

# On-line Hierarchy of General Linear Models for Selecting and Ranking the Best Predicted Protein Structures

Hani Zakaria Girgis and Jason J. Corso and Daniel Fischer

**Abstract**—To predict the three dimensional structure of proteins, many computational methods sample the conformational space, generating a large number of candidate structures. Subsequently, such methods rank the generated structures using a variety of model quality assessment programs in order to obtain a small set of structures that are most likely to resemble the unknown experimentally determined structure. Model quality assessment programs suffer from two main limitations: (i) the rank-one structure is not always the best predicted structure; in other words, the best predicted structure could be ranked as the 10th structure (ii) no single assessment method can correctly rank the predicted structures for all target proteins. However, because often at least some of the methods achieve a good ranking, a model quality assessment method that is based on a consensus of a number of model quality assessment methods is likely to perform better. We have devised the STPdata algorithm, a consensus method based on five model quality assessment programs. We have applied it to build an on-line “custom-trained” hierarchy of general linear models to select and rank the best predicted structures. By “custom-trained”, we mean for each target protein the STPdata algorithm trains a unique model on data related to the input target protein. To evaluate our method we participated in CASP8 as human predictors. In CASP8, the STPdata algorithm has trained 128 hierarchical models for each of the 128 target proteins. Based on the official results of CASP8 our method outperformed the best server by 6% and won the fourth position among human predictors. Our CASP results are purely based on computational methods without any human intervention.

## I. INTRODUCTION

Proteins are very important molecules in nature whose function depends on their three-dimensional structures. The knowledge of the tertiary structure of the proteins has applications in medicine, drug design, the development of pesticides and herbicides, and in food industry [1]. The three-dimensional structure of the proteins can be determined experimentally by X-ray crystallography, nuclear magnetic resonance and other methods. However, these methods are slow, expensive, and not applicable to all proteins. Based on

Hani Girgis is with the Computer Science Department, The Johns Hopkins University, Baltimore, MD, USA hgirgis1@jhu.edu

Jason Corso and Daniel Fischer are with the Department of Computer Science and Engineering, University at Buffalo, The State University of New York, Buffalo, NY 14260, USA jcorso@buffalo.edu, df33cse@google.com

Hani Girgis conducted this research at the University at Buffalo, The State University of New York where he was supported by NIH grant 5R01GM073814

We thank the authors of the model quality assessment programs for making their methods available. In addition we are grateful for the human predictors and the authors of protein structure prediction servers who participated in CASP6, CASP7, and CASP8 for their predicted structures. We would like to thank the CASP8 organizers who gave us the opportunity to evaluate our methods in a truly blind way

We appreciate the anonymous reviewers' suggestions and efforts

a conservative estimate, there are more than  $10^{10}$  proteins on earth [2] in addition to the engineered ones. The importance of the knowledge of proteins shape, the abundance of proteins of unknown structures, and the limitations of the experimental methods have all contributed to the need for the computational methods to predict the three dimensional shape of proteins. The currently available computational methods of protein structure prediction usually generate large number of predicted structures for the same target protein. Therefore, determining or predicting the quality of a predicted structure without knowing the experimentally determined structure is an important problem in the protein structure prediction field. A model quality assessment program (MQAP) aims to predict the quality of a computationally generated structure without knowing the real experimental structure.

In our work we developed a machine learning based model quality assessment program. Traditionally, several machine learning algorithms have been used to learn one function to predict the quality of the computationally generated structures for all target proteins. In contrast to the traditional approaches, we have developed a framework which is capable of learning a custom scoring function for each target protein. In this paper we propose the idea of a custom-trained expert. Related model quality assessment programs [3]–[7] are based on machine learning algorithm. The learning algorithm learns a function that maps a combination of a set of features extracted from the computationally generated structures to a quality score such as the RMSD score [8], the MaxSub score [9], the TM score [10], and the GDT score [11]. The developers of these methods formulated the problem as a regression problem. The majority of these methods except the one developed by Qiu et al. [7] treat all structures the same way during training, while Qiu's method pays attention to the high quality structures during training. In our work we also distinguish between the high quality structures from the low quality structures during training and testing. We formulate the problem as a classification task followed by a regression task. The two main contributions of our work are: (i) we devised the STPdata algorithm to build a custom-trained on-line model for each target protein (ii) we used a hierarchical model to select the best predicted structures then rank these structures. At the first level, we classify the structures into two classes: the high quality structures and the low quality structure. At the second level, we use a regression model to rank the high quality structures.

