improvement in the second stage over the score of the weighted sum.

Keywords and phrases: Protein model selection, Score combination, Scoring functions.

1. INTRODUCTION

Protein structure prediction has been an important and challenging research topic for more than two decades [7, 16]. While genome-sequencing projects generate large amounts of protein sequences, the lack of tertiary structures is a main obstacle to fully understanding the functions of these proteins. Traditionally, experimental determination of protein structures has utilized both X-ray crystallography and nuclear magnetic resonance (NMR), which are time consuming and costly. Computational structure prediction from amino acid sequence is also a viable solution [8]. Recent reviews illustrated the applications of predicted models with different qualities [2, 10]. For example, high-resolution models with root mean square deviation (RMSD) of 1 to 1.5$\dot{A}$ are useful for almost any application, including drug design; and even if the model quality decreases to about 6$\dot{A}$ RMSD, the function of the protein could still be predicted thereby enabling prediction methods like mutagenesis to be designed based on the model [10].

Currently, most structure prediction methods, such as Robetta [11], Rosetta [19], I-TASSER [18, 24, 27], and MUFOLD [29] adopt a sampling-selection strategy. With this strategy, the first step is to generate a large number of candidate models with a sampling procedure; and the second step is to apply a scoring method to identify the most native-like conformations. For this protocol to work, it is required that the sampling procedure is capable of producing at least some near-native conformations and the scoring method is able to identify more native-like structures from the structural model pool [20].

Methods for ranking structural models roughly fall into four categories: physical-based energies, knowledge-based scoring functions, consensus methods, and machine learning based approaches. Physical-based energy functions [12, 15] compute the energy of a protein structure based on physics principles at the atomic level. Physical energies are often too sensitive to small atomic changes, and hence they are not widely used in model selection. Knowledge-based scoring functions, such as OPUS-CA [23], DFire [31] and RW [30], score the models based on the statistical information of structural attributes in known native structures. These scoring functions are widely used in protein structure prediction. However, knowledge-based scoring functions can only reflect some aspects of protein structures. For example, OPUS-CA uses the distance-dependent energies from the C-alpha atoms of a model, while RW is a side-chain orientation dependent potential. While some success is achieved, overall they have limited discerning power for ranking structural models. Consensus methods [3, 6, 22] assume that the models most similar to others in a dataset have better quality. This approach has been the most successful for model quality assessment in Critical Assessment of Protein Structure Prediction (CASP) [14], where the model pool contains the top predictions submitted by the attending groups. However, consensus methods often do not work well when the dataset is not dominated by good models. Hence, individual tools for protein structure prediction usually do not include consensus methods. Machine learning methods [6, 17], which are typically support vector machine or neural network, evaluate models according to some learned "rules". The input

*Corresponding author.

features for training include sequence or structural model attributes, and the output target value is the real model quality. Such methods do not rely on a specific statistical model and train a score function using both native and incorrect structures, which may improve model selection over knowledge-based scoring functions. A drawback of machine learning methods is that it tends to over-train the training dataset and may not generally work well for a different kind of test structural model.

Scoring functions have performance inconsistencies for different proteins. We believe that combining several scoring functions can result in better performance as they complement each other to some extent. Although consensus methods do not work well for a dataset that does not contain predominantly good models, one may overcome this by selecting a subset of good models using scoring functions. Based on these considerations, in this study, we proposed a two-stage optimization approach to take advantages of scoring functions, consensus method and machine learning. In the first protein-dependent optimization, different "noisy" scoring functions were combined to improve the sensitivity of scores for model selection. In this step, each target protein has a pool of structural models without knowing the native structure. For each protein, a subset of models was selected using basic scoring functions to remove likely poor models. Then weights for these scoring functions were optimized on the selected model set of each protein. Ideally, we should use the real GDT-TS score [28] (one of the most widely used scores for protein quality) of models to optimize the weights. Due to lack of native structures, we replaced the real GDT-TS score with a consensus GDT-TS score, which is an estimate of GDT-TS using a consensus approach. The sum of these scores with the optimized weights can be directly used to rank models. However, it was still "noisy" due to the errors introduced by scoring functions and the consensus method. In the second stage optimization, we integrated the weighted scoring functions, correlations of these scores to consensus GDT-TS, model quality computed by consensus method and structural features to train an SVM that maps these features to the real GDT-TS scores based on separate protein targets with structural model pools and known native structures. Through the two sequential optimizations, the resulting score can gain sufficient discerning power to outperform basic scoring functions and consensus method for model selection. We have applied this new method to two benchmarks and demonstrated that the weighted sum of individual scoring functions improved the top-1 and top-5 model selection performance, and a following SVM gained further improvement.

