

**GENOME DATA ANALYSIS, PROTEIN FUNCTION
AND STRUCTURE PREDICTION
BY MACHINE LEARNING TECHNIQUES**

A Dissertation
presented to
the Faculty of the Graduate School
at the University of Missouri

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

by
RENZHI CAO
Professor Jianlin Cheng, Dissertation Supervisor

JULY 2016

The undersigned, appointed by the Dean of the Graduate School, have examined the dissertation entitled:

GENOME DATA ANALYSIS, PROTEIN FUNCTION AND STRUCTURE
PREDICTION BY MACHINE LEARNING TECHNIQUES

presented by Renzhi Cao,
a candidate for the degree of Doctor of Philosophy and hereby certify that, in their opinion, it is worthy of acceptance.

Professor Jianlin Cheng

Professor Shijie Chen

Professor Ye Duan

Professor William Harrison

DEDICATION

To my parents: Yingen and Meizhen,
who made all of this possible by their endless encouragement and patience.

To my wife Wen Wang and my son Yufan Cao.
who give all of their support and bring happiness to me during my tenure as doctoral student.

*If you can keep your head when all about you
Are losing theirs and blaming it on you,
If you can trust yourself when all men doubt you,
But make allowance for their doubting too;
If you can wait and not be tired by waiting,
Or being lied about, don't deal in lies,
Or being hated, don't give way to hating,*

*And yet don't look too good, nor talk too wise.
If you can talk with crowds and keep your virtue,
Or walk with Kings—nor lose the common touch,
If neither foes nor loving friends can hurt you,
If all men count with you, but none too much;
If you can fill the unforgiving minute
With sixty seconds' worth of distance run,
Yours is the Earth and everything that's in it,
And—which is more—you'll be a Man, my son.*

- Rudyard Kipling, Rewards and Fairies (1910)

ACKNOWLEDGMENTS

First of all, I would like to give my appreciation to my long-time advisor and committee chair Dr. Jianlin Cheng. He guides me to this interesting Ph.D research, and gives me a lot of good suggestions and supports. Without his kindness and considerable mentoring, I could not finish my doctoral work.

Second, I am very lucky to have the opportunity to work with people in the lab who are better than me, and drift me to the right direction. I appreciate all people currently in the lab, including: Debswapna Bhattacharya, Jie Hou, Badri Adhikari, Tuan Anh Trieu, Oluwatosin Oluwadare, and also people who graduate before me, including: Dr. Zheng Wang, Dr. Xin Deng, and Dr. Jesse Eickholt.

Third, I want to appreciate my colleagues at Samsung Research America who bring me an extraordinary experience, including: Varun Shimoga Prakash, William Reginald Swaney, Chongyang Xie, Victor Borodkin, Naman Patel, Louisa Toy-Wong, Shuo Wang, Haiqing Jiang, and Xinwen Zhang.

Finally, I would like to thank my committee members for their supports and suggestions, including: Dr. Shijie Chen, Dr. Ye Duan, and Dr. William Harrison.

TABLE OF CONTENTS

ACKNOWLEDGMENTS	ii
LIST OF TABLES	viii
LIST OF FIGURES	xii
ABSTRACT	xvii
CHAPTER	
1 Introduction	1
1.1 Genome data analysis	2
1.2 Genome data analysis protein function prediction	2
1.3 Protein structure prediction	3
2 Deciphering the association between gene function and spatial gene-gene interactions in 3D human genome conformation	6
2.1 Abstract	6
2.2 Introduction	7
2.3 Methods	9
2.3.1 Calculation of gene function similarity between two genes	9
2.3.2 Construction of genome-wide spatial gene-gene interaction networks	9
2.3.3 Calculation of sequence identity	10
2.3.4 Gene function prediction based on spatial gene-gene interaction networks	10
2.4 Results and Discussion	11
2.4.1 The spatial gene-gene interaction network for whole genome and thresholds for substantially interacting gene pairs	11

2.4.2	The function similarity of gene pairs that do not spatially interact and that have substantial interactions	14
2.4.3	The statistics of the number of interactions for substantially interacting gene pairs at each function similarity level	14
2.4.4	The sequential genomic distance for substantially interacting gene pairs at each function similarity level	16
2.4.5	Sequence identity of substantially interacting genes at each function similarity level	17
2.4.6	Identification of interacting genes with high function similarity with sequence identity, genomic distance, and interaction strength	19
2.4.7	The relationship between sequence identity and function similarity for substantially interacting gene pairs and random non-interacting gene pairs	20
2.4.8	The relationship among genomic distance, interaction numbers, and function similarity for interacting gene pairs	23
2.4.9	Evaluation of gene function predictions based on spatial gene-gene interactions	24
3	Integrated protein function prediction by mining function associations, sequences, and protein-protein and gene-gene interaction networks	28
3.1	Abstract	28
3.2	Introduction	29
3.3	Methods	32
3.3.1	NET score	36
3.3.2	SEQ score	37
3.3.3	Score combination	39
3.3.4	Score scaling	39
3.4	Results and discussion	40
3.4.1	Parameters in Apriori algorithm for calculating MIS score	40

3.4.2	Prediction Performance	42
3.4.3	Case study	44
3.5	Conclusion	48
4	Designing and evaluating the MULTICOM protein local and global model quality prediction methods in the CASP10 experiment . .	49
4.1	Abstract	49
4.2	Introduction	50
4.3	Methods	53
4.3.1	Protein Model Quality Prediction Methods	53
4.3.2	Evaluation Methods	54
4.4	Results and Discussions	56
4.4.1	Results of global quality predictions	56
4.4.2	Results of local quality	65
5	Single-model quality assessment on the assessment of scores from probability density function	72
5.1	Abstract	72
5.2	Introduction	73
5.3	Methods	75
5.3.1	Feature generation	75
5.3.2	Feature errors estimation	78
5.3.3	Feature weight estimation	78
5.3.4	Model quality assessment based on probability density function	79
5.4	Results	80
5.4.1	Feature normalization result	80
5.4.2	Feature error estimation result	84

5.4.3	Global quality assessment result	84
5.5	Discussion	89
6	DeepQA: Improving the estimation of single protein model quality with deep belief networks	91
6.1	Abstract	91
6.2	Introduction	92
6.3	Methods	94
6.3.1	Datasets	94
6.3.2	Input features for DeepQA	95
6.3.3	Deep belief network architectures and training procedure	97
6.3.4	Model accuracy evaluation metrics	98
6.4	Results and Discussion	99
6.4.1	Comparison of Deep learning with support vector machines and neural networks	99
6.4.2	Comparison of DeepQA with other single-model QA methods on CASP11	101
6.4.3	Case study of DeepQA on <i>ab initio</i> datasets	102
6.5	Conclusions	103
7	Large-Scale Model Quality Assessment for Improving Protein Tertiary Structure Prediction	104
7.1	Abstract	104
7.2	Introduction	105
7.3	Methods	108
7.3.1	Large-scale protein model quality assessment for protein tertiary structure prediction	108
7.3.2	Evaluation of top ranked models	111
7.4	Results and Discussions	113

7.5	Conclusions	122
8	Massive integration of diverse protein quality assessment methods to improve template based modeling in CASP11	124
8.1	Abstract	124
8.2	Introduction	125
8.3	Methods	128
8.3.1	Massive protein model quality assessment for ranking protein structural models	128
8.3.2	Summary of some individual QA methods used by MULTICOM	134
8.3.3	Evaluation	136
8.4	Results and discussions	137
8.5	Conclusions	149
	BIBLIOGRAPHY	150
	VITA	169

LIST OF TABLES

Table	Page
2.1 Contact thresholds and the corresponding numbers of interacted genes for the spatial gene-gene interaction networks constructed for four cells / cell lines.	13
3.1 The precision, recall, and multiplication of precision and recall for different values of minimum support and confidence according to five-fold cross validation.	41
3.2 Summary of PDB ids with their true functions and the protein function predictions by our methods for case study.	47
4.1 The average correlation (Ave. Corr.), overall correlation (Over. Corr.), average GDT-TS loss (Ave. loss), average Spearman’s correlation (Ave. spearman), average Kendall tau correlation (Ave. Kendall) of MULTICOM servers, DAVIS-QAconsensus, Pcons, and ModFOLD-clust2 on Stage1 of CASP10.	58
4.2 The average correlation, overall correlation, average GDT-TS loss, average Spearman’s correlation, average Kendall tau correlation of MULTICOM servers, DAVIS-QAconsensus, Pcons, and ModFOLD-clust2 on Stage2 of CASP10.	59

4.3	The P-value of pairwise wilcoxon signed ranked sum test for the difference of correlation score between MULTICOM servers on Stage1 and Stage2 of CASP10, and three other methods: DAVIS-QAconsensus, Pcons, and ModFOLDclust2.	60
4.4	Pearson correlation of the FM (template-free modeling) targets on Stage1 of CASP10.	65
4.5	Pearson correlation of all FM (template-free modeling) targets on Stage2 of CASP10.	66
4.6	Evaluation result of local quality score of four servers, DAVIS- QAconsensus, Pcons, and ModFOLDclust2 on Stage1 and Stage2 of CASP10.	69
4.7	The P-value of pairwise wilcoxon signed ranked sum test for the difference of correlation score for local model quality between MULTICOM servers on Stage1 and Stage2 of CASP10, and three other methods: DAVIS-QAconsensus, Pcons, and ModFOLDclust2.	70
4.8	Local quality score of four servers, DAVIS-QAconsensus, Pcons, and ModFOLDclust2 for all FM (template-free modeling) targets on Stage1 of CASP10.	71
4.9	Local quality score of four servers, DAVIS-QAconsensus, Pcons, and ModFOLDclust2 for all FM (template-free modeling) targets on Stage2 of CASP10.	71
5.1	The per-target average correlation, average loss, average spearman, and average kendall tau score of our method Qprob and other pure single-model QA methods on sel20 CASP11 dataset. The p-value of pairwise Wilcoxon signed ranked sum test for the difference of loss and correlation of Qprob against other methods is listed for comparison. Five single-model QA methods which don't attend CASP11 are also listed and highlighted in bold.	86

5.2	The per-target average correlation, average loss, average spearman, and average kendall tau score for our method Qprob and several other pure single-model QA methods on top150 CASP11 dataset. The p-value of pairwise Wilcoxon signed ranked sum test for the difference of loss and correlation of Qprob against other methods is listed for comparison. Five single-model QA methods which don't attend CASP11 are also listed and highlighted in bold.	87
6.1	16 features for benchmarking DeepQA.	96
6.2	The accuracy of Deep Belief Network, Support Vector Machines, and Neural Networks in terms of MAE based on cross validation of training datasets, the average per-target correlation, and loss on stage 1 and stage 2 of CASP11 datasets for all three difference techniques.	100
6.3	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 2 of CASP11.	102
6.4	Model selection ability on <i>ab initio</i> datasets for DeepQA, ProQ2, Dope2, and RWplus score	103
7.1	All 14 QA methods with the details. The highlighted methods are built in house. S: single-model method; M: multi-model method.	108
7.2	The top 10 tertiary structure predictors ranked based on the summation of the Z-scores of the first models, and their summation of the Z-scores of best of the five submitted models.	114
7.3	The top 10 predictors ranked based on the total number times their models were selected by our MULTICOM predictor on all the human targets or template-based (TBM) human targets only.	114

7.4	Comparison of MULTICOM with each QA method and the two different consensus methods (one based on 6 QA methods and another one based on 14 QA methods) on the average GDT-TS score and Z-score of the top models selected, and the significance of difference between each QA method and MULTICOM. <i>Italic font denotes single-model methods.</i>	115
7.5	The total number times that each QA method performed better than other QA methods on all human targets or all template based (TBM) human targets only. <i>Italic denotes single-model methods.</i>	120
8.1	Publicly available single-model QA methods used in our MULTICOM method.	130
8.2	The average scores of the first models submitted by MULTICOM (bold) and top 25 performing server predictors.	138
8.3	The average scores of the best of top five models submitted by MULTICOM (bold) and top 25 performing server predictors.	139
8.4	Comparison of MULTICOM with each QA method on the average GDT-HA score and Z-score of the top models selected, and the significant of each QA method.	147

LIST OF FIGURES

Figure	Page	
2.1	Visualization of gene-gene interaction network. Figure 2.1A is the plot of the numbers of interacted genes against interaction / contact thresholds for four cells / cell lines respectively. X-axis denotes the interaction thresholds and Y-axis the numbers of interacted genes found at the thresholds. Figure 2.1B is the visualization of the largest cluster of the gene-gene interaction network for the Call4 cell line at interaction threshold 16. The network was visualized by Cytoscape [40].	13
2.2	The histograms of gene function similarities of non-interacted gene pairs and substantially interacted gene pairs. Figure 2.2A,2.2B, and 2.2C represent the histogram for Biological Process, Cellular Component, and Molecular Function respectively.	15
2.3	The average number of interactions between substantially interacted gene pairs within each functional similarity bin in three function categories. This is for the primary tumor B-cells (ALL). Figure 2.3A, 2.3B, and 2.3C represent the histogram for Biological Process, Cellular Component, and Molecular Function respectively.	16

2.4	The average genomic distances of substantially interacted gene pairs in each functional similarity bin in three function categories. This is for the primary tumor B-cells (ALL). Figure 2.4A, 2.4B, and 2.4C represent the histogram for Biological Process, Cellular Component, and Molecular Function respectively.	18
2.5	The boxplot of gene sequence identity against function similarity in three GO categories. Figure 2.5A,2.5B, and 2.5C represent the histogram for Biological Process, Cellular Component, and Molecular Function respectively. This figure is generated on the gene-gene interaction network of the ALL B-cell constructed at interaction threshold 18. X-axis denotes the functional similarity scores / bins and Y-axis gene sequence identity.	19
2.6	Plot of function similarity against sequence identify for substantially interacted gene pairs and non-interacted gene pairs. X-axis denotes the gene sequence identity and Y-axis the gene function similarity in all three categories (BP, CC, MF), respectively.	21
2.7	The sequence identity and the number of gene interactions. The number of interactions is normalized to the range of 0 to 1. The result is generated on the ALL gene-gene network with ≥ 1 interactions. X-axis denotes the sequence identity and Y-axis the normalized number of interactions.	22
2.8	The 3D plot of genomic distance, number of interactions and the function similarity in three function categories. Figure 2.8A, 2.8B, and 2.8C represent the histogram for Biological Process, Cellular Component, and Molecular Function respectively. The yellow dots represent long genomic distances and the red ones the opposite.	24

2.9	The histograms of function prediction accuracy (the maximum similarity scores between predicted GO terms and real GO terms) on the spatial gene-gene interaction networks of the Call4 cell line at different interaction thresholds	26
2.10	The histograms of function prediction accuracies for different numbers (1 – 10) of GO terms selected as predictions.	27
3.1	The overall flowchart of our method.	33
3.2	The performance comparison for MIS, SEQ, and SMISS using scaled technique benchmarked on CAFA1. X-axis shows the recall of the prediction, and y-axis shows the precision of the predictione2	44
3.3	The performance of our SMISS with three standard baseline method and three predictors from an automated three-level method. Prediction 57, 58, 59 is the standard baseline method, and Predictors 1, 2, 3 is three predictors from an automated three-level method. X-axis shows the recall for each predictor, and y-axis shows the precision for each predictor.	45
4.1	The per-target correlation scores of each target against the average real quality of the largest model cluster divided by the average real quality of all models in this target on Stage2	61
4.2	The influence of side chain on average correlation and loss of both Stage1 and Stage2. Figure 4.2A shows the average correlation of the predictions with or without side-chain repacking, and Figure 4.2B demonstrates the loss of the predictions with or without side-chain repacking on both Stage1 and Stage2. The tool SCWRL[1] is used for the side-chain repacking.	63

4.3	The hierarchy tree of T0741 on Stage1. All models in the circle form the largest cluster in this target. The rightmost column of Figure 3 lists the real GDT-TS score of each model. The models in the circle form the largest cluster. The model with the underline real GDT-TS score is the best model in this target	66
4.4	The real GDT-TS score and predicted GDT-TS score of MULTICOM-REFINE and MULTICOM-NOVEL for T0684 on Stage 1 and Stage2.	67
5.1	The relationship of sequence length and three energy scores (DFIRE2, RWplus, and RF_CB_SRS_OD scores) on PISCES database.	82
5.2	The probability density distribution for the error estimation of all 11 feature scores.	83
5.3	The summation of Z-score for the top 1 model selected by each method	89
6.1	The Deep Belief Network architecture for DeepQA.	98
7.1	The workflow of the MULTICOM method comprised of six steps. . .	112
7.2	Tertiary structure prediction of domain 2 of T0783 (T0783-D2). (A) The superposition of the MULTICOM human TS1 model on domain 2 with the native structure. (B). The distribution of 191 models in the model pool. (C). The plot of the true GDT-TS scores of models against their predicted ranking.	119
7.3	Tertiary structure prediction of domain 1 of T0767 (T0767-D1). (A) The superposition of the MULTICOM human TS1 model on domain 1 with the native structure. (B). The distribution of 195 models in the model pool. (C). The plot of the true GDT-TS scores of models against their predicted ranking.	121

7.4	The plot of the difference between the initial GDT-TS scores before model combination and the GDT-TS scores after model combination against the initial GDT-TS scores of top one models of 42 targets . . .	122
8.1	Workflow of MULTICOM large-scale model quality assessment method.	131
8.2	Performance of MULTICOM and server predictors with respect to number of residues in domain	140
8.3	Performance of MULTICOM and server predictors with respect to difficulty of target	141
8.4	Accuracy of MULTICOM compared to other server predictors	142
8.5	Case study for CASP11 targets T0853-D1 and T0830-D1.	145
8.6	Comparison of MULTICOM with individual QA methods.	146
8.7	Landscape of MULTICOM's ranking.	148

ABSTRACT

The raw information of a typical human genome has been generated at 2001 by Human Genome Project. However, since there are a huge amount of data, it is still a big challenge for people to understand them, and extract useful structure and function information, such as the function of genes, the structure of proteins encoded by gene, and the function of proteins. Understanding these information is crucial for us to improve longevity and quality of life, and has a lot of applications, such as genomic medicine, drug design, and etc. In the meantime, machine learning techniques are growing rapidly and are good at processing large datasets, but many of them are limited for the impact on larger real world problems.

In this thesis, three major contributions are described. First of all, we generate gene-gene interaction network from human genome conformation data by Hi-C technique, and the relationship of gene function and gene-gene interaction has been discovered. Second, we introduce a novel framework SMISS, which uses new source of information from gene-gene interaction network and uses a new way to integrate difference sources of information for protein function prediction. Finally, we introduce a tool called DeepQA which use machine learning technique to evaluate how well is the predicted protein structure, and a method MULTICOM for protein structure prediction. All of these protein structure and function prediction methods are available as software and web servers which are freely available to the scientific communities.

Chapter 1

Introduction

Genome inside a cell is consist of double-stranded DNA sequences, and some special region of DNA sequences, so called protein-coding gene, can be encoded to proteins, which build the foundation of an organism. Understanding genome information is crucial for longevity and quality of life. By 2001, more than 90 percent of human genome sequence has been released [2], and the price of genome sequencing for a person is decreasing every year from more than billion dollars to less than thousand dollars now. We are in the personal genomics era because of technology development of genome sequencing. Similarly, with the wide application high-throughput next-generation sequencing technologies [3], a large number of proteins have been sequenced during the last decades. However, it is still a big challenge to determine, and understand the structural and function information of genome and protein, since the total amount of data for genome and protein is huge and not straight forward for people to visualize and understand. At the same time, machine learning techniques and data mining techniques are very good at processing large data and discover patterns, which powers many aspects of modern society from speech recognition to web searches, especially deep learning techniques [4]. There are a lot of promising applications for using machine learning techniques to interpret genome, and to predict

protein function and structures, such as genomic medicine, drug design, and etc. [5].

In this dissertation, I mainly focus on my research in genome data analysis, protein function and structure prediction.

1.1 Genome data analysis

As more and more genomes have been sequenced, it is important to annotate and analyze structure and function information of genome, such as gene function. Because of the complexity of human genome, its three dimensional structure is still not determined. However, the Hi-C technique invented in 2009 [6] can be used to determine genome-wide chromosomal interaction data. Based on this technique, we generate spatial gene-gene interacting network and investigate whether spatially interacting genes tend to share similar function. The genomic distance and sequence identity have also been considered for analysis of gene pairs. In addition, we introduce a gene function prediction method based on gene-gene interacting network generated from genome data by Hi-C technique, and the accuracy of this method is high based on our benchmark on a large number of genes. Chapter 2 of this dissertation mainly describe the above-mentioned research, which is published in BMC genomic journals:

R. Cao, J. Cheng. (2015). Deciphering the association between gene function and spatial gene-gene interactions in 3D human genome conformation. BMC genomics, 16:880. [2015 Impact factor 3.986]. [7]

1.2 Genome data analysis protein function prediction

Protein function is important for understanding life at molecular level. However, experimental methods that annotate protein function is still quite expensive, and

also not easy, even impossible some times because of the limitation of experimental method. Meanwhile, computational method for protein function prediction is fast and relatively cheap compared with experimental method. One of the major problems for protein function prediction is how to find new sources and integrate multiple sources of information to improve the accuracy of protein function prediction. With the help of The Critical Assessment of Function Annotation (CAFA) [8], which is an experiment designed for automated protein function annotation and blindly assess the progress of method development for protein function annotation, a large number of protein function prediction methods have been developed and blindly benchmarked. I develop a novel Statistical Multiple Integrative Scoring System (SMISS) for protein function prediction. It integrates the information based on probabilistic theory, including homologs found by PSIBLAST, protein-protein interaction networks, spatial gene-gene interaction networks derived from data by Hi-C technique described by previous section, and amino acid sequence information. This method is benchmarked and blindly tested on CAFA experiment and successfully predicts high accuracy protein functions. Chapter 3 describes details of this protein function prediction system SMISS, and it is mainly from the content of published paper as follows:

R. Cao, J. Cheng. (2016). Integrated protein function prediction by mining function associations, sequences, and protein-protein and gene-gene interaction networks. Methods, 93:84-91. [2015 Impact factor: 3.645]. [9]

1.3 Protein structure prediction

Protein structure determines protein function, and plays an important role in our life. The traditional experimental techniques (e.g, X-ray crystallography and Nuclear magnetic resonance spectroscopy) to determine protein structure is time consuming and expensive, sometime even cannot determine the structure. Because of that and widely

use of next-generation sequencing techniques, the gap between sequenced protein and its native structure is still enlarging. This highlights the importance of computational methods for protein structure prediction. There is a worldwide experiment called Critical Assessment of Techniques for Protein Structure Prediction (CASP) that blindly assesses protein structure prediction methods every two years from 1994. During protein structure prediction, there are two major problems: model sampling and model ranking. The former problem is about how to generate a number of structural models. The latter problem is about how to select and rank predicted structural models without knowing the native structure, so-called protein model quality assessment. My dissertation mainly focuses on solving the latter problem. For the latter problem, there are generally two kinds of methods to solve it. The first is single-model quality assessment method [9-15], which evaluates the quality of protein model without using other model's information. The second is consensus method [16-18], which uses the structural similarity between one model and other models to evaluate the quality of this model. Chapter 4 describes four single-model and consensus quality assessment methods, and compares the performance and characteristics of these two methods. The main content is coming from the following publication:

R. Cao, Z. Wang, J. Cheng. (2014). *Designing and evaluating the MULTICOM protein local and global model quality prediction methods in the CASP10 experiment.* *BMC Structural Biology*, 14:13. [2014 Impact factor: 1.18].[10]

Chapter 5 describes a novel single-model quality assessment method Qprob, which calculates protein model quality score based on probability density distribution of 11 features. Qprob is blindly tested on CASP11 and ranked as one of the top single-model quality assessment methods. The main content of this chapter is from the following publication:

R. Cao, J. Cheng. (2016). *Protein single-model quality assessment by feature-based probability density functions.* *Scientific Reports*, 6:23990, 2016. [2015 Impact

factor: 5.228].[11]

Chapter 6 describes a novel single-model quality assessment method DeepQA, which utilizing 16 features describing the quality of a model from different perspectives, and use deep learning techniques for protein quality assessment. The main content of this chapter is from unpublished manuscript:

R. Cao, D. Bhattacharya, J. Hou, J. Cheng. *DeepQA: Improving the estimation of single protein model quality with deep belief networks. arXiv preprint arXiv:1607.04379 (2016).*

Chapter 7 focus on protein structure prediction method blindly tested on CASP11 as MULTICOM human group which uses a large-scale model quality assessment method. It was officially ranked third out of all 143 human and server predictors on CASP11. The main content of this chapter comes from the following publication:

R. Cao, D. Bhattacharya, B. Adhikari, J. Li, J. Cheng. (2015). *Large-Scale Model Quality Assessment for Improving Protein Tertiary Structure Prediction. 23rd International Conference on Intelligent Systems for Molecular Biology (ISMB), Bioinformatics. 31(12):i116-i123. [2015 Impact factor: 4.98].[12]*

Chapter 8 describes the performance and analysis of our human tertiary structure predictor (MULTICOM) based on the massive integration of 14 diverse complementary quality assessment methods that was successfully benchmarked in the 11th Critical Assessment of Techniques of Protein Structure prediction (CASP11). The main content is from the following publication:

R. Cao, Bhattacharya, B. Adhikari, J. Li, J. Cheng. (2015). *Massive integration of diverse protein quality assessment methods to improve template based modeling in CASP11. Proteins: Structure, Function, and Bioinformatics. DOI: 10.1002/prot.24924. [2015 Impact factor: 2.63].[13]*

Chapter 2

Deciphering the association between gene function and spatial gene-gene interactions in 3D human genome conformation

2.1 Abstract

A number of factors have been investigated in the context of gene function prediction and analysis, such as sequence identity, gene expressions, and etc. However, three-dimensional (3D) conformation of the genome has not been tapped to analyse gene function, probably largely due to lack of genome conformation data until recently. We constructed the genome-wide spatial gene-gene interaction networks for three different human B-cells or cell lines from their chromosomal contact data generated by the Hi-C chromosome conformation capturing technique. We compared the function similarity of gene pairs that do not spatially interact and that have interactions. We found that genes that have strong spatial interactions tend to have highly similar function in terms of biological process, molecular function and cellular component of

the Gene Ontology. And even though the level of gene-gene interactions generally has no or weak correlation with either sequential genomic distance or sequence identity between genes, the interacted genes with high function similarity tend to have stronger interactions, somewhat shorter genomic distance and significantly higher sequence identity. And combining genomic distance or sequence identity with spatial gene-gene interaction information informs gene-gene function similarity much better than using either one of them alone, suggesting gene-gene interaction information is largely complementary with genomic distance and sequence identity in the context of gene function analysis. We developed and evaluated a new gene function prediction method based on gene-gene interacting networks, which can predict gene function well for a large number of human genes.

2.2 Introduction

As more and more genomes are sequenced, one urgent and important task in computational biology is to annotate and analyse the functions of the genes in a genome [14, 15]. A number of factors potentially related to gene function such as sequence identity, gene phylogenetic profiles, sequential genomic co-localizations, gene expressions, and protein-protein interaction have been investigated in the context of gene function prediction and analysis [16, 17, 18, 19, 20, 21]. However, another very important aspect of a genome, i.e. three-dimensional (3D) conformation of the genome, which presumably plays an important role in organizing and regulating genes, has not been tapped to analyse gene function, probably largely due to lack of genome conformation data until recently.

Since the Hi-C technique [6] that can determine the genome-wide chromosomal interaction / contact data was invented in 2009, it has been applied to generate the large-scale genome-wide chromosomal conformation data for a number of genomes

such as human B-cells [22, 23], yeast [24], bacteria [25], and Arabidopsis [26], which provides valuable data for studying the relationships between spatial gene-gene interactions and gene function. Similar technique has also been applied to study the three-dimensional model of budding yeast and other species [27, 28].

In this work, we analysed the intra- and inter-chromosomal interaction (contact) data of three different human malignant B-cell or cell lines (RL follicular lymphoma cell line (RL), primary tumor B-cells from an acute lymphoblastic leukaemia patient (ALL), and MHH-CALL-4 B-acute lymphoblastic leukaemia cell line (Call4)) [22] and one normal B-cell [6] captured by the Hi-C technique. From the Hi-C contact data, we generated the spatial gene-gene interactions for these cells or cell lines in order to investigate if the spatially interacting genes tend to have similar functions.

We compared the function similarity of spatially interacting gene pairs and non-interacting gene pairs in terms of three function categories of Gene Ontology [29] : Molecular Function (MF), Biological Process (BP) and Cellular Component (CC). Our analyses demonstrate that strongly interacting genes tend to have very similar function, and spatial gene-gene interaction is generally not or only weakly correlated with the sequential genomic distances between genes and with sequence identity between genes. However, strongly interacting genes with very similar function often have relative shorter average genomic distance and higher average sequence identity. Combining gene-gene interaction with either genomic distance or sequence identity can inform gene-gene function similarity better than either one of them. Furthermore, we developed a gene function prediction method based on spatial gene-gene interaction networks constructed from the Hi-C data. The method can rather accurately predict the function of a large number of genes based on their interaction with other genes, indicating the gene function prediction power of spatial gene-gene interaction information.

2.3 Methods

2.3.1 Calculation of gene function similarity between two genes

We used the Gene Ontology (GO) terms [29] to describe the function of a gene in three categories: Molecular Function (MF), Biological Process (BP) and Cellular Component (CC). We applied the online tool G-SESAME [30] and the python package FastSemSim [31] to calculate the functional similarity score between any two GO terms. The annotated functions of the human genes were retrieved from the Uniprot database [32]. We used the maximum function similarity score between the GO terms of two genes as the measure of the function similarity between them when we assessed the function similarity of interacted and non-interacting gene pairs.

2.3.2 Construction of genome-wide spatial gene-gene interaction networks

We downloaded the gene information (the start and end positions of the genes) of the human genome (build 36.3) from the NCBI website. We only considered the GENE entries without using other entries, such as PSEUDO, RNA, CDS and UTR. Based on the gene definitions, we constructed spatial gene-gene interaction networks from the Hi-C data of the Primary human B-acute lymphoblastic leukemia (ALL), the MHH-CALL-4 B-ALL cell line (CALL4), and the follicular lymphoma cell-line (RL) sequenced using an Illumina HiSeq 2000 [22], as well as that of the normal human B-cell line (GM06990) [6].

2.3.3 Calculation of sequence identity

The dynamic programming technique is used to calculate the sequence identity of two protein sequences of a gene pair. Given two protein sequences: $X = (x_1, x_2, \dots, x_m)$ and $Y = (y_1, y_2, \dots, y_n)$, we define the i th prefix of X as $X_i = (x_1, x_2, \dots, x_i)$, i is in the range between 1 and m . The longest continuous / non-continuous common subsequence (LCS) of these two sequences ($LCS(X, Y)$) is the longest subsequence which exists in both sequences. We define $c[i, j]$ to be the length of $LCS(X_i, Y_j)$. The following recursive formula is used for calculating the length of $LCS(X_i, Y_j)$: [33]

$$c[i, j] = \begin{cases} 0, & \text{if } i = 0 \text{ or } j = 0 \\ c[i - 1, j - 1] + 1, & \text{if } (i, j > 0 \text{ and } x_i = y_i) \\ \max(c[i, j - 1], c[i - 1, j]), & \text{if } (i, j > 0 \text{ and } x_i \neq y_i) \end{cases}$$

A $m \times n$ matrix is used to store $c[i, j]$. $c[m, n]$ contains the length of $LCS(X, Y)$. We calculate the sequence identity of two protein sequences as $LCS(X, Y)$ divided by the maximum sequence length of X and Y .