To test our method we participated in CASP8 as human predictors. Based on the official results of CASP8 [12] our method outperformed the best server by 6% and won the

fourth position among human predictors. Our results are purely based on computational methods and there was no human intervention.

## II. METHODS

### A. Data and Preprocessing

**Data:** the available data consists of the predicted structures submitted to CASP6 [13], CASP7 [14], and CASP8. We used CASP6 data in training, servers' predictions in CASP7 in validation, and CASP7 humans' predictions in testing. The first author has participated in CASP8; therefore, the CASP8 data is used as an independent test conducted by CASP8 assessors. We consider all of the five structures submitted by each group not only the rank-one structures for two reasons: (i) the rank-one structure predicted by each group is not necessarily the best predicted structure; the best predicted structure may be ranked as the 4<sup>th</sup> or the 5<sup>th</sup> [15] (ii) our goal is to develop a model quality assessment program which is capable to select and rank the best predicted structures among a large set of structures. All data are preprocessed as the following: (i) select structures that are at least 85% complete (ii) select full atoms structures (iii) run the Modeller [16] program on the selections, then remove structures whose MaxSub score with the original model is less than 0.85. During training, target proteins whose best predicted structure has a MaxSub score less than 0.3 are excluded from the training set. In addition, all structures which have MaxSub scores less than 0.1 are also removed from the data sets <sup>1</sup>.

**Algorithm Input And Output:** our learning algorithm predicts the structure rank based on the MaxSub score [9]. The MaxSub score is the three dimensional similarity between two protein structures. The learning algorithm can be extended easily to predict ranks based on additional similarity measures such as the RMSD score [8], the GDT score [11], and the TM score [10]. However, we have decided to use the MaxSub score due to the high correlation among these similarity scores. The inputs to our learning algorithm are five scores obtained from the following MQAPs: ProQ [3], Prosa<sub>pair</sub> [17], [18], ModCheck [19], DFire [20], and the 3dSim which is the three dimensional similarity score. We have used these five scores for two reasons. First, 3dSim, ProQ, and ModCheck are highly correlated with the MaxSub score. Second, ModCheck, DFire, ProQ, and Prosa-Pair have the lowest average of incorrectly ranking the true rank-one structure. 3dSim is the average of the MaxSub scores between a structure and the other structures predicted for the same target protein provided that the MaxSub similarity score between the two models is greater than 0.4. Similar scores are implemented by [4], [5], [7], [21]. Equations 1 and 2 show how the 3dSim score is computed.

$$\delta(i, j) = \begin{cases} 1 & \text{if } \text{MaxSub}(i, j) \geq 0.4; \\ 0 & \text{o.w.} \end{cases} \quad (1)$$

<sup>1</sup>Some of the excluded structures were removed erroneously due to software error. However, this accident does not affect the outcome of the training or the validation of our algorithms.

$$3dSim(m, S) = \frac{\sum_{s_j \in S \text{ and } s_j \neq m} \text{MaxSub}(m, s_j) \times \delta(m, s_j)}{|S| - 1} \quad (2)$$

Where  $m$  is a predicted structure,  $S$  is the set of all predicted structures to the same target protein.

**Data Normalization:** We use the 0-1 normalization method and two rank-based normalization methods at different stages to standardize the data. In both cases, the normalization methods we use are target-wise normalization. In other words, we standardize the values of each score with regard to the structures predicted to the same target protein only. Our target-wise normalization is different from the method implemented in [7]. Equation 3 shows how to normalize a score value  $v$  of a structure  $i$  which is predicted to target protein  $t$ .

$$\bar{v}_i^t = \frac{v_i^t - v_{min}^t}{v_{max}^t - v_{min}^t} \quad (3)$$

Where  $v_{max}^t$  and  $v_{min}^t$  are the maximum and minimum score values of all structures predicted to the same target protein.