2. METHODS

An overview of our method is presented in Figure 1. The first step was to compute the basic scores for each model using scoring functions. Then for each protein, the best weights

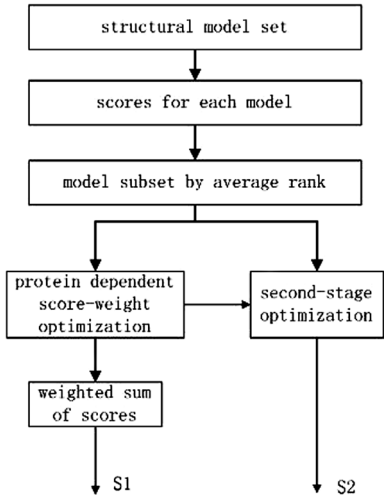

Figure 1. Method Flowchart.

for scoring functions were obtained through the protein-specific optimization on the subset (at most top 300) models selected by the average rank based on basic scores. The resultant weighted sum (S1 in Figure 1) can be directly used to rank the models. The basic scores and weights were integrated into the second stage optimization using an SVM which was trained on models from different proteins with the real GDT-TS score of each model as the target value.

2.1 Scoring functions

In this method, five published protein structure quality assessment (QA) scores were selected, namely OPUS-CA [23], OPUS-PSP [13], DFire [31], DDFire [26] and RW [30]. These scores evaluate structure models from different perspectives. Also we computed two additional statistical based scores, i.e., environment fitness score and secondary structure similarity score, which are widely used in threading-based protein structure predictions [25].

Environment fitness score This score measures the propensity of an amino acid type a to appear in a structural environment env_j on the model. The environment type is specified by the secondary structure type (H: helix, E: beta sheet, or C: coil) and solvent accessibility type (B: buried, I: intermediate or E: exposed). The environment fitness score is given by

$$envfitness = \sum_{j=1}^{N} \sum_{a=1}^{20} prob(env_j, a) \times prob(j, a) \quad (1)$$

where N is the protein sequence length. $prob(env_j, a)$ is the probability of amino acid type a to appear in structural environment env_j obtained through statistical analysis on a set of training native structures [25]. It is worth mentioning that these structures had no overlap with the ones used in the following benchmark tests. $prob(j, a)$ is the probability

of amino acid type a occurring at position j of the protein, which can be calculated from the sequence profile generated by PSIBLAST [1].

Secondary structure similarity score For each model, we computed its actual secondary structure based on its 3D coordinates using DSSP [9]. We also used PSIPRED [4] to predict the secondary structure from its amino acid sequence. The similarity between these two secondary structures is a good indication of model quality. Higher secondary structure similarity usually means better model quality. Suppose the secondary structure type at position j of model is S_d, and the corresponding predicted secondary structure from sequence by PSIPRED is S_p with confidence value P, the score is defined as

$$sssimilarity = \sum_{j=1}^{N} prob_j(S_d, S_p, P) \tag{2}$$

where $S_d, S_p \in \{H, E, C\}, P \in [0, 9]$ and $prob(S_d, S_p, P)$ is the probability of S_d being predicted as S_p with confidence value P, obtained from a training dataset whose proteins had no overlap with the ones used in the following benchmark tests.

2.2 Protein-dependent weights optimization

Let $s_1, s_2, \ldots, s_7$ be the seven scores of a model, and $w_1, w_2, \ldots, w_7$ be the weights for the scores. We optimized the weights by minimizing

$$L_2 = \sum_{i_1, i_2} \left[\sum_{j=1}^{7} w_j (s_j^{i_1} - s_j^{i_2}) - [GDT(i_1) - GDT(i_2)] \right]^2$$

where $w_j < 0$ and i_1, i_2 are two structural models of the same protein. $s_j^{i_1}$ is score j of structure model i_1 and $GDT(i_1)$ is the GDT-TS score of model i_1.