To make comparison, we also apply Needleman-Wunsch algorithm to align two sequences using BLOSUM62 as a substitution matrix, and calculate the sequence identity as the percentage of aligned part between these two sequences.

2.3.4 Gene function prediction based on spatial gene-gene interaction networks

The gene function prediction method has 5 steps: **(1)** calculating the probability of a GO term (GO_1) for a gene given a known GO term (GO_2) of its neighboring gene, i.e., $P(\text{a gene has } GO_1 \mid \text{the gene's neighbor has } GO_2)$, based on the entire interaction networks of the ALL B-cell; **(2)** For each gene on the interaction network of the Call4 cell line, randomly selecting one of its neighboring gene having function annotations;

(3) Obtaining the GO terms of the selected neighboring gene; (4) For each GO term (G_i) of the neighboring gene, calculating the probability of other GO terms (G_j) for the target gene according to the conditional probability $P(G_j | G_i)$ pre-computed in Step (1); and (5) summing up the probabilities of each GO term inferred for the target gene into frequencies and ranking the GO terms based on their frequencies as the predictions for the target gene.

Once one or more GO terms are predicted for a gene, we use FastSemSim to compute the similarity between each predicted GO term and each of the real GO term of the gene. The maximum similarity between a predicted GO term and a real GO term is considered as the accuracy (i.e. similarity score) of the prediction.

2.4 Results and Discussion

2.4.1 The spatial gene-gene interaction network for whole genome and thresholds for substantially interacting gene pairs

We construct the gene-gene interaction network of the whole genome for the Hi-C data of three malignant B-cell / cell lines [22] and one normal B-cell [6]. A node and edge in the gene-gene interaction network represents the gene and spatial interaction between genes. In order to control the influence of the noisy chromosomal contacts in the Hi-C data, we consider that there existed a substantially interaction between two genes only if the number of chromosomal contacts observed between the two genes in the Hi-C data is greater than a pre-defined threshold. The interaction between two genes is considered strong when the number of contacts between them is greater than the pre-defined threshold. Higher the contact number, stronger is the interaction.

Since the number of chromosomal contacts automatically increases with respect to the total number of Hi-C reads in a Hi-C data, we set different thresholds on the

four Hi-C datasets in order to make the number of the substantially interacting genes in these datasets largely the same. Actually, instead of using the number of nodes, similar threshold can be found on the four Hi-C datasets based on the number of edges in the interaction network. **Figure 2.1A** shows how the number of interacting genes in the spatial gene-gene interaction networks of the four Hi-C datasets changes with respect to the contact thresholds. The plot shows that the number of interacting genes / nodes decreases fast at the beginning and eventually levels off as the threshold increases. The decrease is most drastic on the spatial gene-gene interaction networks of the Normal B-Cell since the total Hi-C reads in its dataset is much smaller than the other three data sets. Assuming the number of interacting genes in the four interaction networks is similar, we set different thresholds on the datasets in order to select the same number of interacting genes in the **Figure 2.1A**. **Table 2.1** reports the thresholds used on each dataset in order to obtain $\sim 7,000$ or $\sim 12,000$ interacting genes, respectively. These two sets of thresholds are selected because they are the only two thresholds that can lead to the similar number of interacted genes in the four cells / cell lines. About 7,000 interacted genes can be found in all four cells / cell lines if the first threshold (the higher threshold) is used, and about 12,000 interacted genes are obtained if the second threshold (the lower one) is applied. According to **Figure 2.1A**, the number of interacting genes changes relatively faster at around the second threshold than at around the first threshold. So, the first threshold leads to a more stable gene-gene interacting network, which is used for all the analysis in this work.

Figure 2.1B illustrates the largest interacting gene cluster in the spatial gene-gene interaction network for the Call4 at the interaction threshold 16. At this threshold, 7,019 genes were found to interact, which is close to the level-off point of the curves of the three malignant cells / cell-lines in **Figure 2.1A**. All the genes that are connected by at least one path in the gene-gene interaction network are defined as a

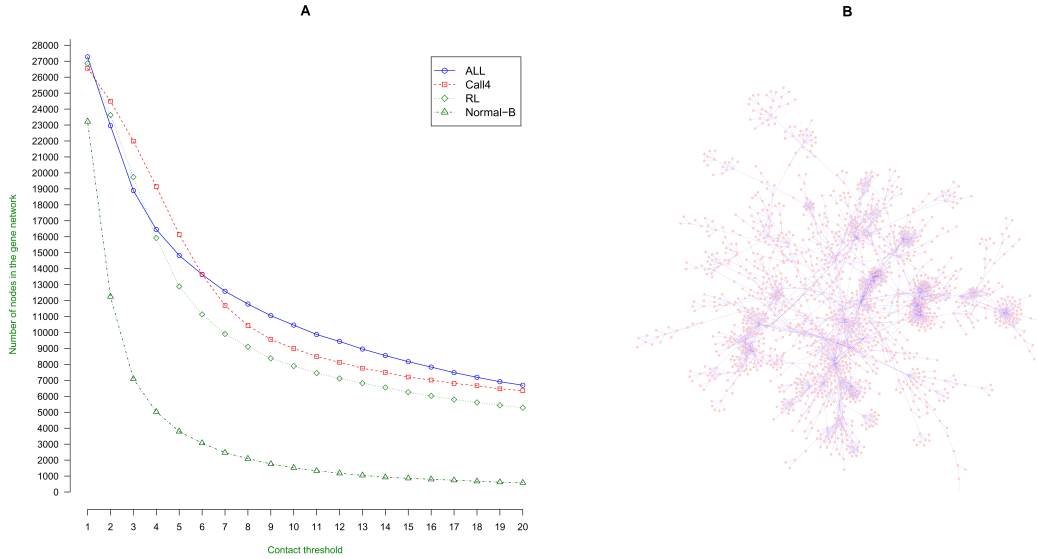


Figure 2.1: Visualization of gene-gene interaction network. Figure 2.1A is the plot of the numbers of interacted genes against interaction / contact thresholds for four cells / cell lines respectively. X-axis denotes the interaction thresholds and Y-axis the numbers of interacted genes found at the thresholds. Figure 2.1B is the visualization of the largest cluster of the gene-gene interaction network for the Call4 cell line at interaction threshold 16. The network was visualized by Cytoscape [40].

Table 2.1: Contact thresholds and the corresponding numbers of interacted genes for the spatial gene-gene interaction networks constructed for four cells / cell lines.

	ALL	Call4	RL	Normal-B
Contact threshold	7	7	5	2
Number of gene nodes	12581	11693	12882	12251
Contact threshold	18	16	12	3
Number of gene nodes	7191	7019	7119	7089

cluster. The cluster with largest number of genes is the largest cluster shown in the figure.

2.4.2 The function similarity of gene pairs that do not spatially interact and that have substantial interactions

We compare the function similarity of gene pairs that substantially interacted (i.e., Hi-C contact number \geq a predefined threshold) and that did not interact in terms of Gene Ontology (GO) function definitions. **Figure 2.2** shows the histogram of the function similarity of non-interacting gene pairs and interacting gene pairs in the three GO categories (BP, CC, MF), respectively. The interacting gene pairs were selected from the genes that had ≥ 18 Hi-C contacts and the non-interacted pairs were the ones randomly selected that had no Hi-C contacts according to the Hi-C data of the ALL cell. The most obvious difference in the function distribution is that substantially more interacting genes had almost identical function (i.e. similarity bin 10 in the figure) than the non-interacting genes, while fewer interacting gene pairs fell into other function similarity bins than non-interacting gene pairs. This is the case for all three GO function categories, even though the level of the difference in the function similarity bin 10 is somewhat different. In order to identify the interacting genes with highly similar functions, we calculate the statistics of the number of spatial interactions for the gene falling into different function similarity bins.

2.4.3 The statistics of the number of interactions for substantially interacting gene pairs at each function similarity level

Figure 2.3 shows the average number of observed chromosomal interactions for the gene pairs in each function similarity bin in each GO function category. It is very interesting to see that the average number of interactions between genes in function

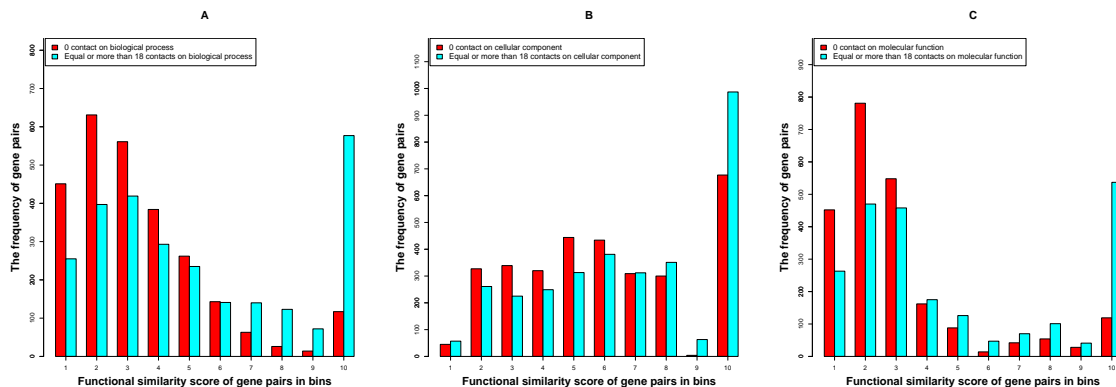


Figure 2.2: The histograms of gene function similarities of non-interacted gene pairs and substantially interacted gene pairs. Figure 2.2A, 2.2B, and 2.2C represent the histogram for Biological Process, Cellular Component, and Molecular Function respectively.

similarity Bins 1-9 is rather similar, while the average number of interactions for the genes in Bin 10 is much higher. The average numbers of interactions between genes in function similarity bins 9 and 10 for three function categories (BP, CC, MF) are (62.22, 775.12), (46.54, 414.28), and (41.61, 835.80), respectively. According to the Welch two-sample t-test, the p-value of the difference in the average numbers of interactions between bin 9 and bin 10 is less than $2.2e-16$ for all three categories. This indicates that the interacting genes with almost identical functions are more strongly interacted than the rest of interacting gene pairs. In other words, the strongly interacting genes tend to have almost identical function.

Since a few outliers (extremely large numbers) may skew the average number substantially, we also calculated the quantiles of the interaction numbers in the function similarity bins. Indeed, the genes in function similarity Bin 10 have substantially more interactions than genes in the other bins. For example, the median interaction number and the quantile at 75% in Bin 10 for Biological Process is 407 and 1187, which are much higher than 31.5 and 47.75 in Bin 9. Interestingly, the genes in the other bins except Bin 10 seem to have similar median interaction numbers despite their different levels of function similarity.

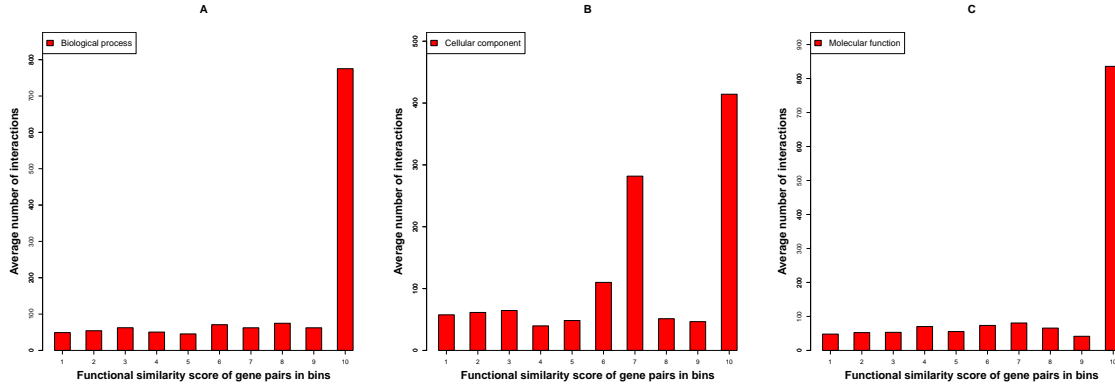


Figure 2.3: The average number of interactions between substantially interacted gene pairs within each functional similarity bin in three function categories. This is for the primary tumor B-cells (ALL). Figure 2.3A, 2.3B, and 2.3C represent the histogram for Biological Process, Cellular Component, and Molecular Function respectively.

2.4.4 The sequential genomic distance for substantially interacting gene pairs at each function similarity level

We gauge the relationship between the sequential genomic distances of interacting gene pairs in function similarities. Figure 2.4 (A, B, C) illustrates the average function similarity in each genomic location distance bin for Biological Process, Cellular Component and Molecular Function, respectively. Gene pairs are classified into ten bins based on their genomic location distance, and each bin has the same number of gene pairs. The gene pairs are substantially interacting genes (≥ 18 Hi-C interactions) identified in the Hi-C data of the ALL cell. The genomic distance between two genes is the number of base pairs between their start locations. Since it is difficult to define the sequential genomic distance between genes on two different chromosomes, inter-chromosomal gene pairs were not considered in the calculation. The results show that gene pairs with short genomic distances usually have high function similarity. For example, gene pairs in the first three bins have high function similarity comparing with gene pairs in other bins for all three categories. Especially for Biological Process and Molecular Function, the function similarity of Bin 1 (relatively in short genomic distance) is around two times higher than the function similarity of Bin 10.

In order to reduce the influence of some genes with extremely large genomic distance, we generated the box plots for genomic distances in each function similarity bin for each function category. The result shows that the median genomic distance of all gene pairs with functional similarity score (< 0.9 in Bins 1-9) is longer than the ones with very high functional similarity score (>0.9 in Bin 10). For example, for biological process category, the median genomic distance in Bin 1 is 574,281 bp, longer than 72,312 bp in Bin 10; for the cellular component, the median genomic distance in Bin 1 is 458,991 bp, longer than 201,949 bp in Bin 10; and for the molecular function, the median genomic distance in Bin 1 is 565,609 bp, longer than 64,167.5 bp in Bin 10. In summary, the genomic distance can somewhat distinguish the interacting gene pairs with very high function similarity from the rest of interacted pairs. However, its effect is more pronounced on Biological Processes and Molecular Function than on Cellular Component.

Similarly, we calculated the genomic distances for 20,000 randomly selected gene pairs in 10 function similarity bins that did not spatially interact. In contrast to the interacting gene pairs, the median genomic distances are relatively close for non-interacting gene pairs in different bins, and gene pairs in high function similarity bins do not always have minimum median genomic distances. Furthermore, the genomic distance of gene pairs with no interaction is relatively longer than substantially interacting gene pairs in different functional similarity bins.

2.4.5 Sequence identity of substantially interacting genes at each function similarity level

We assessed the relationship between sequence identity and function similarity for substantially interacting gene pairs (≥ 18 Hi-C contacts) in the Hi-C data of the ALL cell line. Figure 2.5 (A, B, C) illustrates the box plots of the sequence identity of gene pairs in 10 function similarity bins for Biological Process, Cellular Component, and

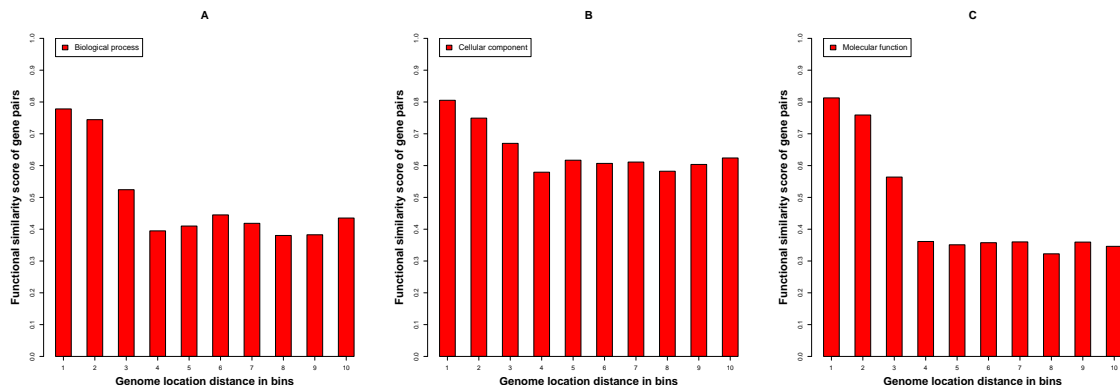


Figure 2.4: The average genomic distances of substantially interacted gene pairs in each functional similarity bin in three function categories. This is for the primary tumor B-cells (ALL). Figure 2.4A, 2.4B, and 2.4C represent the histogram for Biological Process, Cellular Component, and Molecular Function respectively.

Molecular Function, respectively. The median sequence identity of gene pairs in Bin 10 (i.e. similarity score in $[0.9, 1]$) is generally higher than the rest bins, even though the difference is more pronounced for Biological Process and Molecular Function than Cellular Component. For Biological Process and Molecular Function, the median sequence identity in Bin 10 is about 0.6, and for Cellular Component, the median sequence identity of gene pairs in Bin 10 is about 0.4. The median sequence identity in other 9 bins for each function category is similar to each other and substantially lower than Bin 10, even though there are quite some outliers in Bin 10 that have very low sequence identity. Moreover, the sequence identity calculated by Needle-Wunsch algorithm is also included in the figure to make comparison with the one by dynamic programming technique. This figure shows that the average sequence identity in Bin 10 is much higher than most other bins for each category. Interestingly, the average sequence identity increases as the function similarity bin increases, and the average sequence identity in Bin 10 for each category is always relatively high. Therefore, the sequence identity could be a factor to predict if two interacting genes have very high functional similarity score (≥ 0.9). The substantially high sequence similarity between interacting genes with high function similarity may be partially due to the

duplicated genes that still maintain highly similar functions and are spatially close [34, 35].

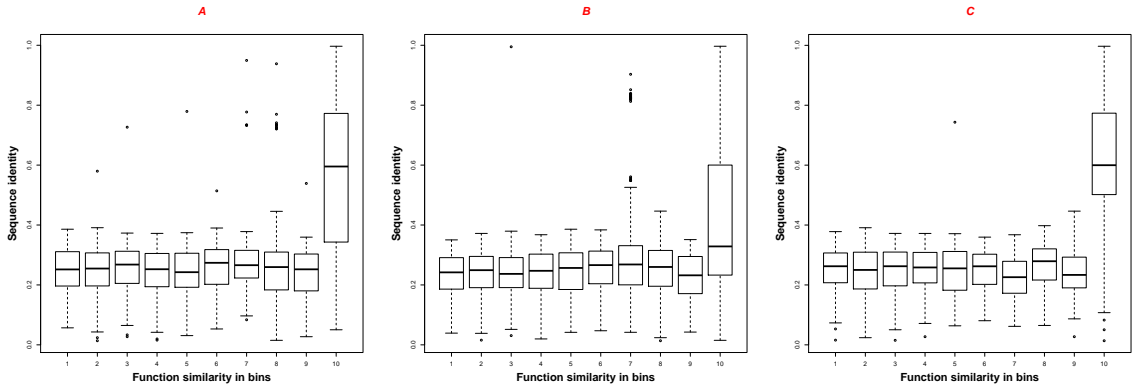


Figure 2.5: The boxplot of gene sequence identity against function similarity in three GO categories. Figure 2.5A, 2.5B, and 2.5C represent the histogram for Biological Process, Cellular Component, and Molecular Function respectively. This figure is generated on the gene-gene interaction network of the ALL B-cell constructed at interaction threshold 18. X-axis denotes the functional similarity scores / bins and Y-axis gene sequence identity.

2.4.6 Identification of interacting genes with high function similarity with sequence identity, genomic distance, and interaction strength

Since the special group of interacting genes with function similarity score ≥ 0.9 tend to have higher sequence identity, shorter genomic distance, and stronger spatial interactions, we tested how these three factors could identify this group of genes.

The results show that applying the thresholds on the three factors can identify 372 – 398 common interacting gene pairs with high function similarity for each function category, while using each threshold can identify some gene pairs not recognized by another factor. Applying sequence identity or genomic distance to interacting genes can identify more gene pairs with high function similarity than using interaction number, suggesting combining sequence identity or genomic distance with gene spatial interaction information could be more sensitive in identifying genes with high

function similarity than using interaction information alone. In general, the substantial number of common gene pairs identified by each of the three factors demonstrates the convergence in the group of interacting genes with high function similarity and the distinct gene pairs found by each factor also suggests the complementarity of the three factors.

2.4.7 The relationship between sequence identity and function similarity for substantially interacting gene pairs and random non-interacting gene pairs

Figure 2.6 plots function similarity against sequence identity of 7,987 interacting genes pairs with ≥ 18 Hi-C contacts (excluding ones without GO annotations) and 20,000 randomly selected, non-interacting gene pairs in the gene-gene interaction network of the ALL cell line. For non-interacting gene pairs, the correlation between sequence identity and function similarity is very low, i.e., 0.02, 0.05, and 0.03 in three function categories (i.e. BP, CC, and MF). In contrast, for the substantially interacting gene pairs, the correlation score is much higher, i.e., 0.67, 0.41, and 0.70 for three function categories, respectively. In order to compare the function similarities of interacting genes and non-interacting genes more rigorously, we also select non-interacting gene pairs by restricting their genomic distances are similar to the selected highly interacting gene pairs (within 35bp). The function similarity against sequence identity for highly interacting gene pairs and random gene pairs with similar genomic distance for four cell/cell lines has been calculated. The correlation between sequence identity and function similarity for non-interacting random gene pairs with the genomic distance restriction is higher than that of non-interacting random gene pairs without the genomic distance restriction, but is still lower than that of substantially interacting gene pairs. For example, the correlation between sequence identity and function similarity for these three gene groups in the ALL cell is 0.37, 0.25, and 0.43

respectively. This suggest both genomic distance and spatial gene-gene interaction between gene pairs affect the correlation between their sequence identity and function similarity, and spatial gene-gene interaction further strengthens the correlation when the genomic distance between genes is similar.

Figure 2.7 plots the numbers of interactions of gene pairs against their sequence identities. The top 20 points with extremely large number of interactions are removed. According to the plot, the number of interactions varies a lot when sequence identity is either around 0 or 1. Indeed, the Pearson’s correlation between sequence identity and the number of interactions for all spatially interacting gene pairs is only 0.223.

The weak correlation between interaction numbers and sequence identity and the relatively strong function prediction power of considering both sequence identity and interaction numbers suggest that they are two rather independent factors informing the function similarity of two genes. In another words, genes with similar sequence more likely interact for the purpose of carrying out similar functions.

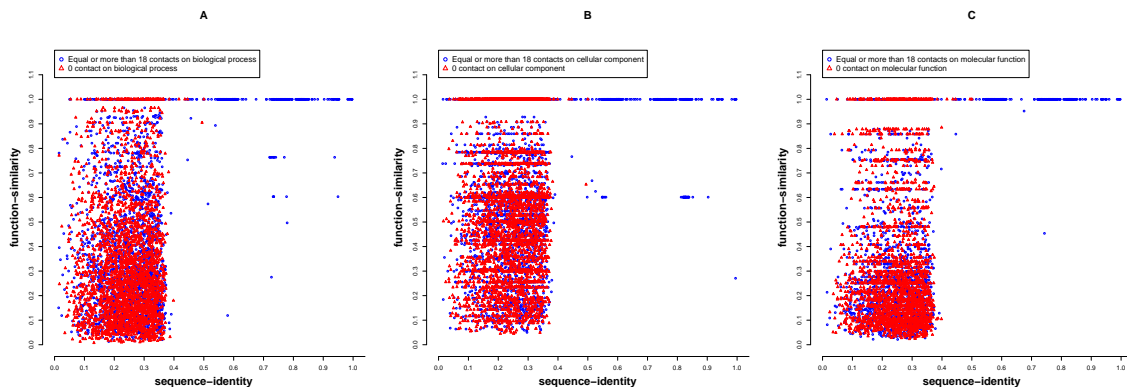


Figure 2.6: Plot of function similarity against sequence identify for substantially interacted gene pairs and non-interacted gene pairs. X-axis denotes the gene sequence identity and Y-axis the gene function similarity in all three categories (BP, CC, MF), respectively.

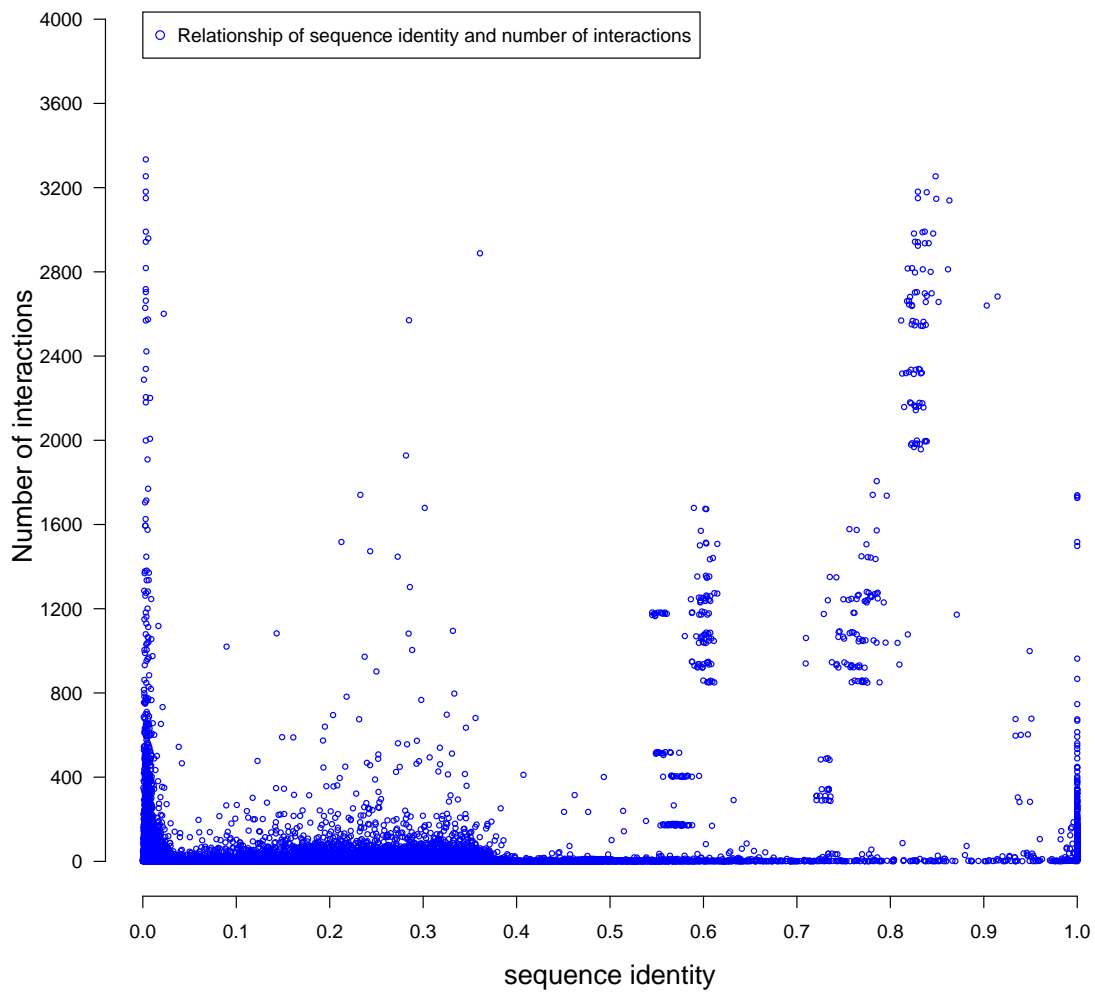


Figure 2.7: The sequence identity and the number of gene interactions. The number of interactions is normalized to the range of 0 to 1. The result is generated on the ALL gene-gene network with ≥ 1 interactions. X-axis denotes the sequence identity and Y-axis the normalized number of interactions.

2.4.8 The relationship among genomic distance, interaction numbers, and function similarity for interacting gene pairs

Figure 2.8 is the 3D plot of genomic distance, number of interactions and function similarity for interacting gene pairs. Since it is impossible to calculate the genomic distance between inter-chromosomal gene pairs, the analysis in **Figure 2.8** only considers intra-chromosomal gene-gene interactions in order to calculate the genomic distance between the genes. According to **Figures 2.8(A)** and **2.8(C)**, although the number of interactions between genes generally increases as their genomic distance decreases, most of gene pairs with short genomic distance, but small number of interactions tend to have low function similarity in terms of biological process and molecular function. According to **Figure 2.8(B)**, for quite a few gene pairs with high function similarity (>0.9) in terms of cellular component, their genomic distance varies a lot when the number of interactions are small, however, when the number of interactions is large, their genomic distance is short. In order to consider the genomic distance for intra-chromosomal gene-gene interactions, we generated two new analyses by separating the gene pairs into two groups: short-range interaction pairs and long-range interaction pairs by using the median genomic distance between interacted gene pairs as threshold. Generally, the pattern regarding the relationships among function similarity, genomic distance and number of gene-gene interactions is similar to that in **Figure 2.8**. However, one interesting finding is that the relationship between genomic distance and function similarity somewhat differ for these two groups. For the gene pairs with genomic distance longer than the median, the function similarity clearly decreases as the increasing of genomic distance, whereas no very clear such pattern has been found in gene pairs with short genomic distance. For the gene pairs with shorter genomic distance, the number of gene-gene interactions has more impact on function similarity than genomic distance. Taken together,

the results suggest the complementarity of the two factors in informing gene function similarity.

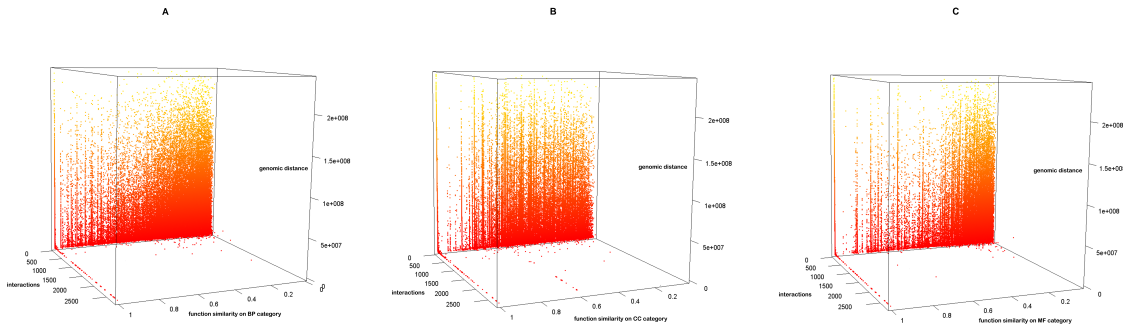


Figure 2.8: The 3D plot of genomic distance, number of interactions and the function similarity in three function categories. Figure 2.8A, 2.8B, and 2.8C represent the histogram for Biological Process, Cellular Component, and Molecular Function respectively. The yellow dots represent long genomic distances and the red ones the opposite.

2.4.9 Evaluation of gene function predictions based on spatial gene-gene interactions

We developed a gene function prediction method based on spatial gene-gene interaction networks, which predicts the function of a gene using the known functions of its spatially interacted neighbours (see Methods section for details). We calculated the probabilistic relationship between GO terms of a gene and the GO terms of its neighbouring genes on the spatial interaction networks constructed from the Hi-C data of the ALL B-cell. The knowledge was applied to make gene function prediction on the Call4 cell-line. We generated networks with different interaction thresholds ($\geq 1, 2, 3, 4, 6, 8, 10, 12, 14, 16$) for the Call4 cell line. For the case of 0 threshold, which means there is no interaction between genes, our current function prediction method based on spatial gene-gene interaction cannot make any prediction. This means that our current function prediction method is limited on predicting the functions of the genes on the gene-gene interaction network, which could be expanded in the future

to make function prediction using other information, such as gene sequence identity.