### B. The Learning Framework

**The STP algorithm:** Sample-Train-Predict [22]. We apply the STP algorithm when the available data have two main properties. First, the available training (labeled) data is constantly growing. For example, the protein structure bank is increasing in size on a weekly basis. Second, the data is intrinsically clustered based on similarity in sequence, structure or function (each cluster has high-level semantic meaning). For instance, a set of predicted structures to the same target protein is viewed as a cluster in our current work. We have devised two variants of the STP algorithm: STPdata and STPmodel. We describe the application of the STPdata algorithm in this article.

Figure 1 outlines the STPdata algorithm. The STP algorithm does its prediction in a batch mode i.e. it takes a cluster of data of unknown target values as its input and outputs the results in a batch mode as well. The STP algorithm has three stages: (i) Sample: select a subset of the training data based on the similarity to the unlabeled data; we use the distribution of the 3dSim scores as the similarity measures between the input (test) cluster and the clusters stored in the database (ii) Train: train a hierarchy of general linear models on the sampled data (iii) Predict: use the trained hierarchical model to select and rank the best predicted structures. We regard the STPdata algorithm as a method to build a custom-trained expert designed specifically to the input target protein predictions.

**The STP Algorithm Input and Output:** input: a  $n \times m$  scores matrix  $S$ , such that  $s_{ij}$  is the  $j^{\text{th}}$  score value of the  $i^{\text{th}}$  predicted structure. The scores matrix  $S$  contains the MQAPs scores of the structures predicted to the same target protein. All scores are standardized between 0.0 and 1.0.

$$S = \begin{pmatrix} S_{11} & S_{12} & \cdot & \cdot & \cdot & S_{1m} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ S_{n1} & S_{n2} & \cdot & \cdot & \cdot & S_{nm} \end{pmatrix}$$

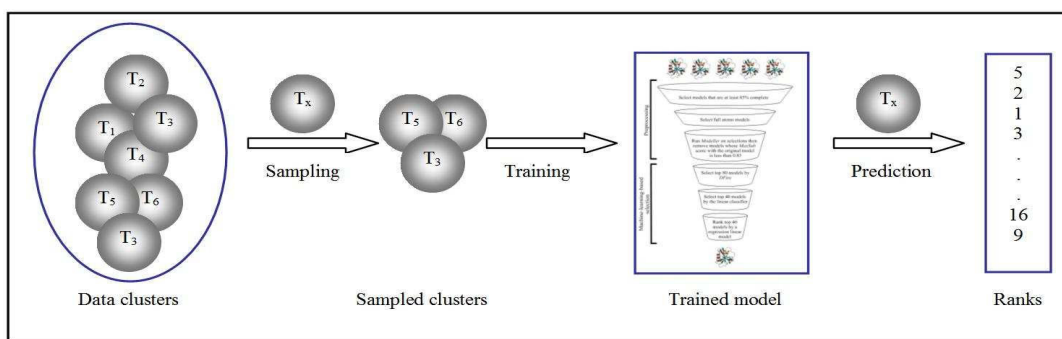


Fig. 1. The STP algorithm does its prediction in a batch mode i.e. it takes a cluster of data of unknown target values  $T_x$  as its input and outputs the results in a batch mode as well. The STP algorithm has three stages: (i) Sample: select a subset of the training data based on the similarity to the unlabeled data (ii) Train: train a hierarchy of general linear models on the sampled data (iii) Predict: use the trained hierarchical model to select and rank the best predicted structures.

Output: The predicted ranks of the best predicted structures.

**STPdata Sampling:** STPdata considers the MQAPs' scores of the structures predicted to the same target protein as one cluster. We represent each cluster by two centers of the bimodal distribution of the 3dSim scores and the percentages of the predicted structures that belong to each mode, in other words, four parameters represent each cluster: two parameters for centers and two parameters for percentages. In section IV, we give the reasons for using the 3dSim distribution as the similarity measure. We obtain the two centers by applying the k-means clustering algorithm with initial centers 0.0 and 1.0 to the 3dSim scores of each cluster. For example, the vector [0.8 0.1 0.6 0.4] means that the k-means algorithm found two centers at 0.8 and 0.1, and 60% of the predicted structure are clustered around the 0.8 center and the other 40% of the predicted structures are clustered around the 0.1 center. We represent the input cluster in a similar fashion. Then, we apply the k-nearest algorithm to the clusters representations to find the nearest 22 clusters to the input cluster. We have decided to use 22 clusters based on experimental results on the training and validation sets.