In practice, GDT-TS score is not available as we do not have the native structure. So we used consensus GDT-TS score, $cgdt()$, to approximate the real $GDT()$ score. A reference set R containing the top 300 models was selected according to the average rank using the seven basic scores. cgdt of a model is defined as the average GDT-TS score to the remaining models in R. Thus the weights were optimized on R by minimizing

$$L_2 = \sum_{i_1, i_2 \in R} \left[\sum_{j=1}^{7} w_j (s_j^{i_1} - s_j^{i_2}) - [cgdt(i_1) - cgdt(i_2)] \right]^2.$$

Let $x_j^k = s_j^{i_1} - s_j^{i_2}$ and $y_k = cgdt(i_1) - cdgt(i_2)$, we have

$$L_2 = \sum_{k} \left[\sum_{j=1}^{7} w_j x_j^k - y_k \right]^2, \quad w_j < 0. \tag{3}$$

Further, let $W = [w_1, \ldots, w_7]^T$ and $X_k = [x_1^k, \ldots, x_7^k]^T$, Eqn. (3) becomes

$$L_2 = W^T \sum_k X_k X_k^T W - 2W^T \sum_k y_k X_k + \sum_k y_k^2, \quad W < 0. \tag{4}$$

Minimization of Eqn. (4) was solved by quadratic programming. Before optimization, all the scores were normalized to Z-score. Z-score of score S is defined as $Z = \frac{S - avg(S)}{dev(S)}$, where $avg(S)$ is the mean value and $dev(S)$ is the standard deviation in the structural model pool. Each scoring function has its "direction"; for example, OPUS-CA is "negative" compared to GDT-TS, which means lower OPUS-CA values usually have higher GDT-TS scores. In the actual optimization, the "directions" of seven scores were all adjusted to be "negative". Also, due to the noise in the training data, weights were constrained to be less than -0.0001 to keep the optimization from reversing or disabling any scores. After optimization, weights were obtained for each score and the score S_1 in Figure 1 was $S_1 = \sum_{j=1}^{7} w_j s_j$.

2.3 Second stage optimization

This optimization was implemented as an SVM. The input features for each model included:

- Weighted scores $w_j s_j, j = 1, \ldots, 7$.
- Spearman correlation of each score $s_j, j = 1, \ldots, 7$ to consensus GDT-TS score. The correlations of different scores indicate their relative performance on models of a specific protein.
- Consensus GDT-TS score *cgdt*.
- Another secondary structure score to strengthen the similarity between the actual secondary structure in model and the predicted one from sequence. It is defined as $SSIden = \frac{\sum_{j=1}^{N} \delta(SS_p, SS_d)}{N}$, where N is protein sequence length and

$$\delta(SS_p, SS_d) = \begin{cases} 1 & SS_p = SS_d \\ 0 & SS_p \neq SS_d. \end{cases}$$

- Solvent accessibility (SA) matching scores, which is similar to *SSIden*. $SAIden = \frac{\sum_{j=1}^{N} \delta(SA_p, SA_d)}{N}$, where N is protein sequence length and

$$\delta(SA_p, SA_d) = \begin{cases} 1 & SA_p = SA_d \\ 0 & SA_p \neq SA_d. \end{cases}$$

  SA_p is the predicted solvent accessibility by SSPro [5] and SA_d is computed from models by DSSP [9] with cutoff 25% (above which means the exposed state and otherwise the buried state).

Although the secondary structure and solvent accessibility information were used in the seven scores, SSIden and SAIden were more direct to help SVM to learn the "weak" relationship between features and real GDT-TS score. The SVM was trained using SVMLight [21] with a linear kernel.