Figure 2.9 illustrates the histogram of the similarities between predicted functions and true functions of the tested genes. For all the thresholds, the similarity score of the predictions for the majority of tested genes were very high (>0.9). When the interaction threshold is set to the lowest number, i.e.1, at least one highly accurate function was predicted for $\sim 9,000$ genes, while much fewer genes had predictions with relative lower accuracy. This indicates that the prediction method is rather robust against the potential noise in the interaction data. As the interaction thresholds increased, the function predictions could be made for fewer genes as there were fewer interacting genes in the spatial gene interaction network. However, the percentage of genes having high accurate predictions (similarity score >0.9) is generally higher. For example, with interaction threshold 1, the number of genes having high accurate predictions (similarity score > 0.9) is 9142, and the number of genes having low accurate predictions (similarity score < 0.1) is 214; with interaction threshold 16, the number of genes having high accurate predictions (similarity score > 0.9) is 1357, and the number of genes having low accurate predictions (similarity score < 0.1) is 33.

The number of GO function terms predicted for each gene also affects the sensitivity and specification of gene function prediction. **Figure 2.10** shows the histograms of the maximum function similarity between predicted GO terms and true GO terms. Not surprisingly, as the number of GO term prediction increased, more and more genes got at least one highly similar GO function prediction.

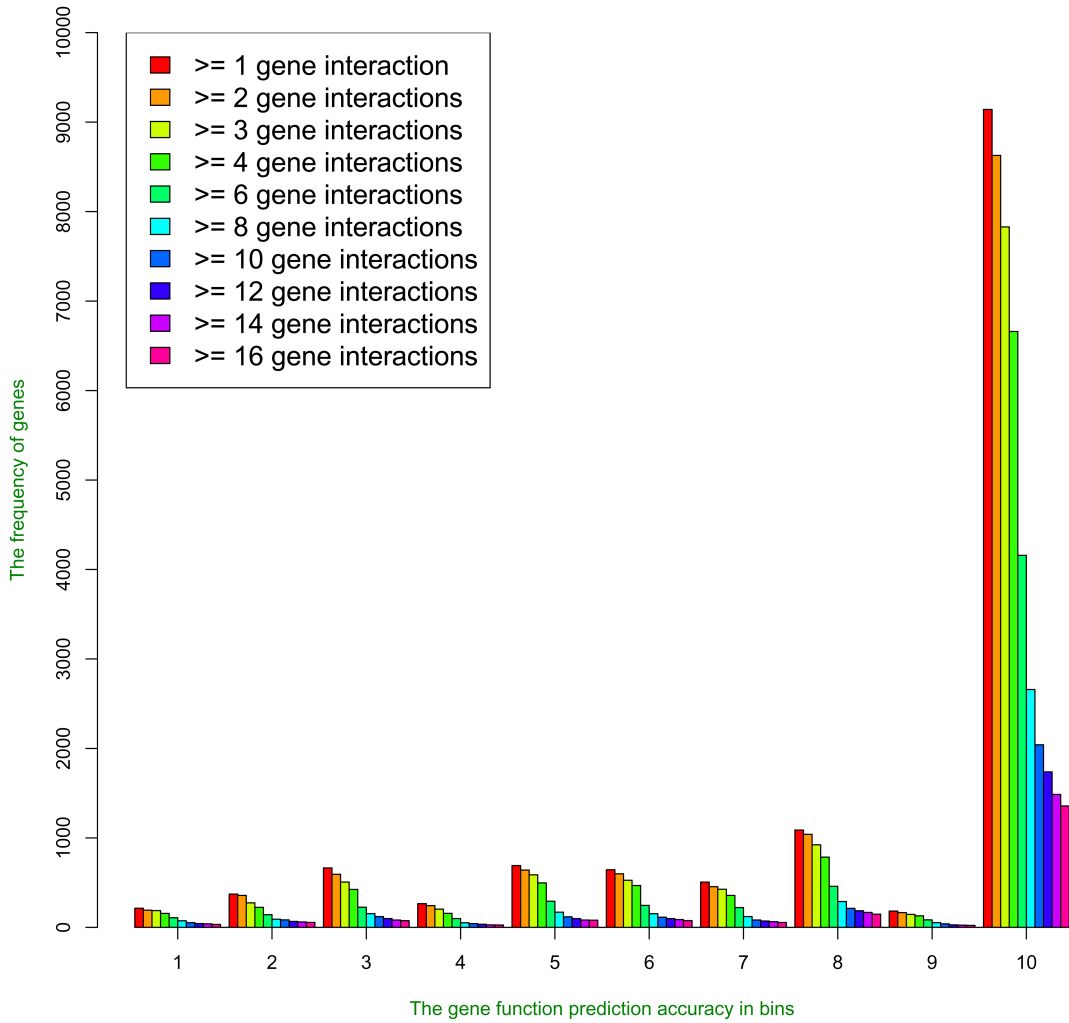


Figure 2.9: The histograms of function prediction accuracy (the maximum similarity scores between predicted GO terms and real GO terms) on the spatial gene-gene interaction networks of the Call4 cell line at different interaction thresholds

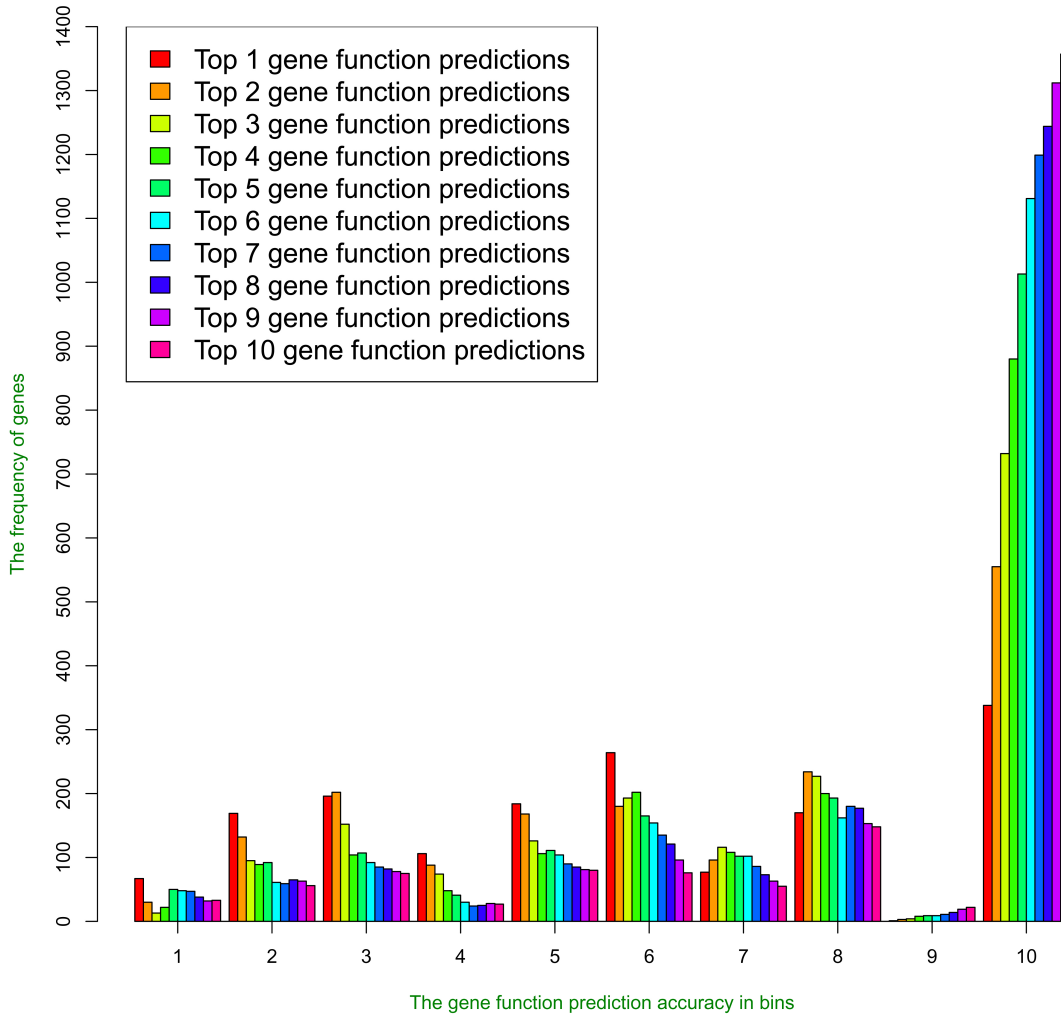


Figure 2.10: The histograms of function prediction accuracies for different numbers (1 – 10) of GO terms selected as predictions.

Chapter 3

Integrated protein function prediction by mining function associations, sequences, and protein-protein and gene-gene interaction networks

3.1 Abstract

Protein function prediction is an important and challenging problem in bioinformatics and computational biology. Functionally relevant biological information such as protein sequences, gene expression, and protein-protein interactions has been used mostly separately for protein function prediction. One of the major challenges is how to effectively integrate multiple sources of both traditional and new information such as spatial gene-gene interaction networks generated from chromosomal conformation data together to improve protein function prediction. In this work, we developed three different probabilistic scores (MIS, SEQ, and NET score) to combine protein sequence, function associations, and protein-protein interaction and spatial gene-gene

interaction networks for protein function prediction. The MIS score is mainly generated from homologous proteins found by PSI-BLAST search, and also association rules between Gene Ontology terms, which are learned by mining the Swiss-Prot database. The SEQ score is generated from protein sequences. The NET score is generated from protein-protein interaction and spatial gene-gene interaction networks. These three scores were combined in a new Statistical Multiple Integrative Scoring System (SMISS) to predict protein function. We tested SMISS on the data set of 2011 Critical Assessment of Function Annotation (CAFA). The method performed substantially better than three base-line methods and an advanced method based on protein profile-sequence comparison, profile-profile comparison, and domain co-occurrence networks according to the maximum F-measure. The web server of the method is available at: <http://tulip.rnet.missouri.edu/profunc/>.

3.2 Introduction

Protein function prediction is important for understanding life at the molecular level and therefore is highly demanded by biomedical research and pharmaceutical applications [36]. There are a large amount of sequence data generated by next generation sequencing every day, however, the annotation of the function of these sequences by experimental is still a big challenge because of the inherent difficulty and considerable expense [7]. In addition, some experiments in *vitro* may not faithfully reflect a protein's activity in *vivo* [29]. Therefore, accurately predicting protein function from sequence using computational methods is a useful way to solve the problem at large scale and low cost.

A number of computational protein function prediction methods had been developed in the last few decades [3, 7, 37, 38, 39, 40, 41, 42]. The most commonly used method is to use the tool Basic Local Alignment Search Tool (BLAST) [43] to

search a query sequence against protein databases containing experimentally determined function annotations to retrieve the hits based on the sequence homology. The function of homologous hits is used as the prediction of the query sequence. Some of this kind of methods are GOtch [44], OntoBlast [45], and Goblet [46]. However, the prediction coverage of BLAST based methods may be low because BLAST is not sensitive enough to find many remote homologous hits. Some other methods such as PFP [47] use profile-sequence alignment tool PSI-BLAST [43] to get more sensitive predictions.

In addition to sequence homology, some methods use other information to predict protein function. In order to incorporate the prediction of functional residues into the prediction of protein function at the whole molecular level [13, 48], some methods predict protein function based on amino acid sequences [49, 50]. Some other methods make function prediction based on protein-protein interaction networks [51, 19, 37, 40, 9, 41] assuming that interacted proteins may share the similar function. Others make function prediction by using protein structure data [13, 42, 52], microarray gene expression data [53], or combination of several sources of information [54, 55, 56, 57]. One of the biggest challenges of protein function prediction is how to obtain diverse relevant biological data, such as protein amino acid sequence, gene-gene interaction data, protein-protein interaction data, protein structure from multiple reliable sources efficiently, and how to integrate these biological data to make protein function prediction [58].

Besides the development of function prediction methods, unbiased benchmarking of different method is also very important for the community to identify the strengths and weaknesses of different methods in order to develop more accurate function prediction methods. The Critical Assessment of Function Annotation (CAFA, <http://biofunctionprediction.org/>) is an experiment designed to provide such a large-scale assessment of protein function prediction methods, and it has benefited the

whole community by involving a significant number of groups to blindly test their function prediction methods on the same set of proteins within a specific time frame [36], which also provide a test ground for benchmarking new methods including our method developed in this work. During CAFA in 2011, 30 teams associated with 23 research groups participated in the effort, and several new methods have been developed to achieve high accuracy of protein function prediction [36]. For example, sequence-based function prediction methods PFP [47, 59] and ESG [60] from professor Kihara’s lab use PSI-BLAST one time and recursively against the target sequence to get the hits for protein function prediction [61, 62], method from the team Jones-UCL integrates a wide variety of biological information sources into a framework for protein function prediction [63], Argot2 combines the clustering process of GO terms dependent on their semantic similarities and a weighting scheme which assesses retrieved hits sharing a certain degree of biological features with the sequence to annotate for protein function prediction [64, 65], method GOstruct uses co-mention and bag-of-words features mined from the biomedical literature for protein structure prediction [66], PANNZER uses weighted k-nearest neighbour methods with statistical testing to maximize the reliability of a functional annotation [67], and MS-kNN method finds k-nearest neighbors of a query protein based on different types of similarity measures and predicts its function by weighted averaging of its neighbors’ functions [68].

In this work, we develop a novel Statistical Multiple Integrative Scoring System (SMISS) for protein function prediction. SMISS integrates the information from homologs found by PSI-BLAST, protein-protein interaction networks, spatial gene-gene interaction networks derived from chromosomal conformation capturing data, and amino acid sequence information, and calculates three different probability scores (MIS score, NET score, and SEQ score) for each GO term based on these information, and makes function prediction based on the combination of these three scores. SMISS is a very open system, which can be easily expanded to include more biological

information to enhance the accuracy of protein function prediction.

The rest of the paper is organized as follows. In the Method section, we describe how to calculate three different scores and integrate them to make protein function prediction. In the Results and Discussion section, we blindly test our method and compare it with three base-line methods and three network-based protein function prediction methods. In the Conclusion section, we summarize the work and discuss the direction of future work.

3.3 Methods

The SMISS (Statistical multiple integrative scoring system for protein function prediction) method uses three different scores: the MIS score (Multiple Integrated Score) which is calculated based on the PSI-BLAST hits and their GO terms inferred from the Swiss-Prot database by data mining techniques, the NET score (Network score) which is calculated from spatial gene-gene interaction networks and protein-protein interaction networks, and the SEQ score (Sequence score) which is calculated from the amino acid sequence of a query protein. We test three different predictors by combining these three scores in different ways. The first one is SMISS-predictor, which combines all three scores. The second one is MIS-predictor, which only use the MIS score. The third one is MIS-NET-predictor, which combines the MIS score and the NET score. **Figure 3.1** shows the overall flowchart of our three predictors. We introduce the method to calculate each score in the following section.

MIS score

The calculation of MIS score is different for two types of GO functions. For the first type, the MIS score is calculated from the PSI-BLAST results while searching against Swiss-Prot [41] database. The default setting of PSI-BLAST has been used

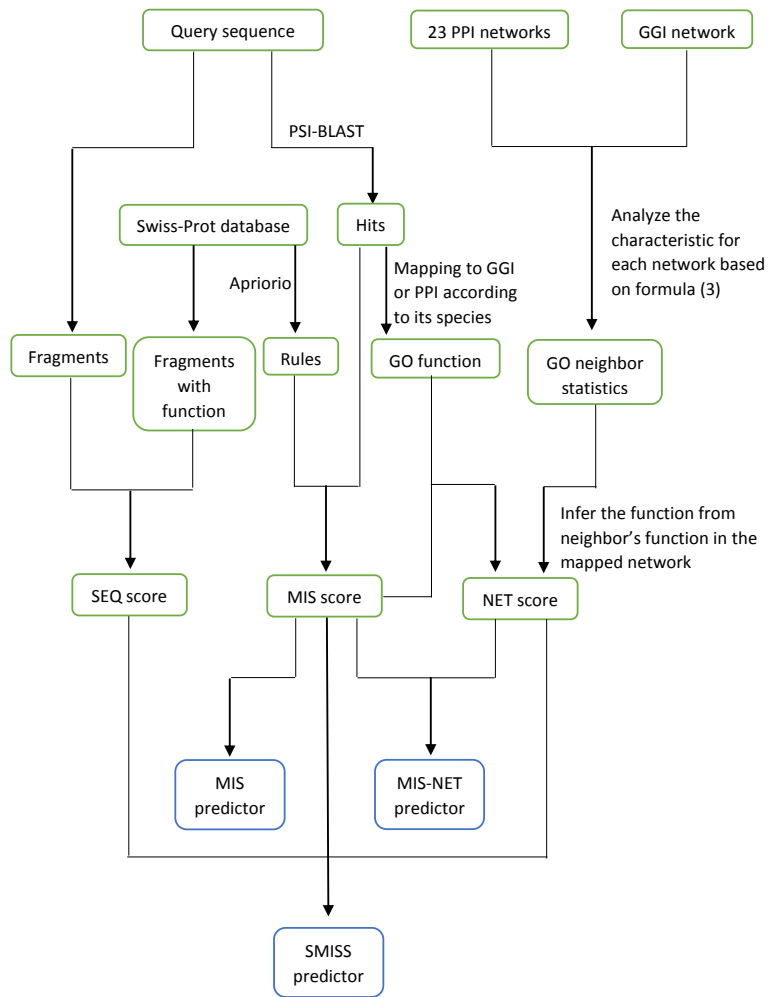


Figure 3.1: The overall flowchart of our method.

with 3 iterations on Swiss-Prot databased released on Jul. 2010 for benchmark on CAFA1, the default e-value threshold (i.e. 10) is applied for prediction, and the predictions with e-value larger than 0.01 are ignored since their confidence score is 0 based on formula (3.1). All the potential distantly homologous protein hits and their e-values are retrieved and stored. The e-value of each protein hit is converted into a probabilistic confidence score using the following formula:

$$S = \frac{-\log_{10}t}{200} - 0.01 \quad (3.1)$$

In this formula, t is the e-value of the protein, and S is the probabilistic confidence score. We constrain the confidence score to be in the range of 0 and 1. That is, the confidence score is set to 0 for all hits with e-value (t) larger than 0.01, and all hits with e-value less than e-202 have confidence score 1. Assuming that N protein hits have confidence score larger than 0, and P_i is the number i protein ($i \in [1, N]$), we can get all gene ontology (GO) terms from the Swiss-Prot database for each P_i . The n_i GO terms for P_i are denoted as $G_{i_1}, G_{i_2}, \dots, G_{i_{n_i}}$. By applying formula (1), we can calculate the confidence score $P(P_i)$ of each GO term associated with P_i . The same confidence score is assigned to each GO term of P_i , such that $P(G_{i_j}) = P(P_i)$, where $j \in [1, n_i]$. Given the GO terms lists (G_{i_j}) with the probabilistic confidence scores ($P(G_{i_j})$), we combine them to generate a list of unique GO terms (G'_k) and calculate the confidence scores ($P(G'_k)$), while $i \in [1, N]$, $j \in [1, n_i]$, and $k \geq 0$ as follows. Assuming the same GO term G_x appears in the GO term lists of two different proteins i and j with confidence scores $P_i(G_x)$ and $P_j(G_x)$ respectively, the following formula is used to update the combined confidence score of the GO term G_x :

$$p(G_x) = 1 - (1 - P_i(G_x)) * (1 - P_j(G_x)) \quad (3.2)$$

We continuously update the confidence score of any two same GO terms existing

in different proteins by formula (3.2), and it can be proved (details omitted) that the final confidence score for each GO term G_x is: $P(G_x) = 1 - \prod_{i=1}^{i=N} (1 - P_i(G_x))$, where $P_i(G_x)$ is the confidence score of the GO term G_x in the i th protein (P_i). After applying formula (3.2), we can finally get a list of unique GO terms (G'_k) with the calculated confidence score $P(G'_k)$.

For the second type of GO terms, the MIS score is assigned as 1. The GO terms are inferred from the protein hits with confidence score 1. To infer the unobserved GO terms, we first apply Apriori algorithm [69] to mine the association rules from Swiss-prot database. Apriori algorithm is used for association rule mining in transaction database, and here we apply it to get the association rules for protein function prediction. First, we extract the GO function from the Swiss-prot database for each protein sequence. Assuming there are N different GO terms, $G_1 \cdots G_N$, N is the total number of GO terms in the database, and each protein's GO functions are considered as a transaction. Secondly, the apriori algorithm is used to generate the association rules, $G_i, \cdots, G_j \Rightarrow G_k$, i, j , and k are all integers equal or less than N . In our case, that is the association rules between different GO terms. There are two parameters for Apriori algorithm for us to tune: the minimum support and minimum confidence. To decide these two parameters, five cross validation techniques are used, while dividing all GO function transactions into five folds, four of them are used for training, and the other one for testing. The precision and recall are used to evaluate the performance. The minimum support is set to 0.05, and minimum confidence is set to 90 based on the five cross validation result, while 51,512 association rules are generated. More details of tuning the parameters are included in the results and discussion section. Finally, after generating the association rules by data mining technique, we check all combination of GO terms with confidence 1, and apply the association rules mined from Swiss-prot database to infer more GO terms. The MIS score of all inferred GO terms are set as 1. In summary, the MIS score is calculated from PSI-BLAST results

by formula (3.2), and is set as 1 for GO terms inferred by Apriori algorithm.

3.3.1 NET score

Protein-protein interaction networks and spatial gene-gene interaction networks have been used for generating the NET score. irefindex network [70] is used for generating 23 protein-protein interaction networks of multiple species. irefindex provides an index of protein interactions available in a number of primary interaction databases, and we parse it for 22 different species to get 22 protein-protein interaction networks, and one additional network for proteins in other species. The gene-gene interaction network [11] is generated from Hi-C contact data of the normal B-cell [22]. We consider two genes are interacting when the total number of Hi-C contact between them is more than a contact threshold [22]. We want to mention that this gene-gene interaction network is used for proteins in Homo sapiens that can be mapped to it. Otherwise, the 23 protein-protein interaction networks are used. Here, if two genes/proteins are connected in a network, their GO terms are assumed to interact. For any two interacted GO terms G_i and G_j from gene-gene/protein-protein interaction networks, we calculate the probability score between them for statistical analysis as follows:

$$p(G_i|G_j) = \frac{G_i|G_j}{\sum_{k=1}^{k=N} F(G_i|G_k)} \quad (3.3)$$

In formula (3.3), $F(G_iG_j)$ is the total number of interactions for the GO term G_j interacting with GO term G_i . N is the total number of GO terms interacting with GO term G_i . We calculate the scores by this formula for all neighboring GO terms of each 23 protein-protein interaction networks and gene-gene interaction network, and store them for protein function prediction. Given a query sequence, first, we retrieve the protein hits lists with e-values by PSI-BLAST for it. Second, we search

each protein from the protein hits lists starting from lowest e-value until we find one which has GO functions. To predict the GO functions, we map the protein to our generated gene-gene interaction/protein-protein interaction networks. Given the protein is in Homo sapiens and the gene directing the production of it exists in our generated gene-gene interaction network, we use the gene-gene interaction network to predict the GO functions, otherwise, the protein-protein interaction network for species of this protein is used for the function prediction. We store the MIS score of the selected mapped gene/protein as M_map. Thirdly, we obtain the neighbors of the mapped gene/protein in the networks, and get all GO terms G_k from each neighbor gene/protein, while $k \in [1, N]$, and N is the total number of GO terms from all neighbors. Finally, we generate all possible GO term neighbors GN_l for each GO term from the statistics calculated on the gene-gene interaction network / protein-protein interaction network. The probability score for each GO term neighbor GN_l is calculated as M_map times the score generated by applying formula (3.3) to the whole gene-gene/protein-protein interaction network between GN_l and G_k . We combine all GO term neighbors GN_l by formula (3.2), and generate the final GO term list. The final probability score for each GO term is the NET score. Here, $l \in [1, NN]$, and NN is the total number of GO term neighbors.

3.3.2 SEQ score

We calculate the SEQ score from the protein sequence itself. We retrieve all protein sequences and the protein function GO terms in the Swiss-Prot database. We use a 5-residue sliding window technique to divide each sequence into sequence fragment with a length of 5. The reason to use a length of 5 is because we want to include more GO terms and fragments smaller than or equal to 4 cannot represent the structural information accurately [71]. So given the protein sequence with length N, there are in total (N-5) sequence fragments. Let's assume a protein has a number of GO function

terms G_i , while $i \in [1, M]$, and M is the total number of GO terms. The sequence of that protein can be divided into $(N-5)$ sequence fragments, and for one specific sequence fragment S_j , the conditional probability of GO term G_i inferred from it can be calculated in the following formula:

$$p(G_i|S_j) = \frac{F(S_j)}{(N-5)} \quad (3.4)$$

N is the sequence length, and $F(S_j)$ is the frequency of the sequence fragment S_j extracted from the sequence by the 5-residue sliding window technique. The frequency could be more than one since one sequence fragment may exist more than one time in the protein sequence. Secondly, we calculate the probability of GO term G_i inferred from sequence fragment S_j for each sequence by applying formula (3.4). Thirdly, we combine all GO terms with the following formula when the same GO term G_i inferred from the same sequence fragment S_j in different sequence:

$$p(G_o|S_j) = 1 - (1 - P_1(G_i|S_j)) * (1 - P_2(G_i|S_j)) \quad (3.5)$$

$P_1(G_i|S_j)$ and $P_2(G_i|S_j)$ are the probability from the two different sequences. In the prediction phase, for each query protein sequence, we divide it into sequence fragment with 5-residue sliding window technique, and for each sequence fragment, we search against the sequence fragment database built from the Swiss-Prot database by formula (3.5), and get all possible GO terms G_i with the probability score $P(G_i)$. The formula (3.2) is used to combine all same GO terms from different sequence fragment. Finally, we generate a GO term list for the query protein sequence and the SEQ probability score for each GO term.

3.3.3 Score combination

We develop three different predictors with different combination of these three scores. The first predictor is SMISS-predictor that combines all three different GO term lists calculated from MIS, NET and SEQ scores respectively. The following formula is used to calculate the finally combined score for each GO term G_i :

$$P(G_i) = 1 - (1 - W_{MIS} * P_{MIS}(G_x)) * (1 - W_{NET} * P_{NET}(G_x)) * (1 - W_{SEQ} * P_{SEQ}(G_x)) \quad (3.6)$$

$P_{MIS}(G_x)$ is the MIS score of this GO term, $P_{NET}(G_x)$ is the NET score of this GO term, $P_{SEQ}(G_x)$ is the SEQ score of this GO term, W_{MIS} is the weight for MIS score, W_{NET} is the weight for NET score, and W_{SEQ} is the weight for SEQ score. We set the weight 0.5 for MIS score, 0.22 for NET score, and 0.28 for SEQ score empirically, which is based on their accuracy on our local benchmark for each score. The second predictor is MIS-predictor, which only uses the GO term list calculated by the MIS score. And the third predictor is MIS-NET-predictor, which generate two different GO term lists by calculating the MIS score and NET score, and finally combines these two GO term lists for the final prediction. The formula (3.6) is used to combine them while the $P_{SEQ}(G_x)$ is set to 0 for MIS-NET-predictor.

3.3.4 Score scaling

The combined scores may be hard to analyze and evaluate when several GO term predictions have very similar scores close to 1 or when there are no predictions with relatively high confidence score. In order to avoid the problem, the combined scores are rescaled. For all predicted GO terms of a query sequence, we rank them based on the confidence score. Each prediction gets a ranking R_i . A new score ($S + 0.01 -$

$0.01 * R_i$) is assigned to all predictions. S is the initial score, S can be set as 1 or the max confidence score. In our method, we set it to 1. Two predictions with the same confidence score have the same ranking. For the predictions with the non-positive scaled score, we reset the score to 0.01.

3.4 Results and discussion

3.4.1 Parameters in Apriori algorithm for calculating MIS score

We apply data mining technique apriori algorithm to obtain more GO terms as the predictions. There are two parameters for the Apriori algorithm: minimum support and minimum confidence. Given a rule $X \Rightarrow Y$ regarding two GO terms X and Y , the minimum support is the minimum probability of an arbitrary transaction (e.g. the set of GO terms of a protein) contains both X and Y , and the minimum confidence is a conditional probability that a transaction having X also contains Y . We use the five-fold cross-validation on the GO terms in the Swiss-Prot database to optimize the two parameters. The performance of using different values of minimum support and minimum confidence is shown in **Table 3.1**. We first fix the minimum confidence at 60, and try different minimum support, and the multiplication of precision and recall is maximized when minimum support is 0.1, and it decreases as the minimum support increases. Then we increase the minimum confidence to 70, and try minimum support values less than 0.1, and the multiplication of precision and recall decreases as the minimum support increases. Another finding is that the minimum confidence actually does not influence the multiplication of precision and recall too much. For the same minimum support 0.1, with minimum confidence 60 and 70, the multiplication of precision and recall is 0.079669 and 0.079751 respectively. So we decide to

try larger minimum confidence score, such as 90, and the result shows smaller minimum support has better performance. The number of rules generating for different minimum support values such as 0.02, 0.03, 0.04 is 171817, 120114, 62707, 51356 respectively. Considering the computation complexity related to the number of rules and their similar performance, we finally set minimum support as 0.05 and minimum confidence as 90.

Table 3.1: The precision, recall, and multiplication of precision and recall for different values of minimum support and confidence according to five-fold cross validation.

Min support	Min confidence	Precision	Recall	Multiplication
0.1	60	0.175247	0.454611	0.079669
0.2	60	0.175762	0.294839	0.051821
0.3	60	0.178529	0.240628	0.042959
0.4	60	0.178	0.217349	0.038688
0.5	60	0.180751	0.203954	0.036865
0.6	60	0.184234	0.194982	0.035922
0.7	60	0.185663	0.179099	0.033252
0.8	60	0.187923	0.176552	0.033178
0.9	60	0.191136	0.166148	0.031757
1	60	0.193348	0.155527	0.030071
0.02	70	0.189585	0.575122	0.109035
0.03	70	0.19235	0.552382	0.10625
0.05	70	0.19523	0.504344	0.098463
0.1	70	0.193433	0.41229	0.079751
0.15	70	0.19347	0.296692	0.057401
0.1	80	0.205309	0.357896	0.073479
0.15	80	0.206317	0.242143	0.049958
0.02	90	0.218213	0.48637	0.106133
0.03	90	0.219549	0.461519	0.101326
0.04	90	0.220407	0.4356	0.096009
0.05	90	0.221515	0.415496	0.092039
0.06	90	0.221194	0.392394	0.086795
0.07	90	0.221575	0.378077	0.083773
0.08	90	0.22069	0.361477	0.079774
0.09	90	0.219519	0.339378	0.0745
0.1	90	0.219174	0.320815	0.070314
0.15	90	0.223325	0.207827	0.046413

3.4.2 Prediction Performance

We evaluate the performance of our method on CAFA1 datasets. CAFA released 48,298 protein targets in total, and 436 of them whose function deposited in Swiss-Prot database are used for our evaluation. Different threshold from 1 to 0.01 decreased by 0.01 is used as thresholds on predicted GO term scores. The predictions with confidence score higher than the threshold will be selected to compare with the true GO terms (threshold metric). Based on this metric, we evaluate the performance of MIS score and how the score scaling technique influences the performance. The precision and recall metrics are used to evaluate the performance of the prediction. Here, in evaluating the performance of our methods on CAFA1 datasets, all predicted and actual GO terms are propagated to the root of the Gene Ontology Directed Acyclic Graph (DAG). All the GO terms in the paths of predicted GO terms toward the root were considered as predicted GO terms, and all the GO terms present in the paths of the actual GO terms toward the root were considered as true GO terms. The overlapping GO terms between predicted and true GO terms are considered as correct predictions. The precision is calculated by the total number of correct predictions divided by the total number of predicted GO terms, and the recall is calculated by the total number of correct predictions divided by the total number of true GO terms. These two metrics are complementary to evaluate the performance of a method from different perspective. The result is shown in **Figure 3.2A**. We test two different score scaling techniques. One is scaled from 1, which sets the starting score to 1. Another is scaled from max, which sets the starting score to the maximum score among all predictions. **Figure 3.2(A)** shows that the MIS score gets similar precision for the recalls in the range of 0.5 and 0.75, but the precision drops drastically when the recall is larger than 0.75. That is because a lot of false-positive predictions are made at a low threshold. Comparing the two score scaling techniques, scaling from 1 has better performance with higher precision, and finally they both can reach

a similar high recall 0.85. Comparing the MIS score with and without score scaling, they both can reach a high recall, but the one with score scaling can reach a higher precision, and the precision decreases more smoothly as recall increases. We calculate the maximum multiplication of precision and recall. MIS score with and without score scaling get 0.239 and 0.231 respectively, suggesting applying score scaling technique slightly improve the performance.