**STPdata Training:** in this stage the algorithm trains a custom-made hierarchy of general linear models (GLM) [23] specifically to the input cluster. The GLM for  $n$  observations can be expressed as

$$A = Sv + e \quad (4)$$

Where  $A$  is a matrix of target values,  $S$  is the  $n \times m$  MQAPs scores matrix as defined earlier,  $v$  is the  $m$ -dimensional regression coefficients vector, and  $e$  is the  $n$ -dimensional errors vector. Next we use the pseudo inverse solution to solve equation 4 for the weights vector  $w$ . Let  $w$  be a  $m + 1$  dimensional vector of 1 and  $m$  regression coefficients, then

$$Sw = A \quad (5)$$

$$w = (S^T S)^{-1} S^T A \quad (6)$$

**Hierarchy of General Linear Models:** the model consists of three levels. At the first level, DFire selects the top 80 predicted structures to eliminate low quality structures. At the second level, a linear classifier separates the top 40 structures

from the lower 40 structures. At the third level a linear regression model ranks the top 40 predicted structures. We have set these parameters to participate in CASP8 based on the number of the predicted structures per target protein in CASP6 and CASP7. However, these two thresholds should be increased when our method is applied to computational methods that produce a large number of predicted structures.

**The Linear Classifier:** the classifier learns to separate two classes: (i) the class of the top 40 ranks (1-40) and (ii) the class of the lower 40 ranks (41-80). We choose to make each class contains 40 structures to make sure that the classifier is not biased to any of the two classes. The STPdata algorithm trains the linear classifier on set  $D_1 = \{(x^1, a_1), \dots, (x^q, a_q)\}$ . Where  $q = 80 \times n$ , such that  $n$  is the number of the target proteins selected at the previous stage, input  $x^i$  is a  $m$ -dimensional vector representing the  $m$  scores of the structure  $i$ ,  $x^i = \{x_1, \dots, x_m\}$  where

$$x_j = \begin{cases} 1 & \text{if the } j^{\text{th}} \text{ MQAP rank is below 40;} \\ -1 & \text{o.w.;} \end{cases}$$

and  $a_i$  is the target to be learnt such that

$$a_i = \begin{cases} 1 & \text{if the MaxSub rank is below 40;} \\ -1 & \text{o.w.;} \end{cases}$$

For example, the pair ([1 1 -1 1 1], 1) means that the 1<sup>st</sup>, 2<sup>nd</sup>, 4<sup>th</sup>, and 5<sup>th</sup> MQAPs rank the structure within the top 40 ranks, and the 3<sup>rd</sup> MQAP ranks the structure within the lower 40 ranks. MaxSub ranks the predicted structure within the top 40 ranks.

**The Linear Regression Model:** the linear classifier in the previous stage selects 40 structures to pass to the third and final stage. A regression linear model is trained to predict the MaxSub rank of the structure. STPdata trains the linear regression model on set  $D_2 = \{(x^1, a_1), \dots, (x^k, a_k)\}$ . Where  $k = 40 \times n$ , such that  $n$  is the number of proteins sampled at the sampling step, input  $x^i$  is a  $m$ -dimensional vector representing the  $m$  MQAPs scores of a structure  $i$ .  $x^i = \{x_1, \dots, x_m\}$ , where  $x_j$  is the structure rank assigned by the  $j^{\text{th}}$  MQAP, and  $x_j \in \{1, 2, \dots, 39, 40\}$ . The output  $a_j$  is the rank assigned by MaxSub, and  $a_j \in \{1, 2, \dots, 39, 40\}$ . For example, the pair ([1 5 7 3 4], 3) means that the five MQAPs

rank the structure as the 1<sup>st</sup>, the 5<sup>th</sup>, the 7<sup>th</sup>, the 3<sup>rd</sup>, and the 4<sup>th</sup> structure, and MaxSub puts the structure on the 3<sup>rd</sup> rank.