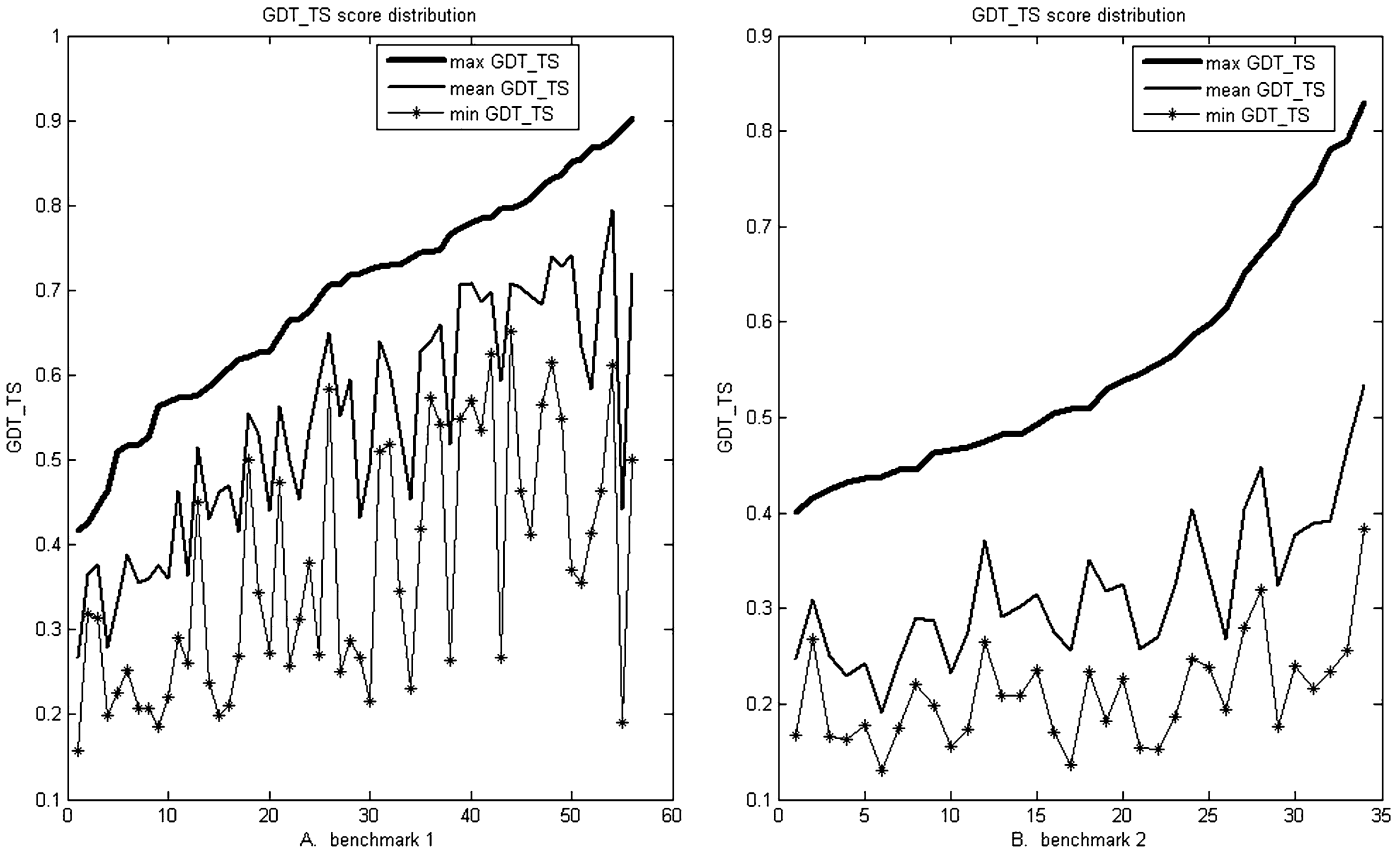

Figure 2. Model quality measured by GDT-TS score to the native structure. The X-axis is the proteins of each benchmark sorted by the GDT-TS score of the best model. (A) Model quality distribution of benchmark1. (B) Model quality distribution of benchmark2.

2.4 Dataset

We applied the method to two benchmarks produced by different model generation methods. Benchmark1 was from Yang Zhang's lab, generated by the I-TASSER ab initio modeling tool, containing 56 proteins. The other one, benchmark2, included models generated by Robetta or Rosetta, containing 34 CASP8 proteins. Each protein in both benchmarks had hundreds of decoys. Figure 2 shows the maximum, average and minimum GDT-TS score of models of each protein for both benchmarks. The best model of each protein had a GDT-TS score greater than 0.4, which ensured that the pool contained some reasonably good models.

3. RESULTS

In the test, each score was used to rank the models of a given protein. We used four metrics to compare the performance of each scoring method. Table 1 compares seven basic scores mentioned above, avezscore, averank and S1 on benchmark1 and benchmark2. The term averank was used to select the top 300 models for each protein to optimize the weights for S1. Table 2 shows the selection performance of cgdt and S2 on the subset models selected by averank.

As shown in Table 1, weighted sum with the optimized weights improved over seven basic scores, in top-1 and top-5 selection performance. For example, for benchmark2, the best scoring function was DDFire, which had GDT1 performance of 0.3976 and avgGDT5 of 0.3833, while S1 achieved GDT1 of 0.4012 and avgGDT5 of 0.3977. Furthermore, weight optimization improved over avezscore and averank in selection performance, especially for benchmark2, as our optimization was carried out on the subset selected by averank. For Pearson and Spearman, we can see from Table 1 that S1 had the best correlation to the real GDT-TS score among the scores being compared on both benchmarks. For example, for benchmark1, although the selection improvement of S1 over the best of other scores was small, the improvement in correlation was quite significant. In Figure 3 we took the protein 1SHF from benchmark1 as an example to show the score distribution. It is evident that S1 had a much better correlation to real GDT-TS than sssimilarity and the top model selected by S1 was better than the one by sssimilarity.