It is interesting to compare the performance of the MIS score and the SEQ score. **Figure 3.2(B)** demonstrates the performance difference of between the two scores. The SEQ score has relatively low precision because it usually makes more predictions and at the same time it can reach a relatively high recall for the same reason. And the SEQ scores with and without scaling techniques have similar performance. **Figure 3.2(C)** illustrates the performance of combining all three different scores by the SMISS predictor. The SMISS predictor outperforms the MIS predictor in both recall and precision. The SMISS can reach a very high recall probably because of the contribution of the SEQ score.

Moreover, we compare the SMISS predictor with three standard baseline methods (Prediction57, Prediction58, and Prediction59) and three predictors (Prediction1, Prediction2, and Prediction 3) that integrates profile-sequence homology search, profile -profile homology search and domain co-occurrence network [41]. Prior method is used for Prediction57, which selects 836 most frequent GO terms counted from the SwissProt database for each target as prediction [41]. Prediction58 is based on BLAST method, which uses the tool BLAST [43] to search the target protein against groups of proteins for predictions [41]. The third baseline method for Prediction59 is GOtcha method [44], which generates the sum of the negative logarithm of the e-values resulted from the BLAST search (GOtcha I-Scores) as the confidence score for GO terms selection [41]. The result is shown in **Figure 3.2(B)**. The three predictors (Prediction1, Prediction2, and Prediction 3) perform better mostly than the standard

baseline methods (Prediction57, Prediction58, and Prediction59). Although the precision of the SMISS predictor is not as high as other methods, it can reach a higher recall than other methods because it can make more GO term predictions. In order to balance both precision and recall, we use F-measure to compare these methods. The maximum F-measure of our SMISS predictor is 0.500, much higher than 0.269, 0.211, and 0.289 of Prediction57, Prediction58, and Prediction59. In addition, it is also higher than 0.347, 0.302, and 0.310 of Prediction1, Prediction2, and Prediction3 (See **Figure 3.3**).

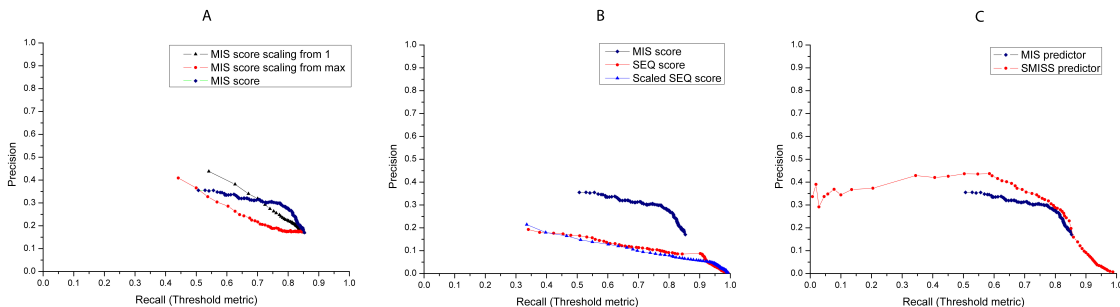


Figure 3.2: The performance comparison for MIS, SEQ, and SMISS using scaled technique benchmarked on CAFA1. X-axis shows the recall of the prediction, and y-axis shows the precision of the predictione2

3.4.3 Case study

We randomly select few proteins whose function is released recently, and submit the query protein sequence in our protein function prediction website to test the usefulness of our method. We only keep the predictions which have confidence score more than 0.9, so that our prediction is not influenced by some random predictions which has low confidence score. **Table 3.2** shows the summary of PDB ids with their true functions and the protein function predictions by our methods used in case study. The first case is 4OPY, which is released at 05/20/2015, and the UniProtKB id is Q9AGJ5. This protein has four GO functions: GO:0030655, GO:0046677,

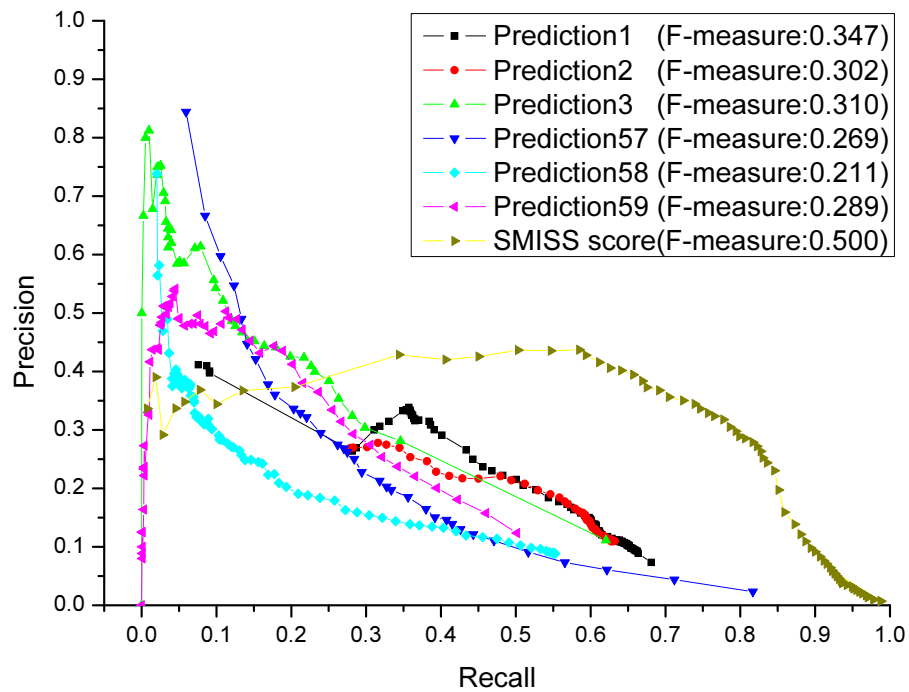


Figure 3.3: The performance of our SMISS with three standard baseline method and three predictors from an automated three-level method. Prediction 57, 58, 59 is the standard baseline method, and Predictors 1, 2, 3 is three predictors from an automated three-level method. X-axis shows the recall for each predictor, and y-axis shows the precision for each predictor.

GO:0008800, and GO:0016787. Our SMISS predictor successfully predict three of them (GO:0030655, GO:0046677, and GO:0008800), so that the precision is 1, and recall is 0.75. In addition to the three GO function predicted by SMISS predictor, the MIS predictor also predicts the function GO:0033251, which is considered as true while propagating the function GO:0016787 to the root. The MIS predictor predicts 12 functions in total for this protein, so that the precision is 0.33, and recall is 1. The MIS-NET predictor only predicts 8 functions, including all true prediction by MIS predictor, so the precision is 0.5, and recall is 1. The SMISS predictor actually makes more function predictions, but only few of them could have confidence score more than 0.9, since our combination process finally assigns high confidence score to the predictions which are predicted from different sources on consensus. The defect for SMISS predictor is that it sometimes misses few true predictions because of its high standard, for example, the function GO:0033251 is not assigned as confidence score more than 0.9 for SMISS predictor, but it is predicted by MIS and MIS-NET predictor. The second case is 4O7V, which is released at 12/31/2014, and the UniProtKB is O57978. There are five GO functions: GO:0006164, GO:0006189, GO:0000166, GO:0004639, and GO:0005524. The MIS predictor successfully predicts four of them (GO:0006164, GO:0006189, GO:0004639, and GO:0005524), missing the function GO:0000166. It makes 15 function predictions, so the precision is 0.27, and recall is 0.80. The MIS-NET predictor has 14 function predictions for this protein, and three of them (GO:0006189, GO:0004639, and GO:0005524) are correct. The confidence score of GO:0006164 by MIS-NET predictor is not more than 0.9 since it is not found from the network, making the precision as 0.6, and recall as 0.21. The SMISS predictor combines the prediction from three different sources, so it also misses the function GO:0006164. It only makes three function predictions with confidence score more than 0.9, and successfully predicts the function GO:0006189, GO:0004639, and GO:0005524. The precision for SMISS predictor is 1, and recall is 0.60. Once

we consider the F-measure, which is the multiplication of precision and recall, we can see that the F-measure for MIS, MIS-NET, and SMISS predictor is 0.22, 0.13, and 0.6 respectively. As is shown, the SMISS predictor combines different sources, even though it may miss some true functions, it is still very useful considering both precision and recall. The MIS and MIS-NET predict more functions with high confidence score, so that it can cover more true GO functions.

Table 3.2: Summary of PDB ids with their true functions and the protein function predictions by our methods for case study.

PDB id	True functions	SMISS prediction/score	MISprediction score	/	MIS-NET prediction/score	
4OPY	GO:0030655	GO:0030655/1.00	GO:0030655/1.00		GO:0030655/1.00	
	GO:0046677	GO:0046677/1.00	GO:0046677/1.00		GO:0046677/1.00	
	GO:0008800	GO:0008800/1.00	GO:0008800/0.99		GO:0008800/1.00	
	GO:0016787			GO:0005886/0.98		GO:0005886/0.99
				GO:0005576/0.97		GO:0005576/0.98
				GO:0042597/0.96		GO:0042597/0.97
				GO:0033251/0.95		GO:0033251/0.96
				GO:0033250/0.95		GO:0033250/0.96
				GO:0008360/0.94		
				GO:0009252/0.94		
4O7V	GO:0006164	GO:0006189/1.00	GO:0006189/1.00		GO:0006189/1.00	
	GO:0006189	GO:0004639/1.00	GO:0004639/1.00		GO:0004639/1.00	
	GO:0000166	GO:0005524/1.00	GO:0005524/1.00		GO:0005524/1.00	
	GO:0004639			GO:0004638/0.99		GO:0005737/0.99
				GO:0034023/0.99		GO:0005829/0.98
				GO:0005829/0.98		GO:0016020/0.97
				GO:0006144/0.97		GO:0003735/0.96
				GO:0006164/0.96		GO:0006412/0.96
				GO:0009113/0.95		GO:0005886/0.95
				GO:0005737/0.94		GO:0003677/0.94
				GO:0004357/0.93		GO:0006351/0.93
				GO:0006163/0.93		GO:0019843/0.92
				GO:0005634/0.92		GO:0008270/0.91
				GO:0016020/0.91		GO:0046872/0.90
				GO:0000082/0.90		

3.5 Conclusion

In this work, we develop a novel protein function prediction system - SMISS. SMISS integrates information from different sources to improve protein function prediction. Given a protein sequence, it generates a list of Gene Ontology (GO) function terms based on the known function annotations of the homologous proteins found by PSI-BLAST. The set of GO terms is then expanded according to the association rules between GO terms learned by mining the Swiss-Prot database, and then the GO terms are further augmented by the function annotations of the neighboring proteins or genes found in protein-protein interaction networks and the novel spatial gene-gene interaction networks of the human genome constructed from the Hi-C chromosomal conformation data of the genome. Finally, the protein sequence is cut into sequence fragments with a length of 5, and more GO terms are predicted from these fragments. The information is measured by three different probabilistic scores (MIS, SEQ, and NET score) respectively and is combined by SMISS for protein function prediction. Based on the test on the protein targets in the 2011 Critical Assessment of Function Annotation (CAFA), SMISS performs better than the baseline methods and other methods of combining profile-sequence search, profile-profile search, and domain co-occurrence networks. SMISS is an open system, which can combine the information from other sources not used in this work. Our future direction is to expand our current system to include other information such as gene expression and genomic location information, and also improve the current method, for example, control potential degeneration of created profiles in PSI-BLAST to improve the MIS score, and search better weight to combine different scores to improve the method.

Chapter 4

Designing and evaluating the MULTICOM protein local and global model quality prediction methods in the CASP10 experiment

4.1 Abstract

Protein model quality assessment is an essential component of generating and using protein structural models. During the Tenth Critical Assessment of Techniques for Protein Structure Prediction (CASP10), we developed and tested four automated methods (MULTICOM-REFINE, MULTICOM-CLUSTER, MULTICOM-NOVEL, and MULTICOM-CONSTRUCT) that predicted both local and global quality of protein structural models. MULTICOM-REFINE was a clustering approach that used the average pairwise structural similarity between models to measure the global quality and the average Euclidean distance between a model and several top ranked models to measure the local quality. MULTICOM-CLUSTER and MULTICOM-

NOVEL were two new support vector machine-based methods of predicting both the local and global quality of a single protein model. MULTICOM-CONSTRUCT was a new weighted pairwise model comparison (clustering) method that used the weighted average similarity between models in a pool to measure the global model quality. Our experiments showed that the pairwise model assessment methods worked better when a large portion of models in the pool were of good quality, whereas single-model quality assessment methods performed better on some hard targets when only a small portion of models in the pool were of reasonable quality. Since digging out a few good models from a large pool of low-quality models is a major challenge in protein structure prediction, single model quality assessment methods appear to be poised to make important contributions to protein structure modeling. The other interesting finding was that single-model quality assessment scores could be used to weight the models by the consensus pairwise model comparison method to improve its accuracy.

4.2 Introduction

Predicting protein tertiary structure from amino acid sequence is of great importance in bioinformatics and computational biology [72, 43]. During the last few decades, a lot of protein tertiary structure prediction methods have been developed. One of them is template-based methods [37, 73, 13, 74, 75], which use known experimentally determined structures as templates, and build structural models for a target protein without known structure. Another of them is template-free methods [9, 76], which do not use a structural template, and fold a protein from scratch. The two kinds of methods were often combined to handle a full spectrum of protein structure prediction problems ranging from relatively easy homology modeling to hard de novo prediction [10, 77, 78, 19]. During protein structure prediction, one important task

is to assess the quality of structural models produced by protein structure prediction methods. A model quality assessment (QA) method employed in a protein structure prediction pipeline is critical for ranking, refining, and selecting models [73]. A model quality assessment method can generally predict a global quality score measuring the overall quality of a protein structure model and a series of local quality scores measuring the local quality of each residue in the model. A global quality score can be a global distance test (GDT-TS) score [17, 18, 79] that is predicted to be the structural similarity between a model and the unknown native structure of a protein. A local quality score of a residue can be the Euclidean distance between the position of the residue in a model and that in the unknown native structure after they are superimposed. In general, protein model quality assessment methods can be classified into two categories: multi-model methods [62, 31, 80, 81, 24] and single-model methods [82, 83, 84, 31, 81]. Multi-model methods largely use a consensus or clustering approach to compare one model with other models in a pool of input models to assess its quality. Generally, a model with a higher similarity with the rest of models in the pool receives a higher global quality score. The methods tend to work well when a large portion of models in the input pool are of good quality, which is often the case for easy to medium hard template-based modeling. Multi-model methods tend to work particularly well if a large portion of good models were independently generated by a number of independent, diverse protein structure prediction methods as seen in the CASP (the Critical Assessment of Techniques for Protein Structure Prediction) experiments, but they worked less well when being applied to the models generated by one single protein structure prediction method because they prefer the average model of the largest model cluster in the model pool. And multi-model methods tend to completely fail if a significant portion of low quality modes are similar to each other and thus dominate the pairwise model comparison as seen in some cases during the 10th CASP experiment (CASP10) held in 2012. Single-model methods strive to

predict the quality of a single protein model without consulting any other models [34, 65, 85, 86, 87]. Despite the performance of single-model methods is still lagging behind the multi-model methods in most cases when most models in the pool of good quality [50, 88], because of their capability of assessing the quality of one individual model, they have potential to address one big challenge in protein structure modeling selecting a model of good quality from a large pool consisting of mostly irrelevant models. Furthermore, as the performance of multi-model quality assessment methods start to converge, single-model methods appear to have a large room of improvement as demonstrated in the CASP10 experiment. In order to critically evaluate the performance of multi-model and single-model protein model quality assessment methods, the CASP10 experiment was designed to assess them in two stages. On Stage1, 20 models of each target spanning a wide range of quality were used to assess the sensitivity of quality assessment methods with respect to the size of input model pool and the quality of input models. On Stage2, about top 150 models selected by a naive consensus model quality assessment method were used to benchmark model quality assessment methods' capability of distinguishing relatively small differences between more similar models. The new settings provided us a good opportunity to assess the strength and weakness of our multi-model and single-model protein model quality assessment methods in terms of accuracy, robustness, consistency and efficiency in order to identify the gaps for further improvement. The rest of the paper is organized as follows. In the Result and Discussion section, we analyze and discuss their performance on the CASP10 benchmark. In the Conclusion section, we summarize this work and conclude it with the directions of future work. In the Method section, we introduce the methods in our protein model quality assessment servers tested in CASP10.

4.3 Methods

4.3.1 Protein Model Quality Prediction Methods

The methods used by the four automated protein model quality assessment servers are briefly described as follows.

MULTICOM-REFINE is a multi-model quality assessment method using a pairwise model comparison approach (APOLLO)[33] to generate global quality scores. The 19 top models based on the global quality scores and the top 1 model selected by SPICKER[89] formed a top model set for local quality prediction. After superimposing a model with each model in the top model set, it calculated the average absolute Euclidean distance between the position of each residue in the model and that of its counterpart in each model in the top model set. The average distance was used as the local quality of each residue.

MULTICOM-CLUSTER is a single-model, support vector machine (SVM)-based method initially implemented in [86]. The input features to the SVM include a window of amino acids encoded by a 20-digit vector of 0 and 1 centered on a target residue, the difference between secondary structure and solvent accessibility predicted by SCRATCH[90] from the protein sequence and that of a model parsed by DSSP[91], and predicted contact probabilities between the target residue and its spatially neighboring residues. The SVM was trained to predict the local quality score (i.e. the Euclidean distance between its position in the model and that in the native structure) of each residue. The predicted local quality scores of all the residues was converted into the global quality score of the model according to the formula[92] as follows:

$$\text{Global quality score} = \frac{1}{L} \sum_{i=1}^t \left(\frac{1}{1 + (\frac{S_i}{T})^2} \right).$$

In the formula, L is the total number of residues, S_i is the local quality score of residue

i, and T is a distance threshold set to set to 5 Angstrom. Residues that did not have a predicted local quality score were skipped in averaging.

MULTICOM-NOVEL is the same as **MULTICOM-CLUSTER** except that amino acid sequence features were replaced with the sequence profile features. The multiple sequence alignment of a target protein used to construct profiles was generated by PSI-BLAST[43].

MULTICOM-CONSTRUCT uses a new, weighted pairwise model evaluation approach to predict global quality. It uses ModelEvaluator [82] an ab initio single-model global quality prediction method to predict a score for each model and TM-score to get the GDT-TS score for each pair of models. The predicted global quality score of a model i is the weighted average GDT-TS score between the model and other models, calculated according to the formula: $S_i = \sum_{j=1}^N (X_{i,j} * \frac{W_j}{\sum_{j=1}^N W_j})$. In this formula, S_i is the predicted global quality score for model i, N is the total number of models, $X_{i,j}$ is the GDT-TS score between model i and model j, W_j is the score for model j predicted by ModelEvaluator, which is used to weight the contribution of $X_{i,j}$ to S_i . In case that no score was predicted for a model by ModelEvaluator, the weight of the model is set to the average of all the scores predicted by ModelEvaluator. The local quality prediction of **MULTICOM-CONSTRUCT** is the same as **MULTICOM-NOVEL** except that additional SOV (segment overlap measure of secondary structure) score features were used by the SVM to generate the local quality score.

4.3.2 Evaluation Methods

CASP10 used two-stage experiments to benchmark for model quality assessment. Stage1 had 20 models with different qualities for each target, and Stage2 had 150 top models for each target selected from all the models by a naive pairwise model quality assessment method. We download the native structures of 98 CASP10 targets, their

structural models, and the quality predictions of these models made by our four servers during the CASP10 experiment running from May to August, 2012 from the CASP website (<http://predictioncenter.org/casp10/index.cgi>).

We used TM-score[92] to calculate the real GDT-TS scores between the native structures and the predicted model as their real global quality scores. The predicted global quality scores of our four servers were used to compare with the real global quality scores. In order to calculate real local quality scores of residues in a model, we first used TM-score to superimpose the native structure and the model, and then calculate the Euclidean distance between each residue's coordinates in the superimposed native structure and the model as the real local quality score of the residue. The real local and global quality scores of a model were compared with that predicted by the model quality assessment methods to evaluate their prediction accuracy.

We evaluated the global quality of our predictions from five aspects: the average of per-target Pearson correlations, the overall Pearson's correlation, average GDT-TS loss, the average Spearman's correlation, and the average Kendall tau correlation. The average of per-target Pearson's correlations is calculated as the average of all 98 targets' Pearson correlations between predicted and real global quality scores of their models. The overall Pearson's correlation is the correlation between predicted and real global quality scores of all the models of all the targets pooled together. The average GDT-TS loss is the average difference between the GDT-TS scores of the real top 1 model and the predicted top 1 model of all targets, which measures how well a method ranks good models at the top. The Spearman's correlation is the Pearson's correlation of the ranked global quality scores. In order to calculate the Spearman's rank correlation, we first convert the global quality scores into the ranks. The identical values (rank ties or duplicate values) are assigned a rank equal to the average of their positions in the rank list. And then we calculate the Pearson's correlation between the predicted ranks and true ranks of the models. The Kendall tau correlation is the

probability of concordance minus the probability of discordance. For two vectors x and y with global quality scores of n models of a target, the number of total possible model pairs for x or y is $N = \frac{(n*(n-1))}{2}$. The number of concordance is the number of pairs (x_i, y_i) and (x_j, y_j) when $(x_i - x_j) * (y_i - y_j) > 0$, and the number of discordance is the number of pairs (x_i, y_i) and (x_j, y_j) when $(x_i - x_j) * (y_i - y_j) < 0$. The Kendall tau correlation is equal to the number of concordance minus the number of discordance divided by N . (http://en.wikipedia.org/wiki/Kendall_tau_rank_correlation_coefficient).

The accuracy of local quality predictions was calculated as the average of the Pearson’s correlations between predicted local quality scores and real local quality scores of all the models of all the targets. For each model, we used TM-score to superimpose it with the native structure, and then calculated the Euclidean distance between Ca atom’s coordinates of each residue in a superimposed model and the native structure as the real local quality score of each residue. The Pearson’s correlation between the real quality scores and the predicted ones of all the residues in each model was calculated. The average of the Pearson’s correlations of all the models for all 98 targets was used to evaluate the performance of the local quality prediction methods.

4.4 Results and Discussions

4.4.1 Results of global quality predictions

The results of the global quality evaluation on Stage1 of CASP10 are shown in **Table 4.1**. We evaluate the global quality in five aspects, and the details for each evaluation methods are described in the evaluation methods part. As shown in the table, the weighted pairwise model comparison method MULTICOM-CONSTRUCT performed best among all the four servers according to all the five measures, suggest-

ing using single-model quality prediction scores as weights can improve the multi-model pairwise comparison based quality prediction methods such as MULTICOM-REFINE. The two multi-model global quality assessment methods had the better average performance than the two single-model global quality assessment methods (MULTICOM-NOVEL and MULTICOM-CLUSTER) on average on Stage1, suggesting that the advantage of multi-model methods over single-model methods was not much affected by the relatively small size of input models (i.e. 20). Instead, the multi-model methods still work reasonably well on a small model pool that contains a significant portion of good quality models. It is worth noting that the average loss of the two single-model quality assessment methods (MULTICOM-CLUSTER and MULTICOM-NOVEL) is close to that of the two multi-model quality assessment methods (MULTICOM-REFINE and MULTICOM-CONSTRUCT) (i.e. +0.07 versus +0.06). To make comparison with other methods, we also add the global quality of the naive consensus method DAVIS-QAconsensus, Pcons, and ModFOLDclust2 on Stage1 of casp10. As we can see in the table, our MULTICOM-CONSTRUCT performs better than the naive consensus method DAVIS-QAconsensus and ModFOLDclust2 on Stage1 of casp10 based on our evaluation, and Pcons performs best among all methods.

Table 4.2 shows the global quality evaluation results on Stage2. Similarly as in **Table 4.1**, the weighted pairwise comparison multi-model method (MULTICOM-CONSTRUCT) performed better than the simple pairwise multi-model method (MULTICOM-REFINE) and both had better performance than the two single-model quality assessment methods (MULTICOM-CONSTRUCT and MULTICOM-NOVEL). That the two single-model quality prediction methods yielded the similar performance indicated that some difference in their input features (amino acid sequence versus sequence profile) did not significant affect their accuracy. In comparison with Stage1, all the methods performed worse on Stage2 models. Since the models in

Stage2 are more similar to each other than in Stage1 in most cases, the results may suggest that both multi-model and single-model quality assessment methods face difficulty in accurately distinguishing models of similar quality. In addition, the performance of naive consensus method DAVIS-QAconsensus, Pcons, and ModFOLDclust2 is also available in the table. Our MULTICOM-CONSTRUCT gets similar performance comparing with DAVIS-QAconsensus and Pcons, and has higher average correlation than ModFOLDclust2 on Stage2 based on our evaluation.

Table 4.1: The average correlation (Ave. Corr.), overall correlation (Over. Corr.), average GDT-TS loss (Ave. loss), average Spearman’s correlation (Ave. spearman), average Kendall tau correlation (Ave. Kendall) of MULTICOM servers, DAVIS-QAconsensus, Pcons, and ModFOLDclust2 on Stage1 of CASP10.

Stage1 of CASP10	Ave. Corr.	Over. Corr.	Ave. loss	Ave. Spearman	Ave. Kendall
MULTICOM-REFINE	0.6494	0.8162	0.0615	0.5989	0.4908
MULTICOM-CLUSTER	0.5144	0.5946	0.0727	0.4364	0.3273
MULTICOM-NOVEL	0.5016	0.4848	0.0791	0.4483	0.338
MULTICOM-CONSTRUCT	0.6838	0.83	0.0613	0.6182	0.5043
DAVIS-QAconsensus	0.6403	0.7927	0.0537	0.5798	0.4745
Pcons	0.7501	0.7683	0.0327	0.6781	0.5457
ModFOLDclust2	0.6775	0.8301	0.0572	0.6206	0.5064

Table 4.1 and **Table 4.2** show there is some difference of our four quality assessment servers. We calculate the wilcoxon signed ranked sum test between all our four servers and our servers against other three methods (DAVIS-QAconsensus, Pcons, and ModFOLDclust2) on Stage1 and Stage2, and the result is shown in **Table 4.3**. As we can see in the table, on Stage1, there are two pairs of servers with the P-value greater than 0.01: MULTICOM-REFINE and MULTICOM-CONSTRUCT, MULTICOM-CLUSTER and MULTICOM-NOVEL. It shows that the difference of average correlation between these two pairs of servers on Stage1 is not statistically significant. However, on Stage2, only the difference of average correlation between MULTICOM-CLUSTER and MULTICOM-NOVEL is larger than 0.01, all other pairs

Table 4.2: The average correlation, overall correlation, average GDT-TS loss, average Spearman’s correlation, average Kendall tau correlation of MULTICOM servers, DAVIS-QAconsensus, Pcons, and ModFOLDclust2 on Stage2 of CASP10.

Stage2 of CASP10	Ave. Corr.	Over. Corr.	Ave. loss	Ave. Spearman	Ave. Kendall
MULTICOM-REFINE	0.4743	0.8252	0.0511	0.4763	0.351
MULTICOM-CLUSTER	0.3354	0.6078	0.0675	0.3361	0.2343
MULTICOM-NOVEL	0.335	0.5057	0.0654	0.3394	0.2358
MULTICOM- CONSTRUCT	0.4853	0.8272	0.051	0.4824	0.3566
DAVIS-QAconsensus	0.505	0.8383	0.0499	0.5031	0.3686
Pcons	0.4891	0.8194	0.0416	0.4843	0.3524
ModFOLDclust2	0.4489	0.8337	0.047	0.4621	0.3393

are less than 0.01. Our server MULTICOM-CLUSTER and MULTICOM-NOVEL have P-value less than 0.01 against other three methods on both Stage1 and Stage2, which shows that the difference of average correlation between these two servers and other three methods is statistically significant on both Stage1 and Stage2.

To elucidate the key factors that affect the accuracy of multi-model or single-model quality assessment methods, we plot the per-target correlation scores of each target on Stage2 against the ratio of the average real quality of the largest model cluster in the pool and the average real quality of all the models in the pool in **Figure 4.1**. To get the largest model cluster for each target, we first calculate the GDT-TS score between each pair of models, and then use (1 - the GDT-TS score) as the distance measure to hierarchically cluster the models. Finally, we use a distance threshold to cut the hierarchical tree to get the largest cluster so that the total number of models in the largest cluster is about one third of the total number of models in the pool.

Figure 4.1 shows that the quality prediction accuracy (i.e. per-target correlation scores of each target) positively correlates with the average real quality of the largest model cluster divided by the average real quality of all models for two multi-model methods (MULTICOM-REFINE, MULTICOM-CONSTRUCT), whereas it

Table 4.3: The P-value of pairwise wilcoxon signed ranked sum test for the difference of correlation score between MULTICOM servers on Stage1 and Stage2 of CASP10, and three other methods: DAVIS-QAconsensus, Pcons, and ModFOLDclust2.