**STPdata Prediction:** once the on-line custom-trained hierarchy of general linear models is trained on the related clusters to the input cluster, the STPdata algorithm outputs the predicted top 40 structures and their ranks.

### III. RESULTS

**CASP8 results:** the first author has participated with ZicoSTP<sup>2</sup> in CASP8 to evaluate its performance by independent assessors. The ZicoSTP database consists of the predicted structures by the servers and the human predictors in CASP6. Here we report the performance of our method on 71 domains from the Human/Server targets. These results are based on the official CASP8 results [12]. Figure 2 shows the improvement in performance of our method over the best five performing servers in CASP8 based on the GDT.TS score. ZicoSTP outperformed the first, second, and third performing servers by 6%, 27%, and 48% respectively.

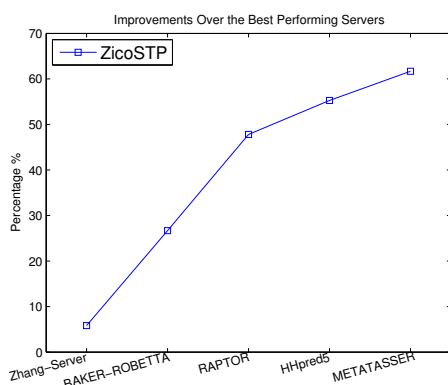


Fig. 2. The improvements of the ZicoSTP over the best performing servers in CASP8

**Evaluation on the CASP7 human predictors structures:** we have tested ZicoSTP to select and rank the best structures predicted by the human predictors in CASP7. All data are preprocessed in the same manner as in the training stage. ZicoSTP outperformed the best human predictor (Zhang) by 3.19% as shown in Table I. ZicoSTP outperformed the best tested model quality assessment program in our study (3dSim) by 8.03%. We measure the performance as the sum of the rank-one structures MaxSub scores.

TABLE I  
THE PERFORMANCE OF ZICOSTP ON THE HUMAN PREDICTORS  
STRUCTURES IN CASP7

Methods	Total
Best Human (Zhang)	47.76
Best MQAP (3dSim)	45.62
ZicoSTP	49.28
Improvement over Best Human	3.19%
Improvement over Best MQAP	8.03%

<sup>2</sup>The official name in CASP8 is ZicoFullSTP.

### IV. DISCUSSION

In this section we take a deeper look at the learnt function. We give the reasons to use the 3dSim distribution as a cluster similarity measure and we compare our method to other related work.

**The learnt functions:** The STPdata algorithm learns a hierarchical model for each target protein. Traditionally, there is only one learnt function used to rank the predicted structures of all target proteins [3], [5]–[7], [24]. In contrast, we use a custom-learnt function to rank the structures predicted to a specific target protein. The STPdata algorithm learnt 128 different hierarchical models for the 128 target proteins in CASP8. Figure 3 shows the weights associated with the ranks assigned by the 5 model quality assessment programs in the linear regression model (the third level). We did this analysis on target proteins from CASP8. These curves show how the learnt weights changes for each target protein. 3dSim ranks has the largest weights followed by Prosa-pair.

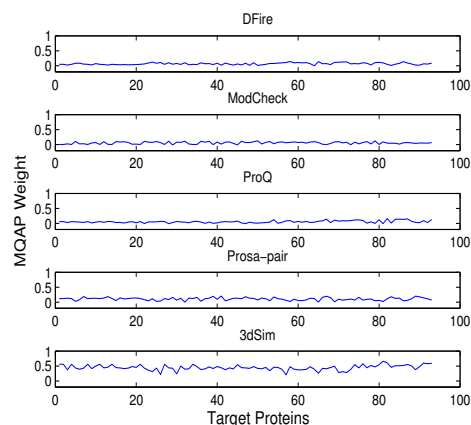


Fig. 3. The weights associated with the ranks assigned by the 5 model quality assessment programs in the linear regression model (the third level of the hierarchical model)

**The 3dSim Distribution as a Clusters Similarity Measure:** we specifically choose the distribution of the 3dSim scores as a similarity measure between the input cluster and the clusters stored in the training database for the following three reasons: (i) the 3dSim score is highly correlated with the MaxSub scores (the target to be learnt) (ii) the distributions of the MaxSub scores and the 3dSim scores are bimodal as shown in figure 4 (iii) the distribution of the 3dSim scores has one of the lowest KL divergence from the distribution of the MaxSub scores indicating the similarity between these two distributions.