Table 2 shows that after selecting the top 300 models for each protein using averank, the GDT-TS loss between the best model in the 300-model set and the best model in the entire pool was acceptable for benchmark1; the average

Table 1. Comparison of scores based on their performance. "GDT1" is the average GDT-TS score of top 1 model; "avgGDT5" is the average of the mean GDT-TS score of top 5 models. "Pearson" indicates the Pearson correlation to real GDT-TS and "Spearman" is the Spearman correlation to real GDT-TS score. "avezscore" is the sum of the seven scores after normalization; "averank" is the average rank using seven basic scores. "S1" is the weighted sum of basic scores

	Benchmark1				Benchmark2			
	GDT1	avgGDT5	Pearson	Spearman	GDT1	avgGDT5	Pearson	Spearman
GDT-TS	0.6918	0.6737	1.0000	1.0000	0.5504	0.5281	1.0000	1.0000
OPUS-CA	0.5935	0.5904	0.4952	0.4159	0.3769	0.3705	0.2980	0.2709
OPUS-PSP	0.5670	0.5715	0.2893	0.2906	0.3171	0.3253	0.0993	0.0941
DFire	0.5984	0.5882	0.5332	0.4416	0.3389	0.3277	0.0723	0.0786
DDFire	0.5984	0.5883	0.5328	0.4411	0.3976	0.3833	0.3050	0.2718
RW	0.5927	0.5855	0.4909	0.4178	0.3707	0.3738	0.2987	0.2727
envfitness	0.5604	0.5691	0.3805	0.2985	0.3501	0.3396	0.1050	0.0962
sssimilarity	0.5836	0.5823	0.3578	0.2938	0.3571	0.3623	0.2366	0.2152
avezscore	0.5966	0.5919	0.5486	0.4530	0.3856	0.3823	0.3291	0.2987
averank	0.5970	0.5895	0.5126	0.4562	0.3861	0.3707	0.3200	0.2969
S1	**0.5989**	**0.5953**	**0.5824**	**0.4841**	**0.4012**	**0.3977**	**0.3709**	**0.3489**

Table 2. Comparison of scores based on reference set. "GDT1" is the average GDT-TS score of top 1 model; "avgGDT5" is the average of the mean GDT-TS score of top 5 models. "cgdt" is the consensus GDT-TS, and "S2" corresponds to the SVM output in Figure 1

	Benchmark1		Benchmark2	
	GDT1	avgGDT5	GDT1	avgGDT5
GDT-TS	0.6892	0.6713	0.5504	0.5273
cgdt	0.6047	0.6030	0.4351	0.4217
S2	**0.6098**	**0.6034**	**0.4446**	**0.4220**

GDT-TS loss was only $0.6918 - 0.6892 = 0.0026$. For benchmark2, the best models of all proteins were kept in the selected top-300 model set, i.e., with 0 GDT-TS loss. Table 2 also shows the leave-one-out performance of the SVM. This research trained different models for benchmarks 1 and 2 as they were generated by different methods and had quite different structural characteristics and distributions which were reflected by the diverse performances of basic scores. In leave-one-out training and testing, all proteins were tested using one model while the remaining were used as training data. Table 2 shows that S2 improved over cgdt on both benchmarks, especially in GDT1 performance. For benchmark1, GDT1 of S2 was 0.6098, which gained about half a GDT-TS point ($0.6098 - 0.6047 = 0.0051$) over cgdt (0.6047). For benchmark2, the improvement over cgdt in GDT1 was $0.4446 - 0.4351 = 0.0095 \cong 0.01$. On the other hand, S2 had significantly better GDT1 and avgGDT5 performance than basic scores. Especially, for benchmark2, the best basic scoring function was DDFire, whose GDT1 was 0.3976, while S2 had GDT1 of 0.4446. The improvement was $0.4446 - 0.3976 = 0.047$.