MULTICOM servers, DAVIS-QAconsensus, Pcons, and ModFOLDclust2 on Stage1 or Stage2	P-value
MULTICOM-REFINE and MULTICOM-CLUSTER on Stage1	7.55E-05
MULTICOM-REFINE and MULTICOM-NOVEL on Stage1	3.28E-05
MULTICOM-REFINE and MULTICOM-CONSTRUCT on Stage1	0.031
MULTICOM-CLUSTER and MULTICOM-NOVEL on Stage1	0.201
MULTICOM-CLUSTER and MULTICOM-CONSTRUCT on Stage1	3.76E-06
MULTICOM-NOVEL and MULTICOM-CONSTRUCT on Stage1	7.01E-07
MULTICOM-REFINE and Pcons on Stage1	0.1723
MULTICOM-REFINE and ModFOLDclust2 on Stage1	0.578
MULTICOM-REFINE and DAVIS-QAconsensus on Stage1	0.6238
MULTICOM- CLUSTER and Pcons on Stage1	2.87E-08
MULTICOM- CLUSTER and ModFOLDclust2 on Stage1	5.52E-05
MULTICOM- CLUSTER and DAVIS-QAconsensus on Stage1	0.002873
MULTICOM- NOVEL and Pcons on Stage1	5.65E-09
MULTICOM- NOVEL and ModFOLDclust2 on Stage1	2.12E-05
MULTICOM- NOVEL and DAVIS-QAconsensus on Stage1	0.002066
MULTICOM- CONSTRUCT and Pcons on Stage1	0.7492
MULTICOM- CONSTRUCT and ModFOLDclust2 on Stage1	0.01223
MULTICOM- CONSTRUCT and DAVIS-QAconsensus on Stage1	0.0002211
MULTICOM-REFINE and MULTICOM-CLUSTER on Stage2	4.13E-05
MULTICOM-REFINE and MULTICOM-NOVEL on Stage2	3.18E-05
MULTICOM-REFINE and MULTICOM-CONSTRUCT on Stage2	2.44E-05
MULTICOM-CLUSTER and MULTICOM-NOVEL on Stage2	0.658
MULTICOM-CLUSTER and MULTICOM-CONSTRUCT on Stage2	7.75E-06
MULTICOM-NOVEL and MULTICOM-CONSTRUCT on Stage2	5.28E-06
MULTICOM-REFINE and Pcons on Stage2	0.2465
MULTICOM-REFINE and ModFOLDclust2 on Stage2	0.08742
MULTICOM-REFINE and DAVIS-QAconsensus on Stage2	0.4976
MULTICOM- CLUSTER and Pcons on Stage2	1.11E-05
MULTICOM- CLUSTER and ModFOLDclust2 on Stage2	0.001202
MULTICOM- CLUSTER and DAVIS-QAconsensus on Stage2	7.50E-06
MULTICOM- NOVEL and Pcons on Stage2	1.07E-05
MULTICOM- NOVEL and ModFOLDclust2 on Stage2	0.001128
MULTICOM- NOVEL and DAVIS-QAconsensus on Stage2	5.72E-06
MULTICOM- CONSTRUCT and Pcons on Stage2	0.9807
MULTICOM- CONSTRUCT and ModFOLDclust2 on Stage2	0.003362
MULTICOM- CONSTRUCT and DAVIS-QAconsensus on Stage2	9.60E-05

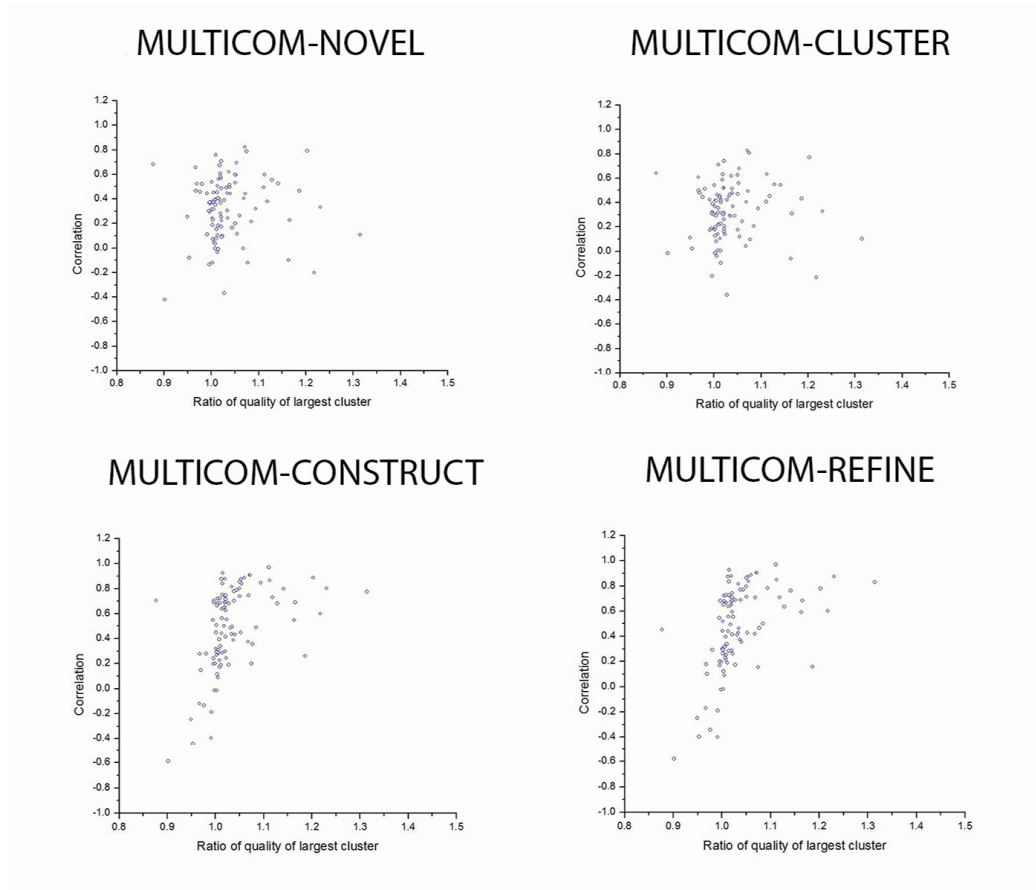


Figure 4.1: The per-target correlation scores of each target against the average real quality of the largest model cluster divided by the average real quality of all models in this target on Stage2

has almost no correlation with single-model methods (MULTICOM-CLUSTER, MULTICOM-NOVEL). The results suggest that the performance of clustering-based multi-model methods depends on the relative real quality of the large cluster of models and that of single-model methods does not. This is not surprising because multi-model methods rely on pairwise model comparison, but single-model methods try to assess the quality from one model.

As CASP10 models were generated by many different predictors from around the world, the side chains of these models may be packed by different modeling tools. The difference in side chain packing may result in difference in input features (e.g. secondary structures) that affect the quality prediction results of single-model methods even though they only try to predict the quality of backbone of a model. In order to remove the side-chain bias, we also tried to use the tool SCWRL[1] to rebuild the side chains of all models before applying a single-model quality prediction method - ModelEvaluator. **Figure 4.2** compares the average correlation and loss of the predictions with or without side-chain repacking. Indeed, repacking side-chains before applying single-model quality assessment increased the average correlation and reduced the loss. We do a wilcoxon signed ranked sum test on the correlation and loss of the predictions before and after repacking side-chains. The P-value for average correlation before and after repacking side-chains on Stage1 is 0.18, and on Stage2 is 0.02. The P-value for loss on Stage1 is 0.42, and on Stage2 is 0.38.

Since mining a few good models out of a large pool of low-quality models is one of the major challenges in protein structure prediction, we compare the performance of single-model methods and multi-model methods on the models of several hard CASP10 template-free targets. **Table 4.4** and **Table 4.5** report the evaluation results of all four servers, DAVIS-QAconsensus, Pcons, and ModFOLDclust2 on all standalone free modeling (FM) targets on Stages 1 and 2, i.e. the targets whose domains are all FM domains. The results show that the single-model methods

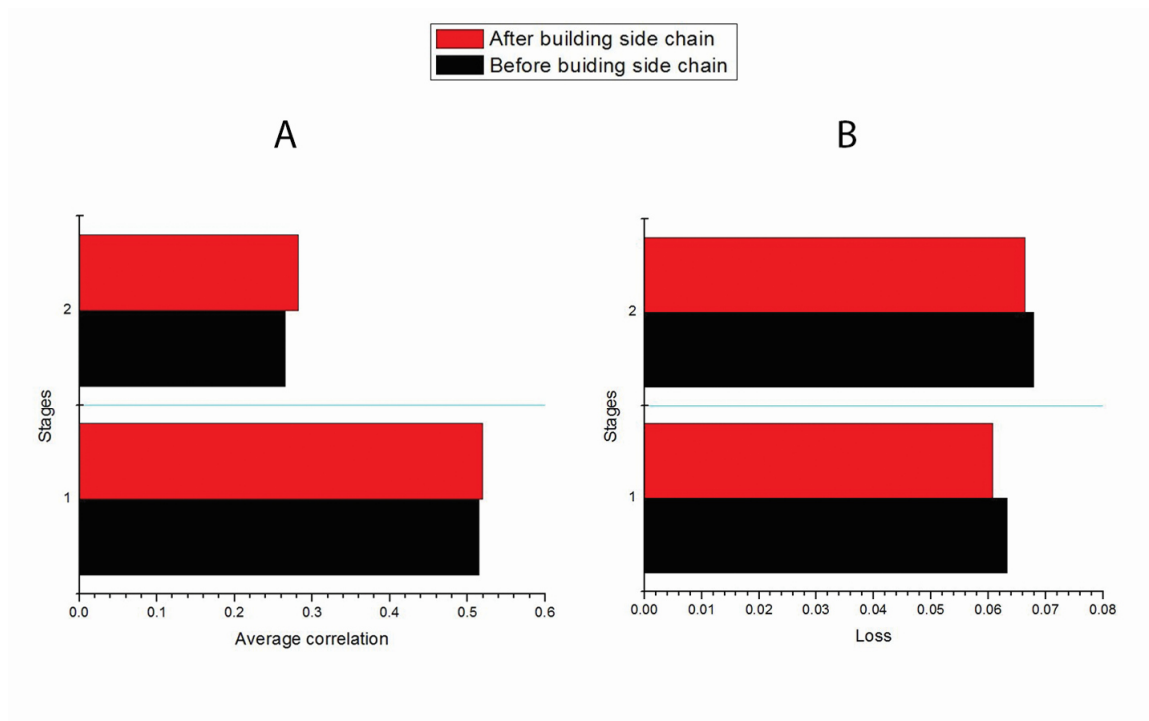


Figure 4.2: The influence of side chain on average correlation and loss of both Stage1 and Stage2. Figure 4.2A shows the average correlation of the predictions with or without side-chain repacking, and Figure 4.2B demonstrates the loss of the predictions with or without side-chain repacking on both Stage1 and Stage2. The tool SCWRL[1] is used for the side-chain repacking.

(MULTICOM-CLUSTER and MULTICOM-NOVEL) clearly performed better than the multi-model methods (MULTICOM-REFINE and MULTICOM-CONSTRUCT) on both stages. And the single-model methods also perform better than the DAVIS-QAconsensus and ModFOLDclust2 on both stages, and get similar performance with Pcons on Stage1, and better performance than Pcons on Stage2. For instance, the average Pearson’s correlation score of MULTICOM-NOVEL on Stage1 is 0.539, which is much higher than 0.082 of MULTICOM-REFINE. The multi-model methods even get low negative correlation for some targets. For example, the Pearson’s correlation score of MULTICOM-REFINE on target T0741 at Stage1 is -0.615. We use the tool TreeView[93] to visualize the hierarchical clustering of the models of T0741 in **Figure 4.3**. The qualities of the models in the largest cluster are among the lowest, but they are similar to each other leading to high predicted quality scores when being assessed by multi-model methods. The example indicates that multi-model methods often completely fail (i.e. yielding negative correlation) when the models in the largest cluster are of worse quality, but similar to each other. Multi-model methods often perform worse than single-model methods when all models in pool are of low quality and are different from each other. In this situation, the quality scores predicted by multi-model methods often do not correlate with the real quality scores, whereas those predicted by single-model methods still positively correlate with real quality scores to some degree. As an example, **Figure 4.4** plots the real GDT-TS scores and predicted GDT-TS scores of a single-model predictor MULTICOM-NOVEL and a multi-model predictor MULTICOM-REFINE on the models of a hard target T0684 whose best model has quality score less than 0.2.

Based on the per-target correlation between predicted and observed model quality scores of the official model quality assessment results [81], the MULTICOM-CONSTRUCT was ranked 5th on Stage2 models of CASP10 among all CASP10 model quality assessment methods. The performance of MULTICOM-CONSTRUCT was

also slightly better than the benchmark DAVIS-QAconsensus (the native consensus method, the quality score of a model is calculated by the average structural similarity GDT-TS score of the model against other models in the model pool) on Stage2, which was ranked at 10th. The methods MULTICOM-REFINE, MULTICOM-NOVEL, and MULTICOM-CLUSTER were ranked at 11th, 28th, and 29th, respectively. However, it was not surprising that the single-model methods such as MULTICOM-NOVEL and MULTICOM-CLUSTER were ranked lower than most clustering-based methods because the latter tended to work better on most CASP template-based targets with good-quality predicted models. But, among all single-model methods, MULTICOM-NOVEL and MULTICOM-CLUSTER were ranked at 3th and 4th. And MULTICOM-CLUSTER and MULTICOM-NOVEL are ranked at 4th and 5th among all single model methods on Stage1 of CASP10 separately.

Table 4.4: Pearson correlation of the FM (template-free modeling) targets on Stage1 of CASP10.

Stage1 of CASP10	MULTICOM-NOVEL	MULTICOM-CLUSTER	MULTICOM-CONSTRUCT	MULTICOM-REFINE	DAVIS-QAconsensus	Pcons	ModFOLD clust2
T0666	0.57	0.454	0.138	0.272	0.274	0.346	0.538
T0735	0.725	0.704	0.414	0.083	0.086	0.667	0.03
T0734	0.522	0.544	0.152	-0.099	-0.096	0.509	-0.014
T0737	0.878	0.878	0.221	0.118	0.124	0.565	0.421
T0740	0.558	0.512	0.71	0.732	0.726	0.684	0.77
T0741	-0.02	0.214	-0.659	-0.615	-0.611	0.475	-0.674
Average	0.539	0.551	0.163	0.082	0.084	0.541	0.179

4.4.2 Results of local quality

Table 4.6 shows the performance of local quality assessment of our four local quality assessment servers, DAVIS-QAconsensus, Pcons, and ModFOLDclust2 on both Stage1 and Stage2. In order to show the statistical significant differences for the per-residue (local) model quality prediction methods, we also calculate the pairwise wilcoxon signed ranked sum test for our four servers and against the other

Table 4.5: Pearson correlation of all FM (template-free modeling) targets on Stage2 of CASP10.

Stage2 of CASP10	MULTICOM-NOVEL	MULTICOM-CLUSTER	MULTICOM-CONSTRUCT	MULTICOM-REFINE	DAVIS-QA consensus	Pcons	ModFOLD clust2
T0666	0.213	0.206	0.49	0.499	0.492	0.338	0.52
T0735	0.466	0.433	0.261	0.159	0.15	0.238	-0.07
T0734	0.459	0.44	-0.134	-0.342	-0.334	0.199	-0.363
T0737	0.787	0.806	0.2	0.155	0.147	0.583	0.525
T0740	0.49	0.451	0.487	0.412	0.411	0.434	0.478
T0741	-0.079	0.022	-0.444	-0.397	-0.397	0.125	-0.382
Average	0.389	0.393	0.143	0.081	0.078	0.32	0.118

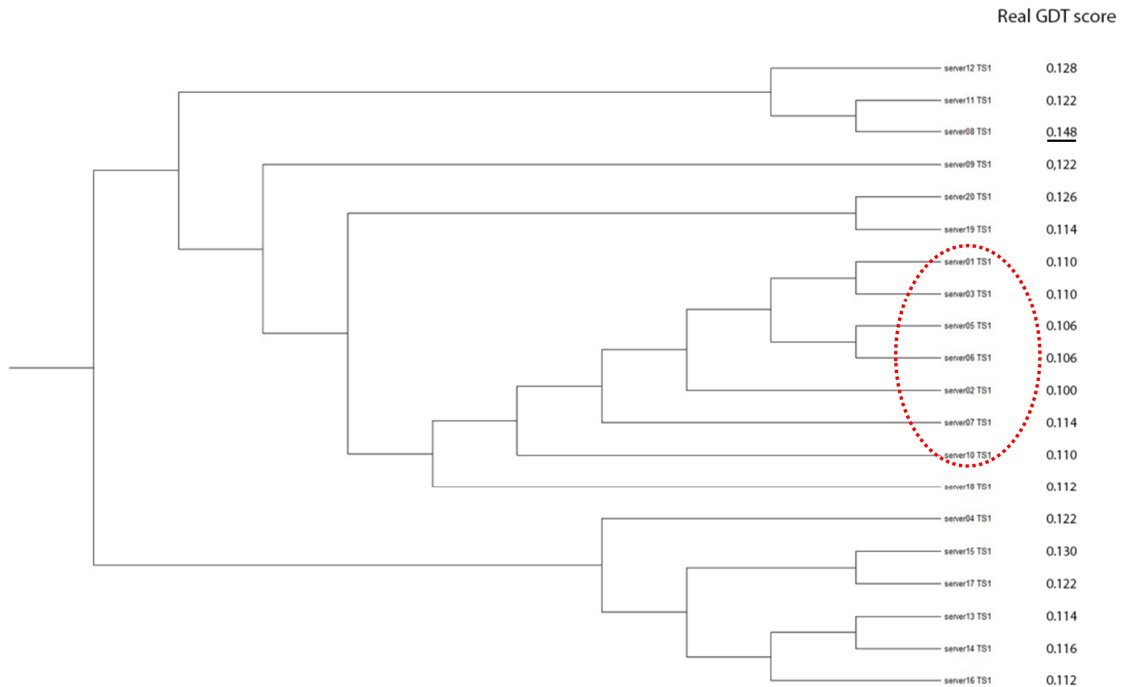


Figure 4.3: The hierarchy tree of T0741 on Stage1. All models in the circle form the largest cluster in this target. The rightmost column of Figure 3 lists the real GDT-TS score of each model. The models in the circle form the largest cluster. The model with the underline real GDT-TS score is the best model in this target

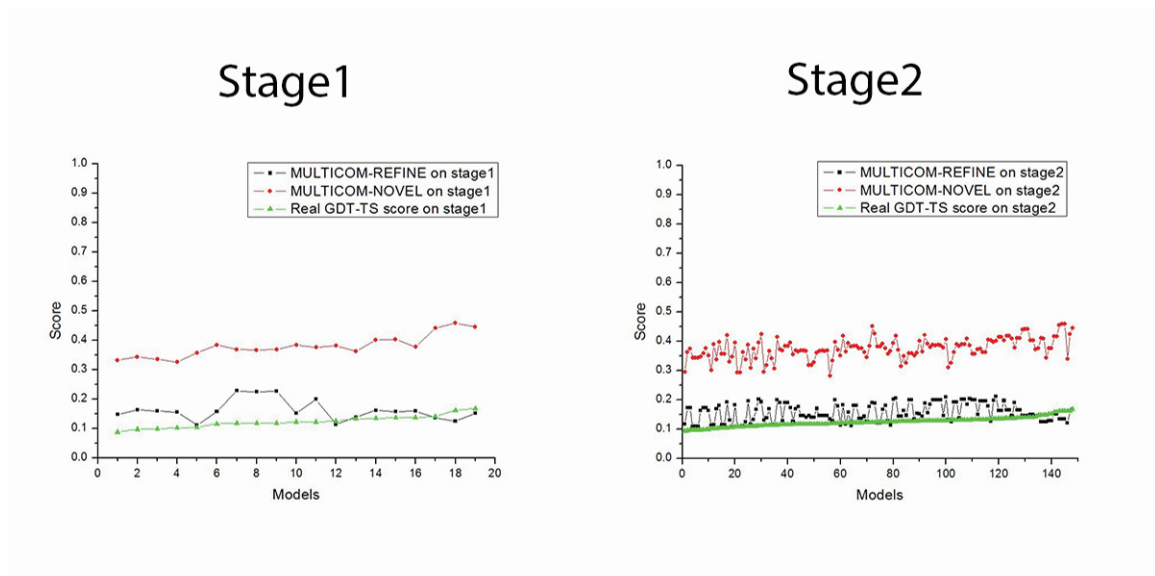


Figure 4.4: The real GDT-TS score and predicted GDT-TS score of MULTICOM-REFINE and MULTICOM-NOVEL for T0684 on Stage 1 and Stage2.

three methods on the local quality, and the result is shown in **Table 4.7**. As we can see from **Table 4.7**, On Stage1, the P-value between MULTICOM-NOVEL and MULTICOM-CONSTRUCT, MULTICOM-REFINE and Pcons, MULTICOM-REFINE and ModFOLDclust2, is larger than 0.01, which shows these pairs are not statistically significant. On Stage2, the P-value between MULTICOM-CLUSTER and MULTICOM-CONSTRUCT, MULTICOM-REFINE and Pcons, is larger than 0.01. **Table 4.6** shows that the multi-model methods performed better than single-model methods on average for all the targets of our four servers. However, the single-model local quality prediction methods (MULTICOM-CONSTRUCT, MULTICOM-NOVEL, MULTICOM-CLUSTER) and the multi-model local quality prediction method (MULTICOM-REFINE) performed not very differently on FM targets as shown in **Table 4.8** and **Table 4.9**. This is not surprising because multi-model methods cannot select real good models as reference methods for evaluating the local quality of residues. According to the CASP official evaluation [81], MULTICOM-REFINE performs best among all of our four servers for the local quality assessment on both Stage1 and Stage2 models of CASP10. Comparing with the naive consensus method DAVIS-QAconsensus, Pcons, and ModFOLDclust2, the multi-model local quality prediction method MULTICOM-REFINE performs best on Stage1, and get similar performance with Pcons on Stage2, but not as good as DAVIS-QAconsensus and ModFOLDclust2 on Stage2.

Table 4.6: Evaluation result of local quality score of four servers, DAVIS- QAconsensus, Pcons, and ModFOLDclust2 on Stage1 and Stage2 of CASP10.

CASP10	Ave. Corr. on Stage1	Ave. Corr. on Stage2
MULTICOM-REFINE	0.6102	0.6251
MULTICOM-CLUSTER	0.2604	0.2956
MULTICOM-NOVEL	0.2882	0.3289
MULTICOM-CONSTRUCT	0.2889	0.3095
DAVIS-QAconsensus	0.5841	0.6633
Pcons	0.5793	0.6226
ModFOLDclust2	0.5997	0.6526

Table 4.7: The P-value of pairwise wilcoxon signed ranked sum test for the difference of correlation score for local model quality between MULTICOM servers on Stage1 and Stage2 of CASP10, and three other methods: DAVIS-QAconsensus, Pcons, and ModFOLDclust2.

MULTICOM servers, DAVIS-QAconsensus, Pcons, and ModFOLDclust2 and on Stage1 or Stage2	P-value
MULTICOM-REFINE and MULTICOM-CLUSTER on Stage1	2.22E-16
MULTICOM-REFINE and MULTICOM-NOVEL on Stage1	6.66E-16
MULTICOM-REFINE and MULTICOM-CONSTRUCT on Stage1	6.66E-16
MULTICOM-CLUSTER and MULTICOM-NOVEL on Stage1	0.0009948
MULTICOM-CLUSTER and MULTICOM-CONSTRUCT on Stage1	0.0008437
MULTICOM-NOVEL and MULTICOM-CONSTRUCT on Stage1	0.1781
MULTICOM-REFINE and Pcons on Stage1	0.01575
MULTICOM-REFINE and ModFOLDclust2 on Stage1	0.2678
MULTICOM-REFINE and DAVIS-QAconsensus on Stage1	0.00699
MULTICOM- CLUSTER and Pcons on Stage1	2.20E-16
MULTICOM- CLUSTER and ModFOLDclust2 on Stage1	2.55E-16
MULTICOM- CLUSTER and DAVIS-QAconsensus on Stage1	2.44E-15
MULTICOM- NOVEL and Pcons on Stage1	2.20E-16
MULTICOM- NOVEL and ModFOLDclust2 on Stage1	3.05E-16
MULTICOM- NOVEL and DAVIS-QAconsensus on Stage1	4.89E-15
MULTICOM- CONSTRUCT and Pcons on Stage1	2.20E-16
MULTICOM- CONSTRUCT and ModFOLDclust2 on Stage1	3.14E-16
MULTICOM- CONSTRUCT and DAVIS-QAconsensus on Stage1	4.78E-15
MULTICOM-REFINE and MULTICOM-CLUSTER on Stage2	2.27E-16
MULTICOM-REFINE and MULTICOM-NOVEL on Stage2	6.66E-16
MULTICOM-REFINE and MULTICOM-CONSTRUCT on Stage2	3.14E-16
MULTICOM-CLUSTER and MULTICOM-NOVEL on Stage2	0.00327
MULTICOM-CLUSTER and MULTICOM-CONSTRUCT on Stage2	0.5493
MULTICOM-NOVEL and MULTICOM-CONSTRUCT on Stage2	1.03E-14
MULTICOM-REFINE and Pcons on Stage2	0.2498
MULTICOM-REFINE and ModFOLDclust2 on Stage2	0.0005575
MULTICOM-REFINE and DAVIS-QAconsensus on Stage2	2.44E-06
MULTICOM- CLUSTER and Pcons on Stage2	2.22E-16
MULTICOM- CLUSTER and ModFOLDclust2 on Stage2	2.22E-16
MULTICOM- CLUSTER and DAVIS-QAconsensus on Stage2	2.22E-16
MULTICOM- NOVEL and Pcons on Stage2	4.44E-16
MULTICOM- NOVEL and ModFOLDclust2 on Stage2	2.22E-16
MULTICOM- NOVEL and DAVIS-QAconsensus on Stage2	2.22E-16
MULTICOM- CONSTRUCT and Pcons on Stage2	4.09E-16
MULTICOM- CONSTRUCT and ModFOLDclust2 on Stage2	2.20E-16
MULTICOM- CONSTRUCT and DAVIS-QAconsensus on Stage2	2.20E-16

Table 4.8: Local quality score of four servers, DAVIS-QAconsensus, Pcons, and ModFOLDclust2 for all FM (template-free modeling) targets on Stage1 of CASP10.

Stage1 of CASP10	MULTICOM-NOVEL	MULTICOM-CLUSTER	MULTICOM-CONSTRUCT	MULTICOM-REFINE	DAVIS-QAconsensus	Pcons	ModFOLDclust2
T0666	0.261	0.216	0.262	0.261	0.195	0.303	0.164
T0735	0.118	0.083	0.122	0.366	0.19	0.214	0.224
T0734	0.025	0.105	0.025	0.402	0.302	0.166	0.232
T0737	0.554	0.664	0.551	0	0.186	0.704	0.122
T0740	0.242	0.196	0.243	0.442	0.368	0.377	0.407
T0741	0.078	-0.035	0.084	0.227	0.108	-0.072	0.136
Average	0.213	0.205	0.215	0.283	0.225	0.282	0.214

Table 4.9: Local quality score of four servers, DAVIS-QAconsensus, Pcons, and ModFOLDclust2 for all FM (template-free modeling) targets on Stage2 of CASP10.

Stage2 of CASP10	MULTICOM-NOVEL	MULTICOM-CLUSTER	MULTICOM-CONSTRUCT	MULTICOM-REFINE	DAVIS-QAconsensus	Pcons	ModFOLDclust2
T0666	0.244	0.226	0.227	0.31	0.322	0.282	0.337
T0735	0.125	0.122	0.127	0.288	0.29	0.15	0.351
T0734	0.129	0.151	0.122	0.172	0.33	0.255	0.305
T0737	0.426	0.578	0.419	0	0.202	0.583	0
T0740	0.268	0.197	0.257	0.27	0.422	0.377	0.425
T0741	0.105	-0.011	0.109	0.165	0.129	0.009	0.119
Average	0.216	0.211	0.21	0.2	0.283	0.276	0.256

Chapter 5

Single-model quality assessment on the assessment of scores from probability density function

5.1 Abstract

Protein quality assessment (QA) has played a very important role in protein structure prediction. We developed a novel single-model quality assessment method Qprob. We calculate the absolute error for each feature value against the true GDT-TS score on CASP9 dataset, and use it to estimate the probability density distribution of each feature for quality assessment. Our method has been blindly tested on the 11th Critical Assessment of Techniques for Protein Structure Prediction (CASP11) as MULTICOM-NOVEL server. The official result from CASP shows that our method ranks as one of the top single-model QA methods. In addition, our method makes big contribution to protein structure human predictor MULTICOM, which is officially ranked 3rd out of 143 predictors. The good performance of our method on template free modeling CASP11 targets shows the good model selection ability of it on hard targets. All of these excellent performance demonstrate that this new prob-

ability density distribution based method is effective and powerful for single-model quality assessment and has a lot of applications for protein structure prediction. The webserver is available at: <http://calla.rnet.missouri.edu/qprob/>. The tools are also available in the webserver.

5.2 Introduction

The number of protein sequences has been generated exponentially during the last few decades because of the application of high-throughput next-generation sequencing technologies [3]. This highlighted the importance of computational methods in bioinformatics and computational biology field, since they are much cheaper and faster than experimental method [12], such as the protein structure prediction, protein function prediction, and etc [7, 9, 12, 37, 41, 94, 77]. A lot of progress has been made recently for protein structure prediction, either template-based methods, or template-free methods. Especially with the help of the Critical Assessment of Techniques for Protein Structure Prediction (CASP), different protein structure prediction methods can be blindly tested and benchmarked. During the prediction of protein structure, one category is crucial, that is protein (model) quality assessment. The model quality assessment problem can be defined as ranking the models without knowing the native structure. The common way of predicting protein structure is to first generate thousands of decoys, and then use the model quality assessment method to select and rank the models. With the rapid developing technology of template-based/template-free modeling, the decoys can be generated by different methods easily. However, selecting and ranking the models is still a very hard problem. In general, there are two different kinds of protein quality assessment (QA) methods: single-model quality assessment [10, 95, 96, 51, 82, 83] and consensus model quality assessment[31, 81, 24]. During the previous CASP experiments, the consensus quality assessment methods usually

perform better than the single-model quality assessment methods, especially when there is a good consensus in the model pool. However, it is also known that the consensus quality assessment may fail badly when there are large portion of bad models in the model pool [10]. In addition, the consensus quality assessment method could not generate a good score when all of them are irrelevant or there are very few models (e.g. Only 1 model in the model pool). Moreover, the computation complexity is also another problem for consensus quality assessment method when there are more than tens of thousands of models. The single-model method could be a good solution for solving this problem. Currently, most single-model method uses the actual model’s information, such as the evolution information [97], residue environment compatibility [38], structural features and physics-based knowledge [96, 51, 82, 83, 34, 39, 98]. Some other methods also tries to combine the single-model and consensus methods, and achieve good performance in the CASP11 [12, 94]. Comparing with other QA methods, first of all, Qprob is a pure single-model QA method, which is different from consensus methods [3, 12, 94]. Also, unlike other single-model QA methods[10, 95, 82] which use unique type of features, Qprob combines structural, physicochemical features and four energy scores. Moreover, there is no QA method that does the error estimation for these features and applies the probability density function based on it for model quality assessment.

In this paper, we benchmarked and normalized four single-model QA energy scores in combining with seven physicochemical and structural features. We normalize the energy scores by benchmarking on PISCES[48] database, and apply the idea similar to EM algorithm to get the best weight for each feature. The probability density function of the error between predicted score and real GDT-TS score is generated. Our assumption is that the error for each predicted score against the real score obeys the normal distribution. By combining the different probability density distribution from each feature, we can predict the global quality score of a model with the high-

est probability. Similar to state-of-the-art single-model QA method performance is achieved when blindly tested our method on CASP11, which demonstrate the powerful of predicting model quality from the probability density distribution.

The paper is organized as follows. In the methods section, we describe each feature and the calculation of the global quality assessment score in detail. In the result section, we describe the performance of our method on CASP11. In the discussion section, we summarize the results and conclude the direction of future works.

5.3 Methods

In this section, we describe how to generate the probability density distribution based on feature error estimation, and use it for global quality assessment. First, we depict the calculation of in total 11 features. Second, we explain the feature errors estimation while the data is benchmarked on CASP9 targets. Third, we report how the weights for each feature are generated. Finally, we describe the probability density function and how it is used for protein quality assessment.

5.3.1 Feature generation

Our method uses a set of features extracted from the structural model and its protein sequence, physicochemical characteristic of the structural model [98], and also four energy scores for predicting the global quality score of a model. The features include:

1. The RF_CB_SRS_OD score [96] is an energy score for evaluating the protein structure based on statistical distance dependent pair potentials. The score is normalized to the range of 0 and 1, which is described in detail at the result section.
2. The secondary structure similarity score is calculated by the difference between

secondary structure predicted by Spine X [99] from the protein sequence and those of a model parsed by DSSP [91].

3. The secondary structure penalty percentage score. This is calculated by the following formula:

$$S_{penalty} = \frac{F_H + F_S}{N}$$

The F_H is the total number of helix secondary structure predicted for each amino acid that is matching with the one parsed by DSSP. The F_S is for the sheet secondary structural matching. N is the sequence length.