**Comparison to The Related Work:** we compare our work to the other machine learning based model quality assessment programs, namely: ProQ [3], SVMOD [6], and SVR [7]. The following seven aspects highlight the differences and the similarities between our work and the related work: (i) the framework learns a unique function per target protein; however, the other methods learn only one function for all target proteins (ii) ZicoSTP uses a subset of proteins

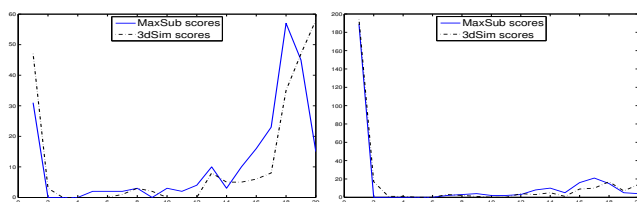


Fig. 4. The distributions of the MaxSub scores and the 3dSim scores

which are related to the input target protein to build on-line a custom-trained hierarchy of GLM's. On the other hand, ProQ, SVMMod, and SVR use all available training data (iii) ProQ and SVMMod do not distinguish between the high quality structures and the low quality ones during training or testing. SVR assigns higher penalties to errors in ranking the high quality structures during training. ZicoSTP contains a linear classifier to separates the high quality structures from the low quality ones during training and testing (iv) ProQ, SVMMod, and SVR rank the predicted structures in one pass; however, our system ranks the predicted structures in two passes (v) the developers of the related MQAP's formulated the task as a regression problem. In our work, we formulate the problem as a classification task followed by a regression task (vi) SVMMod, SVR, and ZicoSTP are based on linear models, namely: SVM (with linear kernel) and GLM. ProQ is based on a multi-layer artificial neural network which is a non-linear model (vii) ProQ predicts the MaxSub and the LGscore scores. SVMMod predicts the RMSD scores. SVR predicts the GDT-TS scores, and our system predicts the MaxSub ranks (viii) SVR uses a target-wise normalization in which there is a specific feature to represent the target identity. We have also used a target-wise normalization; however, our method is different from the one used in the SVR. We did not include an extra feature to describe the target identity. We used two method of normalization at different stages of the STP algorithm: (a) the 0-1 normalization applied to the structures predicted to the same target protein only (b) the rank-based normalization applied to the structures predicted to the same target protein only.

## V. CONCLUSION

In this article we presented an on-line hierarchy of general linear models to select and rank the best predicted structures. Our work has two main contributions. First, the application of an on-line model enables us to have a custom-trained scoring function trained specifically for a certain target protein. Second, the application of a hierarchical model provides a way to distinguish the high quality structures from the low quality structures during both training and testing. Each level of the hierarchical model is a linear combination of the ranks assigned by five model quality assessment programs. To evaluate our method we participated in CASP8 as human predictors. Based on the official results of CASP8 our method outperformed the best server by 6% and won the fourth position among human predictors.