4. DISCUSSION

Our new approach combined the advantages of various methods and avoided some of their limitations. Existing scoring functions such as OPUS-CA and DFire do not work consistently well for model selection of different proteins especially when models are generated by different methods. Consensus method depends only on the dataset itself and does not use any information from native structures. In order to improve the selection performance, for each protein, we trained the weights for each score on a reference set which was selected to enrich the overall quality of the smaller pool. The resultant weighted score was less noisy and more correlated with the real GDT-TS score. With the weighted scores, it is more advantageous for the second stage optimization to learn the weak intrinsic correlation between input features and real model quality.

However, several factors may affect the performance of our method. One such factor is model distribution. S1 and S2 had more GDT-TS loss between the selected top-1 model and the best model in the pool in benchmark2 than in benchmark1. Specifically, GDT-TS loss of S2 in benchmark1 was $0.6918 - 0.6098 = 0.082$; while for benchmark2, the GDT-TS loss was $0.5504 - 0.4446 = 0.1058$. Comparing the two distributions of model pools in Figure 2, it is evident that the gap between max and mean GDT-TS in benchmark2 was much bigger than that in benchmark1. The distribution difference also affects the performance of other scores in the same way. For benchmark2, GDT1 of the real GDT-TS score was 0.5504, while all basic scores were less than 0.4, losing more than 0.15, significantly bigger than that in benchmark1.

For the second stage optimization, selection of features and learning method directly affects the performance of S2. Although S2 is not significantly better than cgdt on either of the two benchmarks, the S2 method has some merit. In particular, cgdt and basic scoring functions have different properties and combining them theoretically may improve the performance. Furthermore, the performance of cgdt depends on the distribution of the model pool or how the model pool is generated. The model pool generated in CASP or by the tools that guarantee good sampling of structural