4. The Euclidean compact score. This score can be used to describe the compact of the protein model. This is calculated by the following formula:

$$S_{Eucli} = \frac{Eucli(i, j)}{\sum 3.8 * |i - j|}$$

The i and j is the index of amino acids, and $Eucli(i, j)$ is the Euclidean distance of amino acid i and j in the structural model. We ignore the calculation of amino acid with itself.

5. The surface score for exposed nonpolar residues. This score describes the percentage of area of the nonpolar residues exposed, and is calculated as follows:

$$S_{surf} = \frac{\sum SE_i}{\sum S_i}$$

S_i is the exposed area of residue i parsed by DSSP, and SE_i is the exposed area of nonpolar residue i . The SE_i is set to 0 once residue i is polar.

6. The exposed mass score. This score describes the percentage of mass of exposed

residues, and is calculated as follows:

$$S_{mass} = \frac{\sum STN_i * M_i}{\sum S_i * M_i}$$

S_i is the exposed area of residue i parsed by DSSP, STN_i is the total area of nonpolar residue i , and M_i is the total mass of residue i .

7. The exposed surface score. This score describes the percentage of area of the residues exposed, and is calculated as follows:

$$S_{\text{exposed surface}} = \frac{\sum S_i}{\sum ST_i}$$

ST_i is the total area of residue i parsed by DSSP, and S_i is the exposed area of residue i .

8. The solvent accessibility similarity score is calculated by the difference between solvent accessibility predicted SSpro4 [90] from the protein sequence and those of a model parsed by DSSP [91].
9. The RWplus score [51] is an energy score evaluating protein models based on distance-dependent atomic potential. The score is normalized to the range of 0 and 1, which is described in detail at the result section.
10. The ModelEvaluator score [82] is a score evaluating protein models based on structural features and support vector machines.
11. The Dope score [83] is an energy score evaluating protein models based on reference state of non-interacting atoms in homogeneous sphere. The score is normalized to the range of 0 and 1, which is described in detail at the result section.

5.3.2 Feature errors estimation

We calculate all feature scores on 99 CASP9 targets, which in total have 22016 models. We assume the error of all feature scores against real GDT-TS score obeys normal distribution. The feature error is calculated for each model using the following formula:

$$FE_{i,j} = F_{i,j} - R_j$$

$FE_{i,j}$ is the error estimate of feature i on model j , $F_{i,j}$ is the predicted score of feature i on model j , and R_j is the real GDT-TS score of model j . Based on the error estimation of each model, we calculate the mean M_i and standard deviation SD_i for each feature i as follows:

$$\begin{cases} M_i = \frac{\sum_{j=1}^N FE_{i,j}}{N} \\ SD_i = \sqrt{\frac{\sum_{j=1}^N (FE_{i,j} - M_i)^2}{N}} \end{cases}$$

i is in the range of 1 and 11 which represent all 11 features. N is the total number of models.

The feature error estimation results (mean and standard deviation of each feature) can be used for global model quality score assessment based on probability density function.

5.3.3 Feature weight estimation

We have in total 11 features, and the naive way to estimate the weight of each feature by trying all possible weight combination is time consuming. Here, we describe a method similar to EM algorithm for estimating the weight of each feature. There are three steps, as following:

1. Randomly assign a weight to each feature. The weight value is in the range of

-0.8 to 0.8 with the step 0.01, and assign the minimum average GDT-TS loss (Min-Loss) to 1.

2. Expectation step: calculating the per-target average loss using the current weight value set W benchmarked on CASP9 targets. Terminate when the current weight value set W is not changed and all features have been go through the maximization step.
3. Maximization step: trying different weight values for feature i and fixing the weight of all other features. For each weight w for feature i , get the average GDT-TS loss from step (2), and updating the Min-Loss if it is less than the current value of Min-Loss. Also, the current weight value set W is updated once we find a different weight w comparing with the current weight of feature i . Repeat step (3) for the next feature $i+1$, unless it finishes at step (2).
4. After applying this algorithm, we finally get a weight value set W which has the minimum average GDT-TS loss benchmarked on CASP9. The best weight based on our benchmark is as follows for each feature: [0.03, 0.09, 0.04, 0.08, 0.08, 0.01, 0.03, 0.10, 0.00, 0.09, -0.02].

5.3.4 Model quality assessment based on probability density function

Given a protein model, we first calculate feature score Pre_i for each feature i (i is in the range of 1 and 11). And then we calculate the adjusted score (an estimation of the real score) by $Adjust_pre_i = Pre_i - M_i$, while the mean M_i and standard deviation SD_i for each feature i is calculated in the feature errors estimation step. We use the following probability density function of global quality X_i for each feature i (the mean

is $Adjust_pre_i$ and standard deviation is SD_i):

$$P_i(X_i) = \frac{e^{-\frac{(X_i - Adjust_pre_i)^2}{2 * SD_i^2}}}{\sqrt{2\pi} SD_i}$$

We normalize the probability score to convert it into the range of 0 and 1 with the following formula:

$$P_norm_i(X_i) = \frac{P_i(X_i)}{P_i(Adjust_pre_i)}$$

The final global quality score is calculated by combining all 11 normal distributions from each feature prediction. Given a value X in the range of 0 and 1, we calculate the combined probability score as follows:

$$P_combine(X) = \sum_{i=1}^{i=11} (W_i + P_{norm_i}(X))$$

The value X which has the maximum combined probability score $P_combine(X)$ is assigned as the global quality score for the model. Here, the calculation of weight W_i is described in previous section.

5.4 Results

In this section, we first briefly describe the feature processing for our method Qprob, and then describe the feature errors estimation result benchmarked on CASP9 datasets, and finally present an evaluation of its performance on CASP11.

5.4.1 Feature normalization result

We use 11 feature scores in total in our method, and there is no need to do normalization for most of them. However, some features, especially the energy scores are

dependent on the sequence length, and not in the range of 0 and 1. The native structure with long sequence may have different energy score compared with the short one, even though both of them are native structure. So we need to normalize these scores before using them. Here, we use PISCES database to benchmark and normalize three scores (DFIRE2 score, RWplus score, and RF_CB_SRS_OD score) based on the sequence length. The version of PISCES is: the percentage identity cutoff is 20%, the resolution cutoff is 1.8 angstroms, and the R-factor cutoff is 0.25. **Figure 5.1** shows the protein sequence length versus three original energy scores (DFIRE2, RWplus, and RF_CB_SRS_OD scores). We draw the regression line in these figures to fit the protein sequence with the score. The following formula describes the relationship of protein sequence length and the energy score based on the regression line:

$$\begin{cases} \text{Dfire score} = -1.971 * L + 37.746 \\ \text{RWplus score} = -232.6 * L + 6589.5 \\ \text{RF_CB_SRS_OD score} = -0.4823 * L + (-15.9066) \end{cases}$$

L is the protein sequence length. To normalize these scores into the range of 0 and 1, we use the following formula:

$$\begin{cases} \text{Norm_S}_{Dfire} = \frac{-P_{Dfire\ score}}{1.971 * L} \\ \text{Norm_S}_{RWplus} = \frac{-P_{RWplus\ score}}{232.6 * L} \\ \text{Norm_S}_{RF_CB_SRS_OD\ score} = \frac{700 - P_{RF_CB_SRS_OD\ score}}{1000 + 0.4823 * L} \end{cases}$$

$P_{Dfire\ score}$ 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. $P_{Dfire\ score}$ is set to the range of $-1.971 * L$ and 0, $P_{RWplus\ score}$ is set to the range of $-232.6 * L$ and 0, and $P_{RF_CB_SRS_OD\ score}$ is set to the range of $0.4823 * L - 300$ and 700 based on the benchmark of all scores in CASP9 targets so that most of models are in this range.

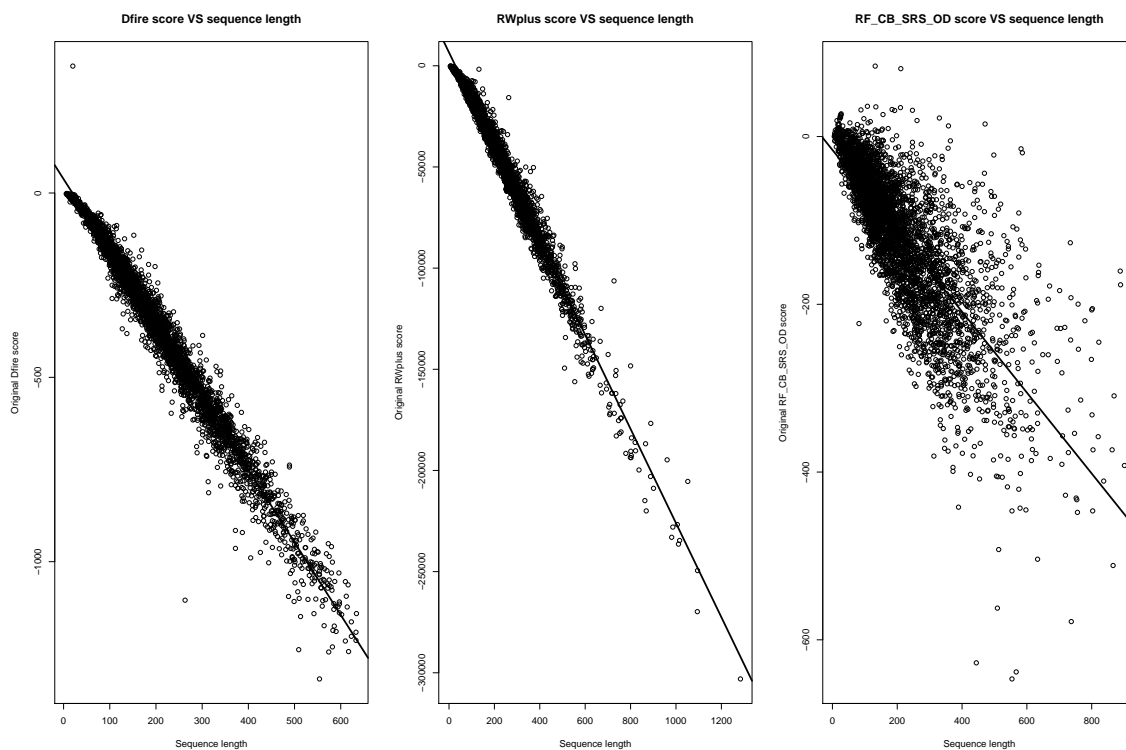


Figure 5.1: The relationship of sequence length and three energy scores (DFIRE2, RWplus, and RF_CB_SRS_OD scores) on PISCES database.

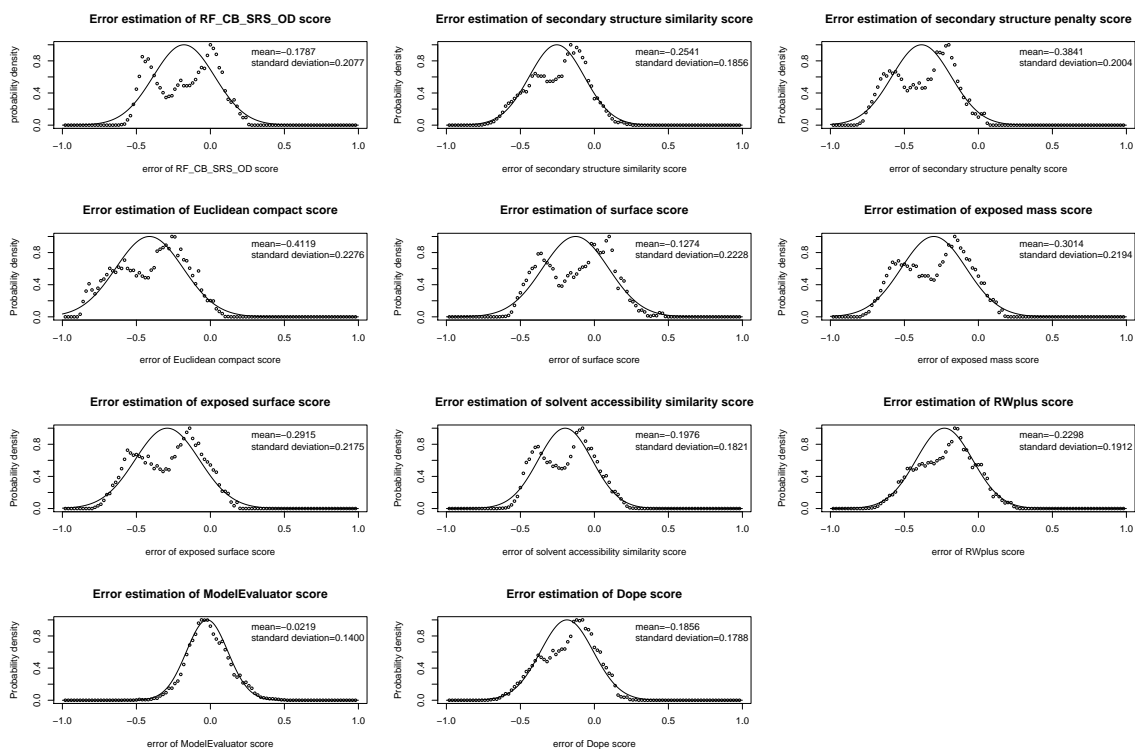


Figure 5.2: The probability density distribution for the error estimation of all 11 feature scores.

5.4.2 Feature error estimation result

We calculate all 11 feature scores (three energy scores are normalized in the previous section) on all models of CASP9 targets, and the error between predicted score and real GDT-TS score for each model are used for the feature error estimation to get the probability density distribution. **Figure 5.2** shows the probability density distribution of all 11 features. The x-axis is the error between predicted score and real GDT-TS score, and the y-axis is the probability density distribution of the error. The mean and standard deviation is also demonstrated in the figures. We use a normal distribution to fit these errors. We can see from the figure that ModelEvaluator score has the mean -0.0219, which is the closest to the real GDT-TS score. In addition, it also has the minimum deviation, which shows it is the most stable feature for evaluating the global model qualities. The Euclidean compact score has the maximum absolute mean (0.4119), showing it is very different from the real GDT-TS score. However, our adjusted score considers the mean of the error estimation, so it may still be very useful for our final prediction.

5.4.3 Global quality assessment result

Our method Qprob is blindly tested on CASP11 as MULTICOM-NOVEL server, and is used for the human predictor MULTICOM (MULTICOM is officially ranked 3rd out of 143 predictors according to the total scores of the first models predicted). According to the analysis result by removing each QA method from MULTICOM, the removal of Qprob causes the biggest decrease in the average Z-score of top one models selected by MULTICOM method (Z-score from 1.364 to 1.321) [12, 94], showing Qprob makes big contribution to MULTICOM. Our method is one of the best single-model QA method based on the CASP official evaluation [42] and our evaluations on **Table 5.1** and **Table 5.1**.

Table 5.1 depicts the per-target average correlation, average GDT-TS loss, average spearman’s correlation, and average kendall tau correlation of our method Qprob and other pure single-model QA methods on Stage 1 (sel20) CASP11 datasets. We also illustrate the p-value of the pairwise Wilcoxon signed ranked sum test for the difference of loss/correlation between Qprob and other pure single-model QA methods to show the significance of differences. The table is ranked by the average GDT-TS loss since the loss metric (the difference of GDT-TS score of the best model and predicted top 1 model) shows the model selection ability of a QA method. From **Table 5.1**, we can see Qprob ranked at third based on the average GDT-TS loss among all pure single-model QA methods on Stage 1 CASP11 datasets. According to 0.01 significant threshold of p-value, there is no significant difference between Qprob and state-of-the-art QA methods ProQ2 and ProQ2-refine on both correlation and loss. The difference on average spearman’s correlation and kendall tau correlation is also small between Qprob and the other two top performing QA methods. Other than CASP11 QA server predictors, we also compare Qprob with five single-model QA scores which are highlighted in bold in **Table 5.1**. The five scores are ModelEvaluator score, Dope score, DFIRE2 score, RWplus score, and RF_CB_SRS_OD score. The result shows Qprob has the best performance on both correlation and loss among these scores. Moreover, the difference of correlation between Qprob and four QA scores (Dope score, DFIRE2 score, RWplus score, and RF_CB_SRS_OD score) is significant, and the difference of loss between Qprob and three QA scores (DFIRE2 score, RWplus score, and RF_CB_SRS_OD score) is significant according to 0.01 significant threshold of p-value. Finally, we also calculate the performance of the baseline consensus QA method DAVIS_consensus (correlation and loss is 0.798 and 0.052 respectively). Not surprisingly, we find out the performance of Qprob is worse than DAVIS_consensus method, and the difference is significant (p-value of correlation and loss is $1.4e-12$ and $1.6e-4$ respectively). The difference is more significant between Qprob and start-of-

the-art consensus QA method Pcons-net [50] (correlation and loss is 0.811 and 0.024, with p-value 1.93e-14 and 1.61e-6 respectively).

Table 5.1: The per-target average correlation, average loss, average spearman, and average kendall tau score of our method Qprob and other pure single-model QA methods on sel20 CASP11 dataset. The p-value of pairwise Wilcoxon signed ranked sum test for the difference of loss and correlation of Qprob against other methods is listed for comparison. Five single-model QA methods which don’t attend CASP11 are also listed and highlighted in bold.

Server name	Ave. corr.	Ave. loss	Ave. spearman	Ave. kendall.	p-value loss	p-value corr.	#
ProQ2	0.643	0.09	0.506	0.379	0.9776	0.2755	84
ProQ2-refine	0.653	0.093	0.535	0.402	0.9935	0.01756	84
Qprob	0.631	0.097	0.517	0.389	-	-	84
ModelEvaluator	0.6	0.097	0.47	0.353	0.9224	0.2678	84
VoroMQA	0.561	0.108	0.426	0.318	0.288	8.61E-05	84
Wang_SVM	0.655	0.109	0.535	0.401	0.09109	0.00313	84
Dope	0.542	0.111	0.416	0.316	0.06388	9.56E-10	84
Wang_deep_2	0.633	0.115	0.514	0.388	0.03468	0.2755	84
Wang_deep_3	0.626	0.117	0.513	0.388	0.00829	0.6034	84
Wang_deep_1	0.613	0.128	0.517	0.386	0.00056	0.403	84
DFIRE2	0.502	0.135	0.388	0.284	0.00059	1.08E-12	84
RWplus	0.536	0.135	0.433	0.323	0.00244	6.52E-11	84
FUSION	0.095	0.154	0.133	0.099	0.00157	4.05E-13	84
raghavagps-gaspro	0.35	0.156	0.263	0.187	0.00019	6.02E-12	84
RF_CB_SRS_OD	0.486	0.162	0.357	0.256	0.00011	4.56E-09	84

We also evaluate the performance of Qprob and other QA methods on Stage2 (top150) CASP11 datasets in **Table 5.2**. Qprob ranked second among all pure single-model QA methods based on the average loss metric. The difference between Qprob and ProQ2 is not significant on both correlation and loss (with p-value 0.2387 and 0.8636 respectively), showing close to state-of-the-art model selection ability among single-model QA methods. Comparing Qprob with five scores (ModelEvaluator score, Dope score, DFIRE2 score, RWplus score, and RF_CB_SRS_OD score), the difference of correlation between Qprob and four scores (ModelEvaluator score, Dope score, DFIRE2 score, and RWplus score) is significant, and the difference of loss between

Table 5.2: The per-target average correlation, average loss, average spearman, and average kendall tau score for our method Qprob and several other pure single-model QA methods on top150 CASP11 dataset. The p-value of pairwise Wilcoxon signed ranked sum test for the difference of loss and correlation of Qprob against other methods is listed for comparison. Five single-model QA methods which don't attend CASP11 are also listed and highlighted in bold.

Server name	Ave. corr.	Ave. loss	Ave. spearman	Ave. kendall.	p-value loss	p-value corr.	#
ProQ2	0.372	0.058	0.366	0.256	0.2387	0.8636	83
Qprob	0.381	0.068	0.387	0.272	-	-	83
VoroMQA	0.401	0.069	0.386	0.269	0.4335	0.5864	83
ProQ2-refine	0.37	0.069	0.375	0.264	0.2442	0.9656	83
ModelEvaluator	0.324	0.072	0.305	0.212	0.00255	0.3084	83
Dope	0.304	0.077	0.324	0.228	1.59E-07	0.74	83
RWplus	0.295	0.084	0.314	0.22	7.00E-09	0.11	83
Wang_SVM	0.362	0.085	0.351	0.245	0.4774	0.1502	83
raghavagps-qaspro	0.222	0.085	0.205	0.139	3.07E-07	0.00622	83
Wang_deep_2	0.307	0.086	0.298	0.208	0.00059	0.03628	83
Wang_deep_1	0.302	0.089	0.293	0.203	0.00091	0.04544	83
DFIRE2	0.235	0.091	0.253	0.175	6.15E-11	0.00404	83
Wang_deep_3	0.302	0.092	0.29	0.202	0.00047	0.00817	83
RF_CB_SRS_OD	0.36	0.097	0.35	0.243	0.06173	0.00204	83
FUSION	0.05	0.111	0.082	0.054	7.16E-11	5.82E-07	83

Qprob and two scores (DFIRE2 score, and RF_CB_SRS_OD score) is significant according to p-value threshold 0.01. In addition, we also compare the performance of Qprob with baseline consensus method DAVIS_consensus on Stage2 (top150) CASP11 datasets. The per-target average correlation of DAVIS_consensus is 0.57, which is better than Qprob (with correlation 0.381). The difference of correlation is significant (with p-value 6.14e-4). However, the per-target loss of Qprob (with loss 0.068) is better than DAVIS_consensus (with loss 0.073). Although the difference of loss is not very significant (p-value is 0.11), this shows similar model selection ability of Qprob on top150 CASP11 datasets comparing with consensus method. Moreover, comparing with top performing consensus QA method Pcons-net (with loss 0.049), the difference of loss between Qprob and Pcons-net is still not very significant (p-value is 0.19). To illustrate the model selection ability of the QA methods on hard target, we evaluate the performance of Qprob and several selected top performing single-model/consensus QA methods on the template free CASP11 targets, which are difficult for protein structure prediction. We calculate the summation of Z-score for the selected top 1 model by each QA method. The result is shown in **Figure 5.3**.

Figure 5.3(A) shows the performance of each method on Stage1 of CASP11 datasets. The single-model QA methods are in bold. We can see the consensus QA methods have relatively better performance, such as the baseline pairwise method DAVIS_consensus which gets the highest Z-score. **Figure 5.3(B)** shows the performance of each QA method on Stage 2 of CASP11 datasets. It is very interesting to see that the single-model QA methods have relatively better performance than consensus QA methods. Especially, our method Qprob and VoromQA have the highest Z-score comparing with other single QA methods. And another interesting finding is the pairwise method DAVIS_consensus has Z-score around 0, which is almost random. This shows the ability of single-model QA method for model selection on hard targets. The MULTICOM-CONSTRUCT ranks third based on the Z-score, which

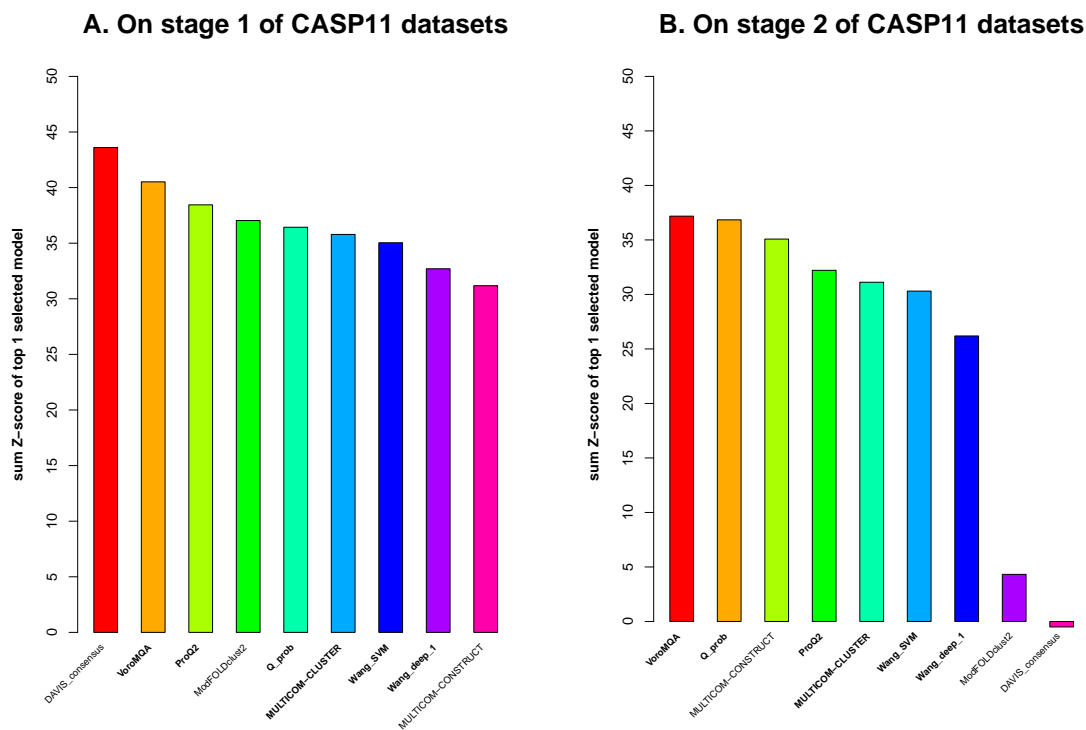


Figure 5.3: The summation of Z-score for the top 1 model selected by each method

combines single and consensus QA methods, showing the combination of single and consensus QA method is also quite useful for model selection.

5.5 Discussion

In this paper, we introduce a novel single-model QA method Qprob. Different from other single-model QA methods, we first time introduce the error estimation by benchmarking several different physicochemical, structural and energy feature scores, and use the combination of probability density distribution for the global quality assessment. We blindly tested our method on CASP11, and it is one of the best single-model QA method based on the CASP official evaluation and our evaluations. The good performance of our method on template free targets demonstrates the model selection ability of our method on hard targets. In addition, our method is also involved in the

model selection of MULTICOM human predictor attending CASP11, which is one of the best human predictors among all server and human predictors. This demonstrate the broad application of our method in model selection and protein structure prediction. In future, we plan to benchmark more features and improve the model selection ability of our method, finally apply it to predict more accurate protein structures.

Chapter 6

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

6.1 Abstract

Protein quality assessment (QA) by 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. 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 experiment 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. DeepQA is a useful 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/>.

6.2 Introduction

The tertiary structures of proteins are important for understanding their functions, and have a lot of biomedical applications, such as the drug discovery [100]. 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 [3, 9]. 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 [100].

There are generally two major challenges in protein structure prediction [12]. 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 [84, 41, 75, 101, 15, 102, 103] which uses the known structure information of homologous proteins as templates to build protein structure model, such as I-TASSER [104], FALCON [102, 103], MUFOLD [22] and RaptorX [105]. The second is *ab initio*

modeling method [37, 76, 106, 107, 108, 109], which builds the structure from scratch, without using existing template structure information. The second challenge is 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 [31, 81, 24], which calculate the average structural similarity score of a model against other models as its model quality. 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, which is a worldwide experiment for blindly testing protein structure prediction methods every two 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 methods is not working well when a large portion of models are of low quality [10]. 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 [10, 95, 96, 51, 82, 83, 84, 11, 108] 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 the model quality from sequence evolutionary information [97], residue environment compatibility [38], structural fea-

tures and physics-based knowledge [96, 51, 82, 83, 34, 39, 98]. On such single-model QA method - ProQ2 [53] 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.

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 [4, 110, 111] and bioinformatics [57]. We benchmark the performance of this method on large QA datasets, including the CASP datasets, four datasets from the recently 3DRobot decoys [52], 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.

6.3 Methods

6.3.1 Datasets

We collect three previous CASP models (CASP8, CASP9, and CASP10) from the CASP website http://predictioncenter.org/download_area/, 3DRobot decoys[52], and 3113 native protein structure from PISCES database [48] as the training datasets.

CASP11 models 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 generated in a Rosetta benchmark [15] each having 100 structural decoys; 20 proteins in a Modeller benchmark [112] each having 200 structural decoys; and 56 proteins in a I-TASSER benchmark [106] 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 1 contains 20 server models spanning the whole range of structural quality and each target at stage 2 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/>. In addition, we validate performance of our QA methods in a dataset produced by our *ab initio* modeling tool UniCon3D, which in total includes 24 targets and 20030 models.

6.3.2 Input features for DeepQA

In total, 16 features are used for benchmarking our method DeepQA, which describe the structural, physio-chemical and energy properties of a protein model. These features include 9 available top-performing energy and knowledge-based potentials scores, including ModelEvaluator score [82], Dope score [83], RWplus score [51], RF'CB'SRS'OD score [96], Qprob scores [11], GOAP score [54], OPUS score [55], ProQ2 score [53], DFIRE2 score [56]. All of these scores are converted into the range of 0 and 1 as the input features for training the deep learning 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 7 input features are generated from the physio-chemical properties

Table 6.1: 16 features for benchmarking DeepQA.

Feature Name	Feature descriptions
(1). Surface score (SU)	The total area of exposed nonpolar residues divided by 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[52] from the protein sequence and those of a model parsed by DSSP [53]
(5). RF_CB_SRS_OD score[26]	A novel distance dependent residue-level potential energy score.
(6). DFIRE2 score [47]	A distance-scaled all atom energy score.
(7). Dope score [29]	A new statistical potential discrete optimized protein energy score.
(8). GOAP score [45]	A generalized orientation-dependent, all-atom statistical potential score.
(9). OPUS score [46]	A knowledge-based potential score.
(10). ProQ2 score [36]	A single-model quality assessment method by machine learning techniques.
(11). RWplus score [27]	A new energy score using pairwise distance-dependent atomic statistical potential function and side-chain orientation-dependent energy term
(12). ModelEvaluator score [28]	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 [54] from a protein sequence and those of a model parsed by DSSP [53]
(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 [30]	A single-model quality assessment score that utilizes 11 structural and physicochemical features by feature-based probability density functions.

of a protein model. These features are calculated from a structural model and its protein sequence [98], 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.

A summary table of all features used for benchmarking DeepQA is given in **Table 6.1**.

6.3.3 Deep belief network architectures and training procedure

Our in-house deep belief network framework [57] is used to train deep learning models for protein model quality assessment. As is shown in **Figure 6.1**, 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 [58]. 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 [113]. 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 (default 0.001), weight cost (default 0.07), and momentum (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.

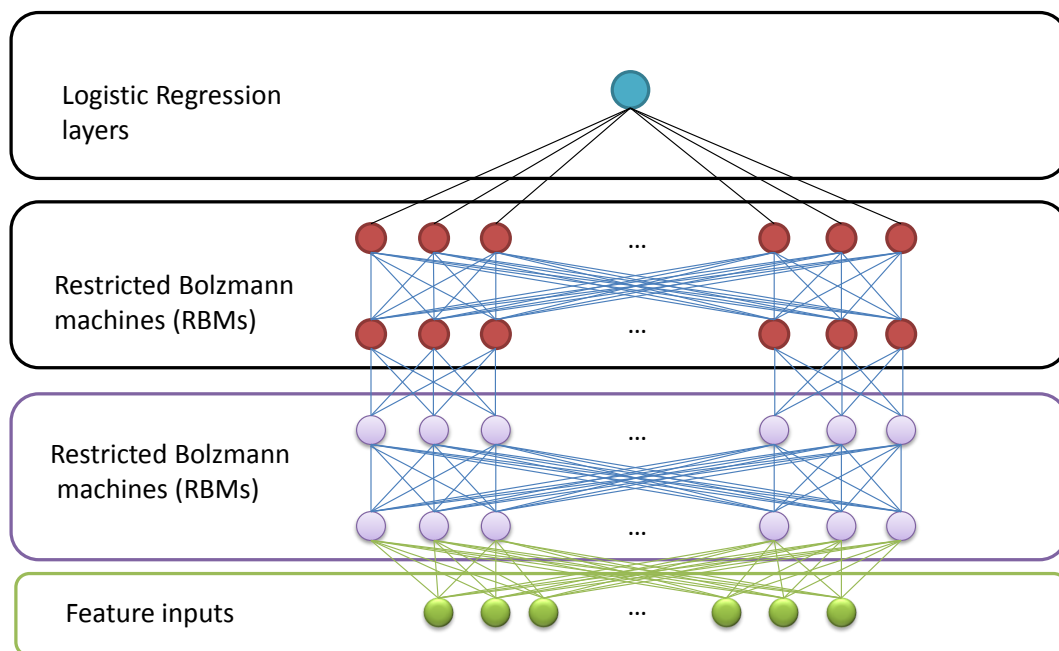


Figure 6.1: The Deep Belief Network architecture for DeepQA.