## REFERENCES

- [1] C. Branden and J. Tooze, *Introduction to Protein Structure*. New York: Garland Publishing, Inc, 1999.
- [2] I. G. Choi and S. H. Kim, "Evolution of protein structural classes and protein sequence families," *The Proceedings of the National Academy of Sciences U.S.A.*, vol. 103, no. 38, pp. 14 056–14 061, 2006.
- [3] B. Wallner and A. Elofsson, "Can correct protein models be identified?" *Protein Science*, vol. 12, pp. 1073–1086, 2003.
- [4] J. Xu, L. Yu, and M. Li, "Consensus fold recognition by predicted model quality," in *Proceedings of the third Asia-Pacific Bioinformatics Conference*, 2005, pp. 73–84.
- [5] B. Wallner and A. Elofsson, "Pcons5: combining consensus, structural evaluation and fold recognition scores," *Bioinformatics*, vol. 21, no. 23, pp. 4248–4254, 2005.
- [6] D. Eramian, M. Shen, D. Devos, F. Melo, A. Sali, and M. Marti-Renom, "A composite score for predicting errors in protein structure models," *Protein Science*, vol. 15, pp. 1653–1666, 2006.
- [7] J. Qiu, W. Sheffler, D. Baker, and W. S. Noble, "Ranking predicted protein structures with support vector regression," *Proteins: Structure, Function, and Bioinformatics*, vol. 71, pp. 1175–1182, 2007.
- [8] E. A. Coutsiyas, S. Chaok, and K. A. Dill, "Using quaternions to calculate rmsd," *Journal of Computational Chemistry*, vol. 25, no. 15, pp. 1849–1857, 2004.
- [9] N. Siew, A. Elofsson, L. Rychlewski, and D. Fischer, "Maxsub: an automated measure for the assessment of protein structure prediction quality," *Bioinformatics*, vol. 16, pp. 776–785, 2000.
- [10] Y. Zhang and J. Skolnick, "Scoring function for automated assessment of protein structure template quality," *Proteins*, vol. 57, p. 702710, 2004.
- [11] A. Zemla, C. Venclovas, J. Moult, and K. Fidelis, "Processing and analysis of casp3 protein structure predictions," *PROTEINS: Structure, Function and Genetics Supplement*, vol. 3, pp. 22–29, 1999.
- [12] Casp8 group performance. [Online]. Available: [http://predictioncenter.org/casp8/groups\\_analysis.cgi?target\\_type=1&gr\\_type=all&domain\\_classifications\\_id=1,2,3,4](http://predictioncenter.org/casp8/groups_analysis.cgi?target_type=1&gr_type=all&domain_classifications_id=1,2,3,4)
- [13] J. Moult, K. Fidelis, B. Rost, T. Hubbard, and A. Tramontano, "Critical assessment of methods of protein structure prediction (casp) - round 6," *Proteins*, vol. 61, no. S7, pp. 3–7, 2005.
- [14] J. Moult, K. Fidelis, A. Kryshchuk, B. Rost, T. Hubbard, and A. Tramontano, "Critical assessment of methods of protein structure prediction - round vii," *Proteins: Structure, Function, and Bioinformatics*, vol. 69, no. S8, pp. 3–9, 2007.
- [15] J. Bujnicki and D. Fischer, "Meta approaches to protein structure prediction," in *Practical Bioinformatics*, J. M. Bujnicki, Ed. Berlin Heidelberg: Springer-Verlag, 2004, vol. 15, pp. 23–34.
- [16] A. Sali and T. Blundell, "Comparative protein modelling by satisfaction of spatial restraints," *Journal of Molecular Biology*, vol. 234, pp. 779–815, 1993.
- [17] M. J. Sippl, "Boltzmann's principle, knowledge-based mean fields and protein folding. an approach to the computational determination of protein structures," *J. Comput. Aided Mol. Des.*, vol. 7, pp. 473–501, 1993.
- [18] —, "Knowledge-based potentials for proteins," *Curr. Opin. Struct. Biol.*, vol. 5, pp. 229–235, 1995.
- [19] D. Jones, "Genthrader: An efficient and reliable protein fold recognition method for genomic sequences," *Journal of Molecular Biology*, vol. 287, pp. 797–815, 1999.
- [20] H. Zhou and Y. Zhou, "Distance-scaled, finite ideal-gas reference state improves structure-derived potentials of mean force for structure selection and stability prediction," *Protein Science*, vol. 11, pp. 2714–2726, 2002.
- [21] K. Ginalski, A. Elofsson, D. Fischer, and L. Rychlewski, "3d-jury: a simple approach to improve protein structure predictions," *Bioinformatics*, vol. 19, pp. 1015–1018, 2003.
- [22] H. Z. Girgis and J. J. Corso, "Stp: the sample-train-predict algorithm and its application to protein structure meta-selection," The State University of New York at Buffalo, Technical Report 2008-16, 2008.
- [23] T. Hill and P. Lewicki, *Statistics Methods and Applications*. Tulsa, OK: StatSoft, 2007.
- [24] J. Lundström, L. Rychlewski, J. M. Bujnicki, and A. Elofsson, "Pcons: A neural-network based consensus predictor that improves fold recognition," *Protein Science*, vol. 10, pp. 2354–2362, 2001.