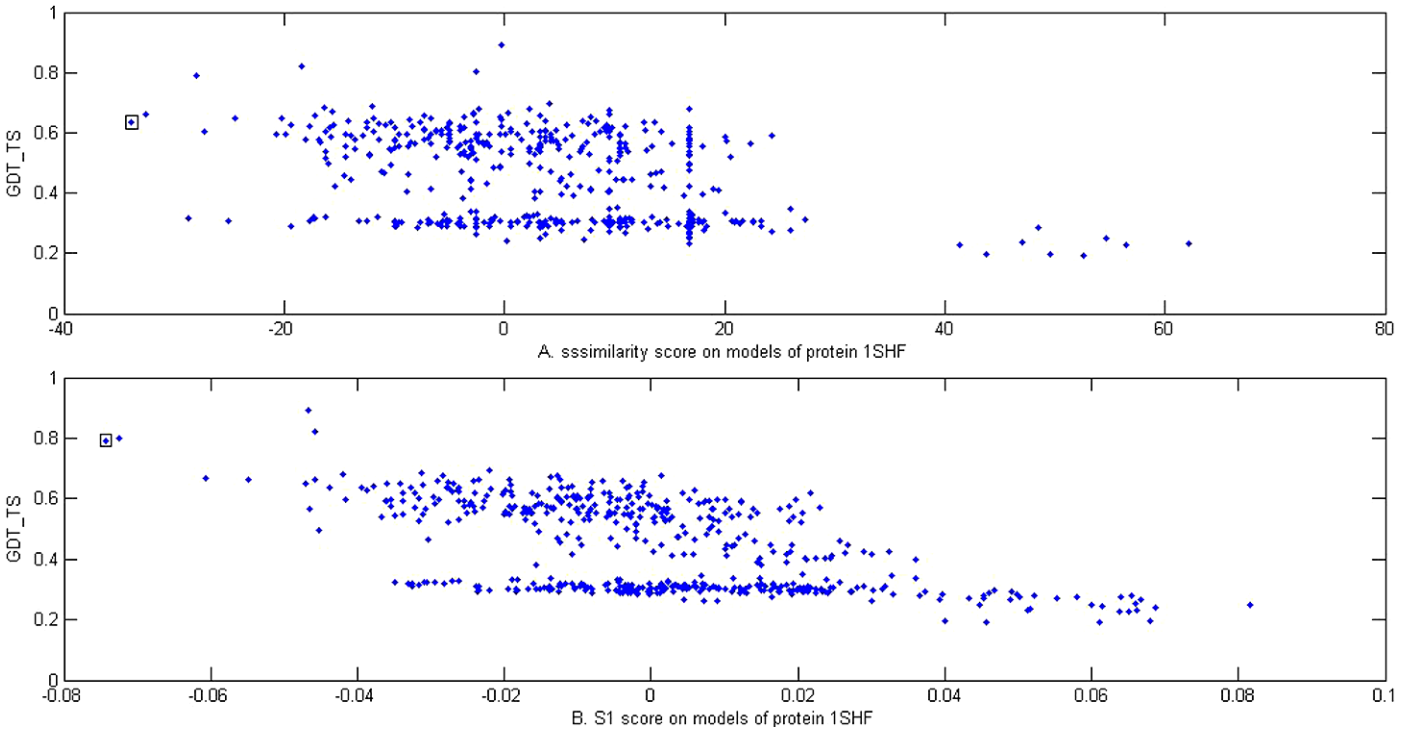

Figure 3. Score distributions for models of protein 1SHF from benchmark1. (A) Score distribution of sssimilarity with respect to GDT-TS. (B) Score distribution of S1 with respect to GDT-TS. The point highlighted in the box is the top model selected by the score.

conformation can lead to good performance of cgdt; otherwise the performance of cgdt may not be good. In addition, this research concluded that the S2 method has significant room for improvement. We are exploring a better way to do the second stage optimization and combine the two stages. For example, one may use the priori general information of model quality vs. a given scoring function and use that information to guide optimization. The SVM here was developed to demonstrate that integrating weighted scores, their statistical features and structure-related features into optimization over different proteins can improve the performance over any individual feature. On the other hand, more advanced machine learning techniques, such as random forests may further enhance the performance.

There are some limitations of our method. Given that it is based on training from a model pool, it may not be applicable to simultaneously assess models from different generation methods as they may have different characteristics or distributions. For example, our method may not be applicable to the model pool generated by different servers in CASP. Our method is mainly designed for model selection with a single tool which is most practical in protein structure prediction applications.

ACKNOWLEDGEMENTS

This work has been supported by National Institutes of Health Grant R21/R33-GM078601 and by the Paul K. and Diane Shumaker Endowment in Bioinformatics. We would like to thank the authors of OPUS-CA, OPUS-PSP, DFire, DDFire and RW for contributing these scores to the research community. We thank Yang Zhang and Jiong Zhang for providing the structural models. Major computations were performed using the UMBC computing resource. We also thank Qingguo Wang and Xiaohu Shi for helpful discussions.

Received 31 March 2011

REFERENCES

[1] Altschul, S. F., Madden, T. L., Schaffer, A. A., Zhang, J., Zhang, Z., Miller, W., Lipman, D. J. (1997). Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. *Nucleic Acids Res.* **25**(17) 3389–3402.

[2] Baker, D., Sali, A. (2001). Protein structure prediction and structural genomics. *Science* **294**(5540) 93–96.

[3] Benkert, P., Tosatto, S. C., Schwede, T. (2009). Global and local model quality estimation at CASP8 using the scoring functions QMEAN and QMEANclust. *Proteins* **77** Suppl 9:173–180.

[4] Bryson, K., McGuffin, L. J., Marsden, R. L., Ward, J. J., Sodhi, J. S., Jones, D. T. (2005). Protein structure prediction servers at University College London. *Nucleic Acids Res.* **33** (Web Server issue) W36–38.

[5] Cheng, J., Randall, A. Z., Sweredoski, M. J., Baldi, P. (2005). SCRATCH: A protein structure and structural feature prediction server. *Nucleic Acids Res.* **33** (Web Server issue) W72–76.

[6] Cheng, J., Wang, Z., Tegge, A. N., Eickholt, J. (2009). Prediction of global and local quality of CASP8 models by MULTICOM series. *Proteins* **77** Suppl 9:181–184.

[7] Cozzetto D., Tramontano, A. (2008). Advances and pitfalls in protein structure prediction. *Curr. Protein Pept. Sci.* **9** 567–577.

[8] Domingues, F. S., Koppensteiner, W. A., Sippl, M. J. (2000). The role of protein structure in genomics. *FEBS Lett.* **476**(1–2) 98–102.

[9] Kabsch, W., Sander, C. (1983). Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers* **22**(12) 2577–2637.

[10] Kihara, D., Chen, H., Yang, Y. D. (2009). Quality assessment of protein structure models. *Curr. Protein Pept. Sci.* **10**(3) 216–228.

[11] Kim, D. E., Chivian, D., Baker, D. (2004). Protein structure prediction and analysis using the Robetta server. *Nucleic Acids Res.* **32** (Web Server issue) W526–531.

[12] Lazaridis, T., Karplus, M. (1999). Discrimination of the native from misfolded protein models with an energy function including implicit solvation. *J. Mol. Biol.* **288**(3) 477–487.