6.3.4 Model accuracy evaluation metrics

We evaluate the accuracy of DeepQA on 84 protein targets on both stage 1 and stage 2 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 [92]. 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 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 1 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 5 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 best model, and also the selected best of top 5 models by QA methods.

6.4 Results and Discussion

6.4.1 Comparison of Deep learning with support vector machines and neural networks

We train the deep learning and two other most 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 [101] is used to train the support vector machine, and the tool Weka [114] is used to train the neural networks. The RBF 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 7500 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, α to 0.19, γ 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 from 0.0001 to 0.01, weight cost from 0.001 to 0.7, and momentum 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.

The correlation and loss on both stage 1 and stage 2 models of CASP11 datasets are calculated for these three methods, and the results are shown in **Table 6.2**. Deep belief network has the best average per-target correlation on both stage 1 and stage 2. The loss of DeepQA is also lower than or equal to the other two methods. The results suggest that deep belief network is a good choice for protein quality assessment problem.

Table 6.2: The accuracy of Deep Belief Network, Support Vector Machines, and Neural Networks in terms of MAE based on cross validation of training datasets, the average per-target correlation, and loss on stage 1 and stage 2 of CASP11 datasets for all three difference techniques.

	MAE based on cross validation	Corr. on stage 1	Loss on stage 1	Corr. on stage 2	Loss on stage 2
Deep Belief Net- work	0.08	0.63	0.09	0.34	0.06
Support Vector Machine	0.12	0.58	0.1	0.32	0.07
Neural Network	0.08	0.51	0.12	0.25	0.07
Mean	0.09	0.57	0.1	0.3	0.07

6.4.2 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, Modevaluator score, Secondary structure penalty score, Euclidean compact score, and Qprob score.

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 6.3**). On stage 1 of CASP11, the average per-target correlation of DeepQA is 0.64, which is the same as the ProQ2 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 2 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. 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, we apply DeepQA to select models in the stage 2 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.

Table 6.3: 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 2 of CASP11.

QA methods	Corr. on stage 1	Loss on stage 1	Corr. on stage 2	Loss on stage 2
DeepQA	0.64	0.09	0.42	0.06
ProQ2	0.64	0.09	0.37	0.06
Qprob	0.63	0.1	0.38	0.07
VoroMQA	0.56	0.11	0.4	0.07
ProQ2-refine	0.65	0.09	0.37	0.07
Wang_SVM	0.66	0.11	0.36	0.09
raghavagps-qaspro	0.35	0.16	0.22	0.09
Wang_deep_2	0.63	0.12	0.31	0.09
Wang_deep_1	0.61	0.13	0.3	0.09
Wang_deep_3	0.63	0.12	0.3	0.09
FUSION	0.1	0.15	0.05	0.11
Mean	0.55	0.12	0.32	0.08

6.4.3 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 6.4**) 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.

Table 6.4: Model selection ability on *ab initio* datasets for DeepQA, ProQ2, Dope2, and RWplus score

QA methods	TM-score on top 1 model	RMSD on top 1 model	TM-score on best of top 5	RMSD on best of top 5
DeepQA	0.23	19.01	0.26	17.14
ProQ2	0.22	19.73	0.25	17.93
Dope	0.22	19.55	0.24	18.1
RWplus	0.22	19.68	0.25	17.38
Mean	0.22	19.49	0.25	17.64

6.5 Conclusions

In this paper, we develop a 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.

Chapter 7

Large-Scale Model Quality Assessment for Improving Protein Tertiary Structure Prediction

7.1 Abstract

Sampling structural models and ranking them are the two major challenges of protein structure prediction. Traditional protein structure prediction methods generally use one or a few quality assessment methods to select the best-predicted models, which cannot consistently select relatively better models and rank a large number of models well. Here, we develop a novel large-scale model quality assessment method in conjunction with model clustering to rank and select protein structural models. It unprecedentedly applied 14 model quality assessment methods to generate consensus model rankings, followed by model refinement based on model combination (i.e., averaging). Our experiment demonstrates that the large-scale model quality assessment approach is more consistent and robust in selecting models of better quality than any individual quality assessment method. Our method was blindly tested during the 11th Critical Assessment of Techniques for Protein Structure Prediction

(CASP11) as MULTICOM group. It was officially ranked 3rd out of all 143 human and server predictors according to the total scores of the first models predicted for 78 CASP11 protein domains and 2nd according to the total scores of the best of the five models predicted for these domains. MULTICOM's outstanding performance in the extremely competitive 2014 CASP11 experiment proves that our large-scale quality assessment approach together with model clustering is a promising solution to one of the two major problems in protein structure modeling. The web server is available at: <http://sysbio.rnet.missouri.edu/multicom/cluster/human/>.

7.2 Introduction

Protein tertiary structure prediction has been an important scientific problem for few decades, especially in bioinformatics and computational biology [105]. Despite more and more native structures are included in Protein Data Bank (PDB) [114] database, the gap between the sequenced proteins and the native structures is still enlarging due to the exponential increase of protein sequences produced by large-scale genome and transcriptome sequencing. It is estimated that less than 1% of protein sequences have the native structures in PDB database [115]. Therefore, accurate computational methods for protein tertiary structure prediction that are much cheaper and faster than experimental structure determination techniques are needed to reduce this large sequence-structure gap. Furthermore, computational structure prediction methods are important for obtaining the structures of membrane proteins whose structures are hard to be determined by experimental techniques such as X-ray crystallography [100].

The two major problems of protein structure prediction are model sampling and model ranking. The former is to generate a number of structural models (conformations) for a protein target, and the latter is to rank these models and to select

the presumably best ones as final predictions. The two main ways of generating protein models are template-based modeling and template-free modeling. Template-based modeling methods use the known structures (templates) of the proteins that are homologous or analogous to a target protein to construct structural models for the target [102, 107, 111]. For instance, during 2014 CASP11 experiment, almost all the structure prediction servers such as I-TASSER [84, 110], MULTICOM [52, 107], MUFOLD [93], and RaptorX [15] used the template-based model technique to predict structures of some CASP11 targets for which some homologous template structures could be found. Template-free modeling methods predict the protein tertiary structure from scratch without using template information. This is especially important when there are no structural homologs existing in the database or the template identification techniques cannot find good templates [107]. Some CASP11 prediction servers such as ROSETTA [76], QUARK [116], and FALCON [11] used template-free modeling method to generate structural models for some hard CASP11 targets.

Once some structural models are generated for a protein, the remaining challenge is to assess the quality of these models and select the most accurately predicted models. There are generally two main kinds of quality assessment (QA) methods: single-model quality assessment methods [51, 83, 34, 82, 108, 117], which evaluate the quality of one single model without using the information of other models; and multi-model quality assessment methods [50, 24, 32, 90], which use the structural similarity between one model and other models of the same protein to assess its quality. The multi-model quality prediction methods generally perform better than the single-model quality prediction methods given the pool of models is sampled by independent structure predictors. However, multi-model quality assessment method is largely influenced by the proportion of good models in the pool or the average quality of the largest model cluster in the pool, whereas single-model quality assessment methods may work better in assessing a small number of models of wide-range quality

usually associated with a hard target or a pool of models with very low proportion of good ones [90].

Currently, most protein structure prediction methods use one or at most a few quality assessment methods to rank and select models, generally leading to the poor performance in selecting models of good quality due to the extreme difficulty of ranking models and intrinsic limitations of individual quality assessment methods. Some structure prediction methods also apply clustering techniques to group models into different clusters whose center is considered as the best model in each cluster based on the structural similarities. The hypothesis behind it is that near-native structures are more likely clustered in a large free-energy basin in the free-energy landscape [103, 104]. The clustering based approaches generally select an average model rather than the best model and cannot work well if the quality of the largest cluster is not good. Therefore, although numerous methods have been developed to assess, rank, and select models, protein model ranking is still largely an unsolved problem.

In order to address this challenge, we developed a large-scale consensus quality assessment method (MULTICOM) to combine 14 complementary model quality assessment methods to improve the reliability and robustness of protein model ranking. The general model ranking is also synergistically integrated with model clustering techniques to increase the diversity and quality of the final selected models. On the very competitive 2014 CASP11 benchmark, this new method substantially outperforms any single quality assessment method, suggesting its unique value in addressing one major problem of protein structure prediction

7.3 Methods

7.3.1 Large-scale protein model quality assessment for protein tertiary structure prediction

Table 7.1: All 14 QA methods with the details. The highlighted methods are built in house. S: single-model method; M: multi-model method.

Methods	Type	Features
MULTICOM-NOVEL	Single	Structural, physical, chemical features
OPUS-PSP	S	Contact potentials based on side chain functional groups
ProQ2	S	Structural features
RWplus	S	Side-chain orientation dependent potential
ModelEvaluator	S	Structural features, contacts
Modelcheck2	S	Structural features, contacts, disorder, conservation
RF_CB_SRS	S	Distance dependent statistical potential
SELECTpro	S	Energy-based (h-bond, angle, electrostatics, vdw)
Dope	S	Statistical potential
DFIRE2	S	Energy-based potential
ModFOLDclust2	Multi	Pairwise model similarity (geometry)
APOLLO	M	Pairwise model similarity
Pcons	M	Pairwise model similarity
QApro	M+S	Weighted pairwise model similarity
MULTICOM (human)	Consensus	Average ranking

Given a pool of structural models generated for a target protein (e.g. hundreds of models generated for a CASP11 target), the MULTICOM method used unprecedentedly 14 complementary model quality assessment (QA) methods to predict the quality score of each model first (**Table 7.1**). These QA methods include both single-model and multi-model QA methods. The single-model methods include our new single-model global quality assessment method MULTICOM-NOVEL based on the difference between secondary structure and solvent accessibility predicted by Spine X

[99] and SSpro4 [90] from the protein sequence and those of a model parsed by DSSP [91], physical-chemical features (i.e., surface polar score, weighted exposed score, and etc.) [98], the normalized quality score generated by ModelEvaluator [82], RWplus score [51], dope score [83], and RF_CB_SRS_OD score [96]; ProQ2 [34]; model check2 method produced by an improved version of ModelEvaluator [82]; a recalibrated SELECTpro energy [117]; Dope [83]; DFIRE2 [19]; OPUS`PSP [118]; Rplusplus [51]; ModelEvaluator [82] and RF_CB_SRS_OD [96]. The multi-model QA methods include ModFOLDclust2 [24]; Pcons [50]; APPOLLO [32]; Q Apro - a weighted combination of ModelEvaluator and APOLLO [90]. The details of each method are described in **Table 7.1**.

During the 2014 CASP11 experiment, MULTICOM used two different combinations of the QA scores produced by 14 QA methods to generate consensus rankings to rank all models of each target. The first one is the complete combination, in which each of 14 QA methods was applied to all the models of a target and generated a ranking for them based on their QA scores, and the average rank of 14 ranks of each model assigned by the 14 QA methods was used as its final rank. The second one is the consensus rankings based on the same average ranks produced by only six QA methods including (MULTICOM-NOVEL QA score, Q Apro score, Pcons score, Modelcheck2 score, Dope score, OPUS`PSP score). These six methods were selected because their combination performed best on all the models of 46 CASP10 when all possible combinations were benchmarked before CASP11 experiment started. On these CASP10 models, the average loss score of top one model based on 6 QA methods is 0.037, lower than 0.057 of all 14 QA methods. However, considering that the optimization process in benchmarking could over fit the data, we let MULTICOM use the consensus rankings of both the 6 selected QA methods and all 14 QA methods.

During the modeling ranking process, if the same top one model was selected by the two consensus rankings, which happened in more than 50% cases, the consensus

ranking of the 6 QA methods were used as the final ranking of all the models. But if they disagreed with each other, the score of top 1 model selected by the pairwise QA method APOLLO was used to break the tie as follows. On one hand, if the score of APOLLO's top one model was ≥ 0.3 , which generally meant quite some models in the model pool were of good quality due to relatively high pairwise similarity between them, the final ranking was set as the consensus ranking of the 6 QA methods or all 14 QA methods depending on whose top one model was more similar to the top one model of APOLLO than the other. Furthermore, the top one models of the two consensus rankings and of the top predictors (e.g., MULTICOM-CLUSTER and Zhang-Server) were compared with the top one model of APOLLO, and the model most similar to the top one model of APOLLO was used the top one model in the final ranking without changing the ranking of all other models. On the other hand, if the score of the top one model selected by APOLLO was < 0.3 , which only occasionally happened and suggested that the target was hard and most models were of bad quality, MULTICOM calculated the percent of matching between the secondary structures extracted from the top one model selected by either 6- or 14-QA consensus ranking with the secondary structure of the target predicted from its sequence. The final ranking was one of 6 or 14 consensus ranking whose top one model had the higher percentage of matching of secondary structures.

Since the top five models selected by the final ranking above sometime could be very similar to each other, the risk for all of them to fail altogether was high for hard targets. To reduce this risk, MULTICOM only kept the top two models of the final ranking as the two predicted structures. And then, in order to increase the diversity of top five models selected as final predictions for each target, MULTICOM used MUFOLD-CL [4] to cluster models, and then selected the other three models according to the final ranking in separate clusters different from those of the top two models. MUFOLD-CL [4] is a model clustering method based on the comparison of

the protein distance matrices. Comparing with other clustering techniques based on structural distance such as RMSD [22], it is much faster, but yields similar accuracy, which is desirable for clustering a large number of protein models. During the selection of the other three models from different clusters, MULTICOM also skipped the models ranked at bottom 10% according to our newly-developed MULTICOM-NOVEL QA method. This guaranteed that the top five selected structures were largely different, which indeed improved the score of the best of top five models.

Finally, MULTICOM used a model combination approach [101] to integrate each selected model with other similar models in the model pool to generate its refined model. The workflow of our MULTICOM method described above is illustrated in **Figure 7.1**.

7.3.2 Evaluation of top ranked models

We downloaded publically available native structures for 42 CASP11 human targets from the CASP’s website (<http://www.predictioncenter.org/casp11/index.cgi>). During CASP11, our MULTICOM method was blindly benchmarked on these targets together with 142 human and server predictors. The predicted structural models were assessed on 55 domains of the 42 targets. For comparison, we downloaded both the other predictors’ predictions and our submitted predictions from the CASP11’s website. During CASP11, each predictor submitted up to five predicted model with the first one (TS1) designated as the best model. We evaluated the performance of each predictor’s first model by calculating the GDT-TS score between it and its native structure. The TM-score [92] was used for calculating the GDT-TS score. The Z-score of a model was calculated as the model’s GDT-TS score minus the average GDT-TS score of all the models in the model pool of a target divided by the standard deviation of all GDT-TS scores. The negative Z score was converted to 0 during summation of Z-scores. The sum of the Z-scores of the first models predicted by a

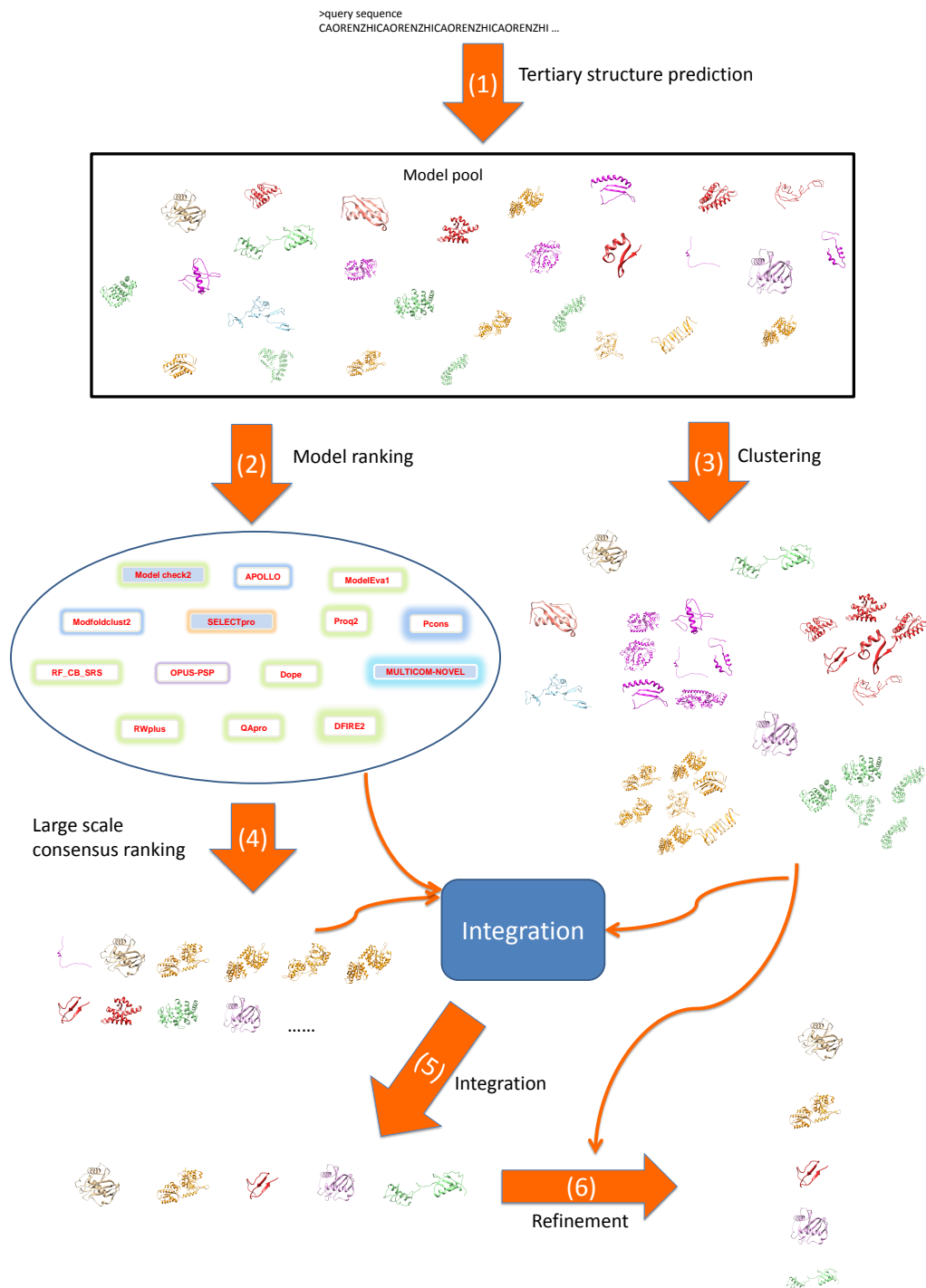


Figure 7.1: The workflow of the MULTICOM method comprised of six steps.

predictor for the 42 targets was used to measure its overall performance. Similarly, the sum of the Z-scores of the best of the five submitted models predicted by a predictor for the 42 targets was used to measure its performance if the best of all five submitted models was considered.

7.4 Results and Discussions

We evaluated the performance of MULTICOM human predictor along with 44 CASP-11 server predictors on 42 CASP11 human targets. The sum of Z-scores of all first (i.e. TS1) models or the best of five submitted models predicted by these predictors was reported in **Table 7.2**. Other human server predictions were not considered in the analysis here since they were not publicly available. It is shown that MULTICOM performs better than all server predictors. Its total Z-score of first models is around 4 points higher than the best server predictor Zhang-Server, and its total Z-score of the best of five models is 6 points higher than the best server predictor QUARK. These results demonstrate MULTICOM’s ability to rank a large pool of models for selecting top one or five models. According to CASP11’s official evaluation of all 143 human and server predictors, MULTICOM was ranked 3rd based on the sum of Z-score of the first model and 2nd based on the sum of Z-score of best of the five submitted models. The MULTICOM’s outstanding performance in the extremely competitive CASP11 experiment demonstrates that our large-scale model quality assessment is powerful for ranking and selecting good models from a pool of models of different quality.

In order to investigate types of the models selected by MULTICOM and the contribution of individual structure predictors, we calculated the number of times that the models predicted by each predictor were ranked within top five by MULTICOM. **Table 7.3** shows the contribution of top 10 server predictors whose models were selected by MULTICOM to refine to generate the final predictions. It shows that a

Table 7.2: The top 10 tertiary structure predictors ranked based on the summation of the Z-scores of the first models, and their summation of the Z-scores of best of the five submitted models.

Server name	Sum of Z/rank	Sum of Z of best of five/rank
MULTICOM (human)	57.49/1	78.42/1
Zhang-Server	53.62/2	70.57/3
QUARK	51.90/3	71.93/2
Nns	35.07/4	51.79/6
myprotein-me	34.11/5	52.73/5
MULTICOM-CLUSTER	31.39/6	39.03/10
MULTICOM-CONSTRUCT	31.33/7	38.65/11
RBO_Aleph	30.77/8	40.65/9
BAKER-ROSETTASERVER	28.80/9	63.64/4
MULTICOM-NOVEL	25.71/10	43.43/7

Table 7.3: The top 10 predictors ranked based on the total number times their models were selected by our MULTICOM predictor on all the human targets or template-based (TBM) human targets only.

Rank	Servers on all human targets	Num. on all	Servers on TBM	Num. on TBM
1	Zhang-Server	58	Zhang-Server	43
2	BAKER-ROSETTASERVER	36	BAKER-ROSETTASERVER	27
3	QUARK	29	QUARK	22
4	RBO_Aleph	29	myprotein-me	20
5	myprotein-me	28	Nns	19
6	Nns	21	Seok-server	14
7	Seok-server	17	RBO_Aleph	13
8	MULTICOM-REFINE	10	MULTICOM-REFINE	8
9	FUSION	7	RaptorX	4
10	RaptorX	5	FUSION	4

Table 7.4: Comparison of MULTICOM with each QA method and the two different consensus methods (one based on 6 QA methods and another one based on 14 QA methods) on the average GDT-TS score and Z-score of the top models selected, and the significance of difference between each QA method and MULTICOM. Italic font denotes single-model methods.

QA method	Ave. GDT-TS score on all	Ave. GDT-TS score on TBM	Ave. Z score on all	Z p-value of Z score diff.	Ave. Z score removed
MULTICOM	0.374	0.425	1.364	-	-
Consensus of 14 QA scores	0.369	0.42	1.217	-	-
Consensus of 14 Z-scores	0.357	0.402	1.406	-	-
<i>SELECTpro</i>	0.351	0.407	0.893	1.83E-05	1.338
<i>ProQ2</i>	0.343	0.387	0.887	1.19E-02	1.365
<i>MULTICOM-NOVEL</i>	0.34	0.383	0.861	5.61E-03	1.321
ModFOLDclust2	0.339	0.399	0.734	2.07E-04	1.356
APOLLO	0.338	0.403	0.584	9.33E-05	1.379
<i>Dope</i>	0.334	0.382	0.819	1.86E-03	1.36
Pcons	0.333	0.397	0.565	1.83E-05	1.325
<i>ModelEva</i>	0.333	0.378	0.87	9.84E-03	1.334
<i>Dfire2</i>	0.329	0.367	0.826	1.66E-03	1.36
QApro	0.328	0.371	0.783	2.89E-02	1.43
RWplus	0.327	0.373	0.752	5.19E-04	1.365
<i>OPUS-PSP</i>	0.326	0.366	0.793	5.78E-03	1.356
<i>RF_CB_SRS</i>	0.3	0.343	0.372	7.13E-05	1.365
<i>Modelcheck2</i>	0.297	0.347	0.559	1.19E-02	1.34

diverse set of server predictors including Zhang-Server made significant contributions to the final prediction, suggesting the large-scale quality assessment used by MULTICOM can reliably assess a very diverse set of models generated by different tertiary structure predictors in the field.

To study how our large-scale model quality assessment method improves model ranking, we compared its performance with that of each individual QA method and the two other simple consensus methods (one based on the sum of 14 original QA scores and another based on the sum of 14 Z-scores calculated from original scores). The first two columns in **Table 7.4** reports the average GDT-TS score of the first models selected by these QA methods for all 42 human targets and a subset of 30 template based human targets, respectively. The results show that MULTICOM performs better than every individual QA method, and sometime the improvement is substantial. And not surprisingly, the multi-model quality assessment methods outperformed single-model quality assessment methods on template based human targets whose model pool was often of good quality. For instance, a multi-model QA method APOLLO ranks 6th on all human targets, but 3rd on template based human targets. The third columns in **Table 7.4** shows the average Z-score of the first models selected by different QA methods. It is interesting to notice that the single-model QA methods tend to have higher Z-score than the multiple-model QA methods. For example, the multiple QA method APOLLO has a relatively high average GDT-TS score (0.338) of the first selected models, however, its average Z-score of the first selected models is lower than most single QA methods. The reason is probably because the multiple-model QA methods tend to work well on easy targets whose models have similarly good quality and thus low Z-scores, whereas single-model QA methods may select some good models for some hard targets whose models are mostly bad, resulting in a high Z-score.

Considering average ranking is just one way of combining different QA scores,

we tested another two ways to combine QA scores for comparison. The first one simply calculated the average of original 14 QA scores to rank models. The second one first converted all original QA scores of each method into Z-scores, and then used the average of 14 Z-scores to rank models. **Table 7.4** shows that consensus of 14 QA Z-scores performed best in terms of the average Z-score of the top one models, whereas MULTICOM performed best in terms of the average GDT-TS score of the top one models. The results demonstrate that the way of integrating different QA scores influences the quality of the final ranking.

Moreover, we compared MULTICOM with a simple combination approach that used a good single-model QA method (i.e. ProQ2) to rank models of very hard targets and a good clustering method (APOLLO) to rank the models of other targets. If the maximum APOLLO pairwise score of the models of a target is < 0.2 , it is considered a hard target, otherwise an easy target. The average Z-score and GDT score of the top 1 model selected by this simple combination method is 0.980 and 0.350, respectively, which is higher than that (0.584 and 0.338) of APOLLO, but substantially lower than that (1.364 and 0.374) of MULTICOM.

Furthermore, compared to the two other top-ranked consensus methods participating in CASP11 experiment - TASSER (ranked 9th in CASP11) and keasar (ranked 27th) that used several QA methods according to the official CASP11 experiment, MULTICOM was rank 3rd, demonstrating its effectiveness and robustness.

We also used Wilcoxon signed ranked sum test to assess the significance of the difference between MULTICOM and each individual QA method. The fifth column of **Table 7.4** shows p-value of the top one model's Z-score difference between MULTICOM and each QA method. According to 0.05 threshold, MULTICOM performed significantly better than any individual QA method.

In addition, in order to test the impact of each single-model QA method on the performance of the consensus approach, we tested how removing each QA method

may change the average Z-score of top 1 model selected by the consensus ranking of the remaining 13 QA methods. The results were in Column 6 in **Table 7.4**. According to the results, the removal of MULTICOM-NOVEL caused the biggest decrease in the average Z-score of top one models selected by the consensus method.

Moreover, we counted the total number of times one QA method selected better models than all other QA methods. In the cases where more than one QA method selected the same better model, all of them were counted as better than others methods once. **Table 7.5** shows that MULTICOM consistently selected better top models more frequently than any other QA method. Interestingly, SELECTpro only selected better model once (**Table 7.5**), yet it had the higher average GDT-TS scores for all the top one models than the other 13 individual QA methods (**Table 7.4**), suggesting that SELECTpro selected top models with relatively higher GDT-TS score for most targets, but not necessarily the best models compared with other individual QA methods.

In addition to assessing the overall performance, we specifically investigated two examples to illustrate how MULTICOM assessed the quality of the models of the following two targets. The first case is T0783-D2 (domain 2 of Target T0783). **Figure 7.2(A)** illustrates the distribution of the GDT-TS scores of the models of this domain, where most of the models actually have the true GDT-TS score less than 0.2 (i.e. very low quality), some models have the GDT-TS score around 0.4 (medium quality), and a few models have GDT-TS score 0.6 (relatively good quality). **Figure 7.2(B)** is the plot of true GDT-TS scores of these models against their ranking predicted by MULTICOM. It is shown that MULTICOM ranked the best model with the highest GDT-TS score (e.g. nns_TS1) as no. 1. In this case, all the individual single QA methods ranked this model within top 5, but a pairwise method ranked it at no. 19. Combining these individual rankings, the consensus ranking predicted by MULTICOM was able to select this model to combine with other three similar models

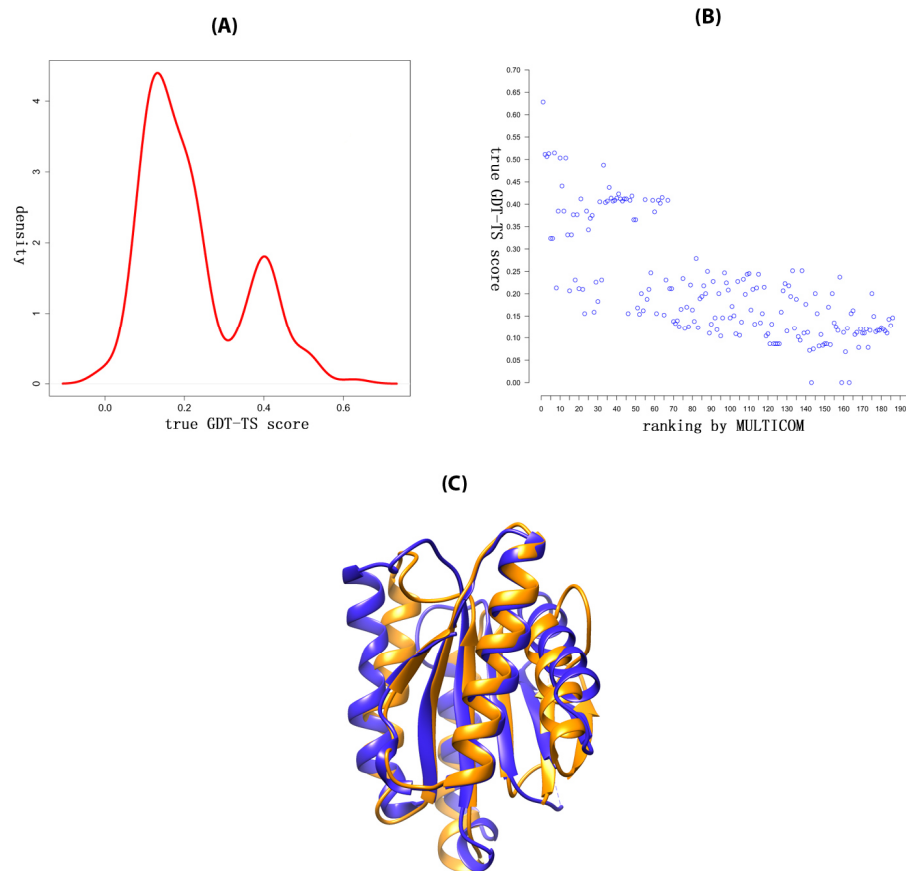


Figure 7.2: Tertiary structure prediction of domain 2 of T0783 (T0783-D2). (A) The superposition of the MULTICOM human TS1 model on domain 2 with the native structure. (B). The distribution of 191 models in the model pool. (C). The plot of the true GDT-TS scores of models against their predicted ranking.