[13] Lu, M., Dousis, A. D., Ma, J. (2008). OPUS-PSP: An orientation-dependent statistical all-atom potential derived from side-chain packing. *J. Mol. Biol.* **376**(1) 288–301.

[14] Moult, J., Pedersen, J. T., Judson, R., Fidelis, K. (1995). A large-scale experiment to assess protein structure prediction methods. *Proteins* **23**(3) ii–v.

[15] Petrey, D., Honig, B. (2000). Free energy determinants of tertiary structure and the evaluation of protein models. *Protein Sci.* **9**(11) 2181–2191.

[16] Petrey, D., Honig, B. (2005). Protein structure prediction: Inroads to biology. *Mol. Cell* **20** 811–819.

[17] Qiu, J., Sheffler, W., Baker, D., Noble, W. S. (2008). Ranking predicted protein structures with support vector regression. *Proteins* **71**(3) 1175–1182.

[18] Roy, A., Kucukural, A., Zhang, Y. (2010). I-TASSER: A unified platform for automated protein structure and function prediction. *Nat. Protoc.* **5**(4) 725–738.

[19] Simons, K. T., Bonneau, R., Ruczinski, I., Baker, D. (1999). Ab initio protein structure prediction of CASP III targets using ROSETTA. *Proteins*, Suppl 3:171–176.

[20] Stumpff-Kane, A. W., Feig, M. (2006). A correlation-based method for the enhancement of scoring functions on funnel-shaped energy landscapes. *Proteins* **63**(1) 155–164.

[21] Thorsten J. (1999). Making large-scale SVM learning practical. *Advances in Kernel Methods – Support Vector Learning* 169–184.

[22] Wallner, B., Elofsson, A. (2007). Prediction of global and local model quality in CASP7 using Pcons and ProQ. *Proteins* **69** Suppl 8:184–193.

[23] Wu, Y., Lu, M., Chen, M., Li, J., Ma, J. (2007). OPUS-Ca: A knowledge-based potential function requiring only Calpha positions. *Protein Sci.* **16**(7) 1449–1463.

[24] Wu, S., Skolnick, J., Zhang, Y. (2007). Ab initio modeling of small proteins by iterative TASSER simulations. *BMC Biol.* **5** 17.

[25] Xu, Y., Xu, D. (1998). Uberbacher EC: An efficient computational method for globally optimal threading. *J. Comput. Biol.* **5**(3) 597–614.

[26] Yang, Y., Zhou, Y. (2008). Specific interactions for ab initio folding of protein terminal regions with secondary structures. *Proteins* **72**(2) 793–803.

[27] Zhang, Y., Arakaki, A. K., Skolnick, J. (2005). TASSER: An automated method for the prediction of protein tertiary structures in CASP6. *Proteins* **61** Suppl 7:91–98.

[28] Zhang, Y., Skolnick, J. (2004). Scoring function for automated assessment of protein structure template quality. *Proteins* **57**(4) 702–710.

[29] Zhang, J., Wang, Q., Barz, B., He, Z., Kosztin, I., Shang, Y., Xu, D. (2010). MUFOLD: A new solution for protein 3D structure prediction. *Proteins* **78**(5) 1137–1152.

[30] Zhang, J., Zhang, Y. (2010). A novel side-chain orientation dependent potential derived from random-walk reference state for protein fold selection and structure prediction. *PLoS One* **5**(10) e15386.

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

Zhiquan He
Department of Computer Science
University of Missouri
MO 65211
USA
Christopher S. Bond Life Sciences Center
University of Missouri
MO 65211
USA
E-mail address: zhy78@mizzou.edu

Jingfen Zhang
Department of Computer Science
University of Missouri
MO 65211
USA
Christopher S. Bond Life Sciences Center
University of Missouri
MO 65211
USA
E-mail address: zhangjingf@gmail.com

Yang Xu
Department of Computer Science
University of Missouri
MO 65211
USA
Christopher S. Bond Life Sciences Center
University of Missouri
MO 65211
USA
E-mail address: yxqc4@mizzou.edu

Yi Shang
Department of Computer Science
University of Missouri
MO 65211
USA
E-mail address: shangy@missouri.edu

Dong Xu
Department of Computer Science
University of Missouri
MO 65211
USA
Christopher S. Bond Life Sciences Center
University of Missouri
MO 65211
USA
E-mail address: xudong@missouri.edu