Table 7.5: The total number times that each QA method performed better than other QA methods on all human targets or all template based (TBM) human targets only. Italic denotes single-model methods.

QA methods	Frequency on all targets	QA methods	Frequency on TBM
MULTICOM	17	MULTICOM	11
QApro	12	QApro	8
<i>ProQ2</i>	11	<i>ModelEva</i>	7
<i>ModelEva</i>	9	<i>ProQ2</i>	7
<i>Dfire2</i>	9	<i>Dope</i>	7
<i>Dope</i>	9	<i>RWplus</i>	6
<i>RWplus</i>	8	<i>Dfire2</i>	6
<i>MULTICOM-NOVEL</i>	8	<i>MULTICOM-NOVEL</i>	6
<i>OPUS-PSP</i>	8	<i>OPUS-PSP</i>	6
<i>Modelcheck2</i>	4	<i>APOLLO</i>	4
<i>RF_CB_SRS</i>	4	<i>Modelcheck2</i>	3
APOLLO	4	<i>RF_CB_SRS</i>	3
ModFOLDclust2	3	ModFOLDclust2	3
Pcons	2	Pcons	2
<i>SELECTpro</i>	1	<i>SELECTpro</i>	1

(nns_TS3, nns_TS2, and FFAS-3D_TS1) to generate a refined model as final prediction. **Figure 7.2(C)** is the superposition of this model with the native structure, which is an alpha-best-alpha protein. Our final model has a well-predicted four-strand beta-sheet in the middle and two well-positioned alpha helices in periphery. The final GDT-TS score of this model is 0.625.

The second case is T0767-D1 (domain 1 of Target T0767). **Figure 7.3(A)** shows the distribution of the true GDT-TS score for the whole model pool. Most models are of low quality (i.e. the true GDT-TS score around 0.25), which makes model quality assessment difficult. Therefore, three pairwise QA methods (APOLLO, Pcons, and ModFOLDclust2) failed to rank the models of good quality at or near the top, whereas some single-model quality assessment methods ranked them higher. **Figure 7.3(B)** is the plot of the true GDT-TS scores of these models against their ranking predicted by MULTICOM. It is shown that our large-scale model quality assessment combining

both single- and multi-model QA methods was able to rank the third best model at the top, even though it missed the best model BAKER-ROSETTASERVER_TS2 in the model pool. The initial model selected by MULTICOM was Zhang-Server_TS5 with GDT-TS score 0.5658. **Figure 7.3(C)** visualizes the superposition of the predicted model and the native structure. It is shown that the beta sheet was predicted rather accurately, whereas the alpha helices were only partly correctly predicted.

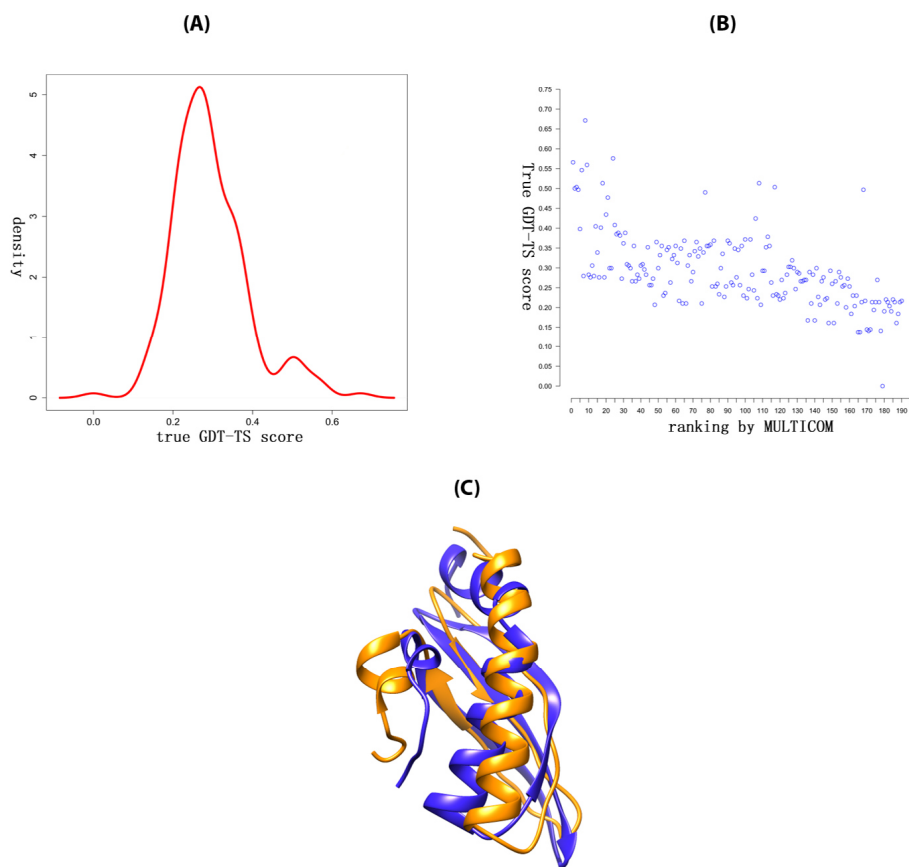


Figure 7.3: Tertiary structure prediction of domain 1 of T0767 (T0767-D1). (A) The superposition of the MULTICOM human TS1 model on domain 1 with the native structure. (B). The distribution of 195 models in the model pool. (C). The plot of the true GDT-TS scores of models against their predicted ranking.

Finally, we investigated if the model combination could refine and improve the quality of the selected models. **Figure 7.4** shows the difference between the initial GDT-TS scores of the models before refinement and the GDT-TS scores of the final

models after the refinement process on 42 CASP11 human targets. The GDT-TS scores of the models of 19 targets were increased by the model combination, those of another 19 targets were decreased, and those of the remaining 4 targets stayed the same. The average change of GDT-TS scores of all 42 targets was 0, suggesting the refinement process did not improve the global quality of the models on average, which is consistent with the observation on the performance of most current model refinement protocols [119].

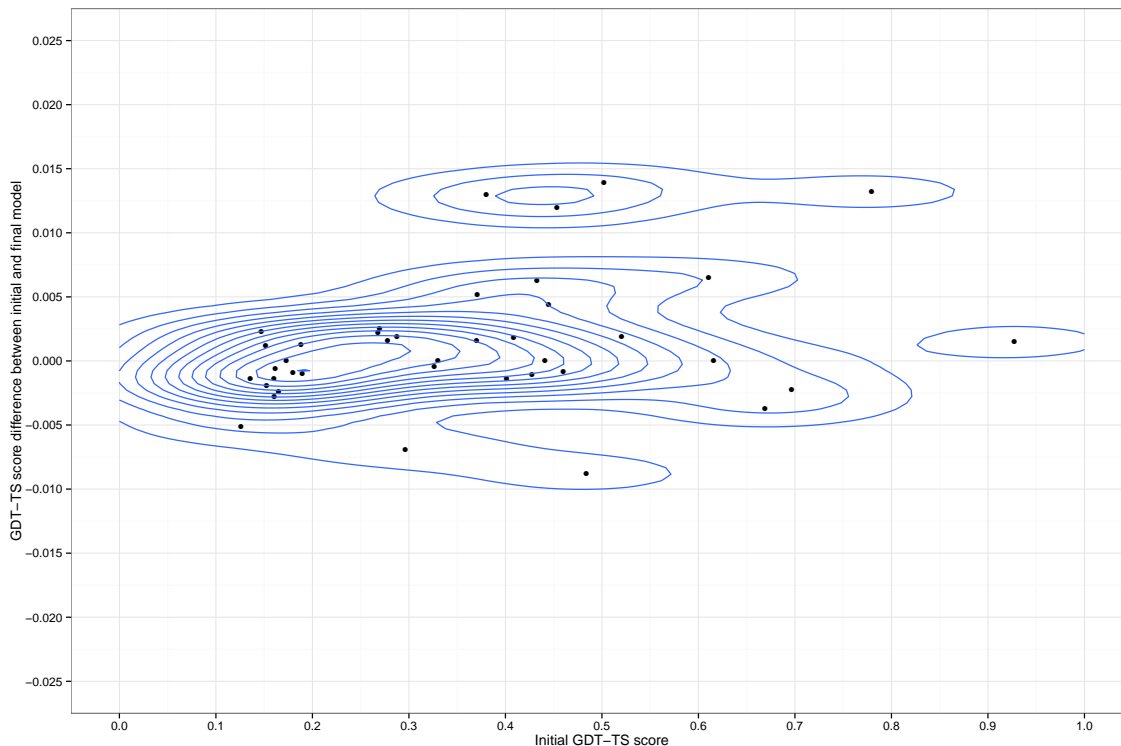


Figure 7.4: The plot of the difference between the initial GDT-TS scores before model combination and the GDT-TS scores after model combination against the initial GDT-TS scores of top one models of 42 targets

7.5 Conclusions

We developed a large-scale model quality assessment technique in conjunction with model clustering and refinement to improve protein tertiary structure prediction. In-

spired by the previous work [120] that integrated several primary QA methods, our method that combined a large number of protein model quality assessment methods reliably and consistently improved protein model ranking – one of the major challenges of protein structure prediction. For the first time, we demonstrate that this large-scale consensus QA approach is more robust and accurate than any individual quality method by integrating their strength together. Our tertiary structure prediction based on this method outperformed all the server predictors during the very competitive CASP11 experiment in 2014. The CASP11 official assessment also ranked our method as one of the top three best tertiary structure prediction methods on all the CASP11 human targets. This outstanding performance demonstrates our large-scale model quality assessment approach is a promising direction to advance the state of the art of protein model ranking and selection. Moreover, our approach adopts an open quality assessment system, into which, adding more complimentary methods may potentially improve the ranking, but incorporating redundant methods does not necessarily lead to an improvement. However, our general combination approach demonstrates the importance of developing more individual QA methods and the possibility of optimally combining them together to advance the field of protein structure prediction.

Chapter 8

Massive integration of diverse protein quality assessment methods to improve template based modeling in CASP11

8.1 Abstract

Model evaluation and selection is an important step and a big challenge in template-based protein structure prediction. Individual model quality assessment methods designed for recognizing some specific properties of protein structures often fail to consistently select good models from a model pool because of their limitations. Therefore, combining multiple complimentary quality assessment methods is useful for improving model ranking and consequently tertiary structure prediction. Here, we report the performance and analysis of our human tertiary structure predictor (MULTI-COM) based on the massive integration of 14 diverse complementary quality assessment methods that was successfully benchmarked in the 11th Critical Assessment of Techniques of Protein Structure prediction (CASP11). The predictions of MULTI-

COM for 39 template-based domains were rigorously assessed by six scoring metrics covering global topology of $C\alpha$ trace, local all-atom fitness, side chain quality, and physical reasonableness of the model. The results show that the massive integration of complementary, diverse single-model and multi-model quality assessment methods can effectively leverage the strength of single-model methods in distinguishing quality variation among similar good models and the advantage of multi-model quality assessment methods of identifying reasonable average-quality models. The overall excellent performance of the MULTICOM predictor demonstrates that integrating a large number of model quality assessment methods in conjunction with model clustering is a useful approach to improve the accuracy, diversity, and consequently robustness of template-based protein structure prediction.

8.2 Introduction

In the genomic era, high-throughput genome or transcriptome sequencing technologies have generated a large amount (~ 100 million) of protein sequences. It is important to obtain the tertiary structures of these protein sequences in order to understand their biochemical, biological and cellular functions [72, 43, 121]. Experimental techniques (e.g. X-ray crystallography or NMR spectroscopy) can determine protein structures. However, these techniques cannot solve the structures of all proteins because they are relatively expensive and time consuming. Thus far, only a small portion of proteins ($\sim 99,000$) have experimentally verified structures. Therefore, cheaper and faster computer-assisted prediction of protein tertiary structures is becoming increasingly popular and important [122, 123, 124, 125, 126].

Computational prediction methods of protein tertiary structures generally fall into two categories: template-based modeling and template-free modeling. Template-based modeling methods generate the tertiary structure for a target protein by iden-

tifying its homologous structure templates and transferring the template structures to the structure of the target for further refinement [78, 127, 128]. These methods are the most widely used protein modeling methods, and their predictions are relatively accurate and usable if good homologous templates could be found. If no homologous templates could be found for a target protein, template-free modeling methods are employed to construct structural models for the target protein from scratch or from the combination of small structural fragments [127, 15]. Since 1994, every two years both template-based and template free modeling methods (e.g. [104, 129, 22, 15, 116, 103]) were blindly and rigorously evaluated in the Critical Assessment of Protein Structure Prediction (CASP) experiments. In this work, we report our findings and analyses regarding the template-based predictions of our MULTICOM predictor based on massive integration of diverse and complementary protein model quality assessment methods in the CASP11 experiment held in 2014.

Evaluating the quality of predicted models and selecting the most accurate ones from them is an important step and a big challenge in protein structure prediction. There are two typical kinds of protein model quality assessment (QA) methods: single-model quality assessment method and multi-model quality assessment method [10]. Single-model quality assessment methods [10, 95, 34, 85, 87, 92, 103, 116, 15, 130, 131] evaluate the quality of a single model without referring to other models and assigned it a global quality score. Multi-model quality assessment methods [31, 80, 24, 32, 132, 133, 134] (also called clustering based methods) evaluate the predicted models for a target protein based on their pairwise structural similarity. For instance, some multi-model quality assessment methods [133, 134] employ clustering techniques to cluster models into different groups according to their structural similarities, and then select the center model in each group as the presumably best model most similar to the native structure.

Because of the difficulty of predicting the real quality of a predicted protein model

and the limitation of current techniques, one individual QA method generally cannot select the best model from the model pool. For example, single model QA methods may not be sensitive enough to rate a largely correct topology with significant local structural flaws higher than a native like but incorrect topology. Multiple model QA methods often fail when the majority of the predicted models is of bad qualities and is structurally similar to each other [10]. The model selected by the clustering-based methods usually is not the best model if models in the largest cluster are of bad quality.

Therefore, some protein tertiary structure prediction methods in recent CASP experiments tried to use the consensus of QA methods to evaluate the predicted models. For example, Zhang-Server [132] evaluated the predicted models using the consensus score of seven MQAP methods (e.g. the I-TASSER C-score [104], structural consensus measured by pair-wise TM-score [92], RW [51], RWplus [51], Dfire [135], Dope [83], verify3D [131]). MUFOLD [136] used three single-model QA methods (e.g. OPUS-CA [55], Dfire [135], ModelEvaluator [82]) to filter out poor models and then used consensus QA method (e.g. clustering) to evaluate the remaining models. Pcons [50] combined structural consensus [33] with a single model machine learning-based QA method ProQ2 [34] to evaluate the predicted models. Combining multiple quality assessment methods appeared to be an important approach to improve model evaluation as demonstrated in the CASP experiments. However, more extensive and sophisticated methods of integrating a large number of diverse and complementary QA methods need to be developed and analyzed.

Here we conduct a thorough analysis of our recently developed tertiary structure prediction methods based on a large-scale protein model quality assessment method – MULTICOM [137] on its template-based model predictions in 2014 CASP11 experiment in order to investigate the strengths and weaknesses of massive quality assessment methods. Unlike other tertiary structure prediction methods using only

one or several model quality assessment methods, MULTICOM integrated 14 complementary QA methods, which included both single-model QA methods and multi-model QA methods. Our tertiary structure prediction method participated in the CASP11 experiment as a human predictor and was ranked as one of top few methods for template-based protein structure modeling. The results indicate that the combination of the array of QA methods in conjunction with good model sampling and clustering is a promising direction for improving protein tertiary structure prediction.

8.3 Methods

Our MULTICOM method (human group MULTICOM in CASP11 experiment), although categorized as MULTICOM human predictor, is largely an automated method. MULTICOMs success, primarily, is because of exploiting appropriate use and combination of existing QA methods some of which we developed in house to complement existing methods, and not because of human intervention. Although the method has been discussed briefly in [137], here we discuss it comprehensively with an emphasis on the details of the method and an extensive evaluation strategy.

8.3.1 Massive protein model quality assessment for ranking protein structural models

Figure 8.1 provides an overview of the entire workflow of MULTICOM. MULTICOM takes a pool of structural models predicted by a variety of available protein structure prediction tools as input. This pool of models is supplied in parallel to both individual QA ranking methods and a model clustering tool - MUFOLD-CL [138]. The rankings generated by all QA methods are combined to obtain two consensus rankings. Since the consensus rankings may put similar models in the top ranks, in order to increase diversity in the top five selected models, the model clustering information is used to

replace some similar top-ranked models with structurally different models from other model clusters if necessary. The final selected models are further refined by a model combination approach [139].

Specifically, in CASP 11 experiment we used the hundreds of models for each target predicted by all CASP participants as input. Input models are first ranked by existing and our in-house developed single-model and multiple-model quality assessment (QA) methods - a total of 14 QA methods [137] whose software were available. These include 8 single-model methods, see **Table 8.1**, two in-house single-model QA methods: (a) MULTICOM-NOVEL, and (b) Modelcheck2 - an improved version of ModelEvaluator score [82]. We also use 4 multiple-model QA methods: (a) ModFOLDclust2 [24], (b) APOLLO [32], ' Pcons [50], and (d) QApr0 [10].

The integration of both single-model QA methods and multi-model QA methods is to leverage the strengths of the two kinds of methods and alleviate their weaknesses in order to rank models better than any of the individual method. The single-model methods may distinguish quality variation among good models, but may mistakenly favor a physically appealing, but low-quality models over largely correct models with significant local flaws. In contrast, the multi-model methods can often select some good models of average quality, but fail to identify models of better-than-average quality.

The rankings obtained using these individual methods are combined in two ways: (a) a mean is computed for each model to produce an average ranking of all 14 methods, (b) rankings of only 6 selected methods are used to produce an average ranking. The selected 6 methods include 4 single-model QA methods, MULTICOM-NOVEL, Modelcheck2, Dope [83], and OPUS_PSP [118], and 2 multiple-model methods, QApr0, and Pcons. Before the CASP11 experiment started, we tested all possible

Table 8.1: Publicly available single-model QA methods used in our MULTICOM method.

Method	Description
OPUS-PSP	Method based on side-chain derived orientation-dependent all-atom statistical potential
ProQ	Uses support vector machines to predict local as well as global quality of protein models; features of ProQ combined with updated structural and predicted features
RWplus	Method based on a new pair-wise distance-dependent atomic statistical potential function (RW) and side-chain orientation-dependent energy term
ModelEvaluator	Uses only structural features with support vector machine regression; assigns absolute GDT-TS score to a model by comparing secondary structure, relative solvent accessibility, contact map, and beta sheet topology with prediction from sequence
RF_CB_SRS_OD	Uses residue-based pairwise distance dependent statistical potential at various spatial pair separations
SELECTpro	Structure-based energy function with energy terms that include predicted secondary structure, solvent accessibility, contact map, beta-strand pairing, and side-chain hydrogen bonding
Dope	Uses probability theory to derive an atomic distance-dependent statistical potential
DFIRE	Based on statistical energy function that uses orientation dependent interaction from protein structures treating each polar atom as dipole

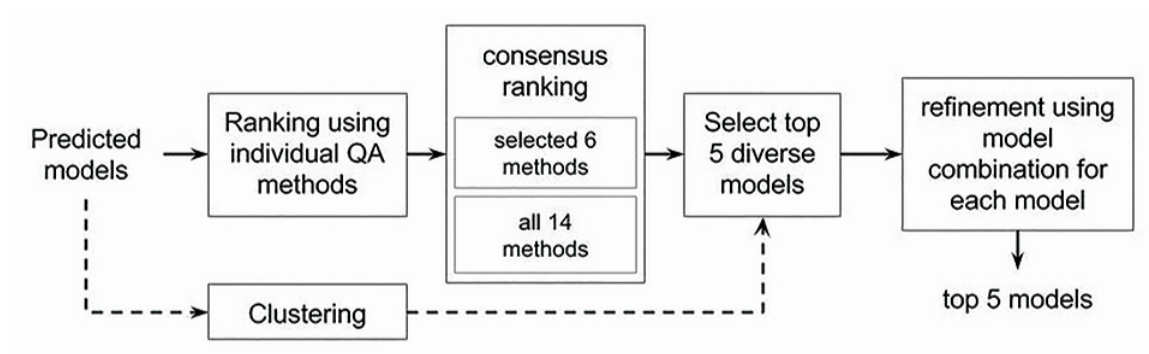


Figure 8.1: Workflow of MULTICOM large-scale model quality assessment method.

ways of combining the rankings on the data of 46 CASP10 targets, and found that combination of these selected 6 methods resulted in the lowest average loss of 0.037 GDT-TS score for the top one selected models in comparison with the best possible models, 0.02 GDT-TS score lower than combination of all 14 methods. However, since the benchmark testing was not comprehensive and may overfit the data, we retained both consensus approaches in our overall method for CASP11 experiment.

During CASP11, in order to choose between 6-methods based consensus and all 14-methods based consensus for our overall method, we predict the 'difficulty' of the target using the multi-model QA tool APOLLO in order to use separate methods for 'hard' and 'easy' cases [137]. APOLLO's score of greater than 0.3 generally hints higher quality of models because of high pairwise similarity between them, for example, when matching templates are found for the target. Hence, if APOLLO's similarity score for top ranked model is greater than 0.3, we compare this top model with the two top ranked models ranked by 6-method and all 14-method consensus, and finally select the ranking whose top model is more similar to APOLLO's top model. Here, in addition to using APOLLO to break the ties between the two consensus methods, the other rational is to filter out models of an incorrect topology that the consensus methods may accidentally rank at the top due to their use of many single-model quality assessment methods, by taking advantage of APOLLO's capability of selecting a good model in the case of easy prediction. According to our experiment on the CASP10 data, if the highest pairwise similarity score of the models measured by APOLLO is greater than 0.3, which often suggests the prediction is relatively easy, the top model selected by APOLLO generally has a good, but not necessarily the best topology. So, the idea of using the top model selected by APOLLO to re-rank the top models of the consensus methods here is to make sure the bad models with incorrect topologies selected by those methods will be completely ruled out. So, instead of directly using APOLLO's ranking, APOLLO is only used to provide some auxiliary information to

make sure one of the top models ranked by those methods would be correct when the prediction is relatively easy. Overall, the ranking of models is largely dominated by the two consensus methods, which performed better than using APOLLO score alone.

Furthermore, in order to further improve the reliability of the top one model, the top one models of the two consensus rankings and of the top server predictors (e.g., MULTICOM-CLUSTER and Zhang-Server) were compared with the top one model of APOLLO, and the model most similar to the top one model of APOLLO was used as the top one model in the final ranking without changing the ranking of all other models. However, if APOLLO's pairwise score for the top ranked model is less than or equal to 0.3, we consider the target to be 'hard', and use ab initio biased decision to make the selection of consensus ranking. For this, we predict secondary structure of input target sequence using PSIPRED [86, 140] and compare this with secondary structure of top ranked models in both consensus rankings by computing accuracy. Again, we select the consensus ranking whose top model has higher secondary structure similarity with predicted secondary structure. Despite the seemingly complexity of the modeling ranking strategy used by MULTICOM, the selection of top one model was largely determined by the two consensus methods with some influence from the other factors such as APOLLO's ranking scores, top server predictors' top models, and predicted secondary structures.

After selection of the appropriate ranking, instead of simply using top 5 ranked models as final rank, we use model clustering information to increase diversity in the top 5 list of models which is important especially for hard targets whose real structure is often very uncertain. As top five models selected by the approach above may be similar, if one is incorrect, all of them will fail. Therefore, it useful to include different models in the top five list. As such, MULTICOM always keeps the top two ranked models. If the model ranked third belongs to any of the clusters that the

previously selected models belong to, it will be removed from the complete rank and the remaining ranking below is lifted up repeatedly until we find a model in a different cluster. The process is repeated for fourth and fifth ranks ensuring diversity in the final top five models. In addition, we employed a model filtering technique to ensure that low quality models do not make their way up to the top 5 ranks. That is, during the re-ranking process, models that were ranked at bottom 10% by our in-house MULTICOM-NOVEL QA method were skipped because those models were mostly bad models such as largely unfolded models according to our experiment. Clustering is performed based on structural similarity of the models using MUFOLD-CL [138], a model clustering method based on the comparison of protein distance matrices. Our comparison of MUFOLD-CL with other techniques based on structural distance like RMSD [141], show similar accuracy but MUFOLD-CL runs much faster.

As the last step of MULTICOM method, a model combination approach is used to integrate each selected model with other similar models in the pool to obtain a refined model [139]. Basically, the Modeller is used to use each selected model and other similar models as templates to regenerate a number of combine models for a target. The model with minimum Modeller energy is selected as the refined model.

8.3.2 Summary of some individual QA methods used by MULTICOM

APOLLO, one of the 4 multiple-model methods we use, generates a pair-wise average GDT-TS score by performing a full pairwise comparison between all input models. The predicted GDT-TS score for a model is the average GDT-TS score between the model and all other models in the model pool. For models that are incomplete predictions (only parts of the target are predicted), the score is scaled down by the ratio of the models' sequence length divided by the target length. ModFOLDclust2, another multiple model method, uses mean score of the global predicted model quality

scores from the clustering based method ModFOLDclust and ModFOLDclustQ as its score to rank models. The Pcons protocol, on the other hand, analyzes input models looking for recurring three-dimensional structural patterns and assigns each model a score based on how common its three-dimensional structural patterns are in the whole model pool. Specifically, it estimates the quality of residues in a protein model by superimposing a model to all other models for the same target protein and calculating the S-score for each residue [50], which positively correlates with the level of recurrence of local conformations. Pcons predicts the global quality of a model by assigning a score reflecting the average similarity to the entire ensemble of models. The principle of Pcons is that recurring patterns are more likely to be correct than patterns that only occur in one or just a few models. The multiple model method, QApro, combines the scores of ModelEvaluator and APOLLO by summing the product of APOLLO’s pairwise GDT-TS and ModelEvaluator score normalized by the sum of all ModelEvaluator scores.

Besides the four multi-model QA methods and some publicly available single-model QA methods (see their description in Table I), we developed a new in-house single-model QA method, MULTICOM-NOVEL, which uses features extracted from the structure and sequence to predict model quality. To assess the global quality we used following features, (1) amino acids encoded by a 20-digit vector of 0 and 1, (2) difference between secondary structure and solvent accessibility of the model (parsed using DSSP) and the prediction by Spine X (and also SSpro4) from the protein sequence, (3) physical-chemical features (pairwise Euclidean distance score, surface polar score, weighted exposed score, total surface area score), (4) normalized quality score generated by ModelEvaluator [82], RWplus score [51], dope score [83], and RF_CB_SRS_OD score [96]. Performing statistical analysis for all global features on PISCES [142] database, we obtain feature density maps, i.e. the distribution of the

difference between the feature and GDT-TS score. For a model whose true quality is unknown, MULTICOM-NOVEL calculates the score for each feature, and combines these scores with the feature density maps to predict the model’s GDT-TS score. For local quality assessment, however, MULTICOM-NOVEL uses support vector machine with environment scores in different Euclidean distance ranges (8, 10, 12, 14, 16, 18, 20, and 30 angstrom) for each amino acid as input features. These environment scores extracted from a 15-residue sliding window that include secondary structure, solvent accessibility, and amino acid types, capture environmental information within a spatial sphere of a residue.

8.3.3 Evaluation

Together with 142 human and server predictors, our MULTICOM method was blindly tested on 42 human targets during CASP11 experiment. For the 39 TBM human domains of these 42 human targets, we downloaded native structures from CASP’s website (<http://www.predictioncenter.org/casp11/index.cgi>) for evaluation of the predicted structural models. We also downloaded the top 5 predictions by other server predictors to compare our results. All our evaluations use 6 different evaluation metrics GDT-HA [143, 144], SphereGrinder (SG) [145], RMSD, Local Distance Difference Test (LDDT) [69], GDC-all [144], Molprobity score [144]. GDT-HA is a high accuracy version of global distance test (GDT) measure, which has half the size of distance cut off comparing with GDT measure. SG (SphereGrinder) score is an all-atom local structure fitness score, which was designed to complement and add value to GDT measure. Root-mean-square deviation (RMSD) is a measure for the superimposed proteins, which evaluates the average backbone atoms’ distance. It is not ideal for comparing cases when the structures are substantially different [143]. The Local Distance Difference Test (LDDT) is a superposition-free score that evaluates local distance differences of all atoms in a model. GDC-all score is global measures

similar to GDT-HA, but it includes the positions of side-chain carbon atoms. Mol-probity is a knowledge based metrics, which evaluates the physical reasonableness of molecular models. Besides the six evaluation metrics we also use various kinds of Z-scores. Z-score of a model is calculated as the model’s GDT-TS score minus the average GDT-TS score of all the models in the model pool of a target divided by the standard deviation of all GDT-TS scores.

8.4 Results and discussions

First, we systematically evaluate the performance of MULTICOM using global and local quality metrics to perform comparative analysis of MULTICOM against all the server predictors participating in CASP11 on 39 TBM human domains.

The distributions of accuracy for individual targets are subsequently explored along with specific case studies highlighting the importance of clustering in conjunction with model selection. Finally, we investigated the consistency and robustness of our massive model quality assessment method compared to any individual quality assessment method.

Table 8.2 shows the six quality scores of the first models submitted by MULTICOM and 25 top performing server predictors for 39 TBM human domains. According to the average scores of the first models, MULTICOM performs better than the overall best performing server predictor (Zhang-Server) in terms of GDC, LDDT and Sph-Gr score, and slightly worse than Zhang-Server in terms of GDT-HA, Mol, and RMSD. **Table 8.3** reports the six quality scores of the best of top five models submitted by MULTICOM and the server predictors. According to the average score of the best of top five models, MULTICOM performs better than the overall best performing server predictor (Zhang-Server) in terms of GDT-HA, GDC, LDDT, and Sph-Gr score, and slightly worse than Zhang-Server in terms of Mol and RMSD score. The results show

that, in addition to effectively selecting good top-one models, MULTICOM applies clustering technique to increase the diversity of top five models [12] improves the quality of the best of five selected models.

Table 8.2: The average scores of the first models submitted by MULTICOM (bold) and top 25 performing server predictors.

Gr.	Name	Num	GDT-HA	GDC	Mol	LDDT	RMSD	Sph-Gr
277s	Zhang-Server	39	38.18	28.06	7.19	3.01	0.5	50.96
290	MULTICOM	39	38.14	28.38	7.06	3.2	0.51	51.15
499s	QUARK	39	37.59	27.65	7.76	2.96	0.49	48.95
038s	nns	39	34.91	26.06	8.81	2.88	0.46	48.03
008s	MULTICOM-CONSTRUCT	39	32.71	23.93	9.91	2.84	0.41	41.1
216s	myprotein-me	39	31.64	23.86	10.06	2.49	0.41	42.38
346s	HPredA	39	29.78	21.34	10.77	4.28	0.36	36.01
420s	MULTICOM-CLUSTER	39	32.49	23.96	10.42	2.9	0.42	39.49
279s	HPredX	39	31.86	23.16	11.69	4.27	0.38	38.9
050s	RaptorX	39	32.23	23.42	8.97	2.47	0.44	41.05
184s	BAKER-ROSETTASERVER	39	31.88	23.6	10.09	1.96	0.44	42.39
212s	FFAS-3D	39	30.46	21.67	10.02	3.27	0.38	36.69
300s	PhyreX	38	29.9	21.66	9.74	3.47	0.35	38.6
041s	MULTICOM-NOVEL	39	30.6	22.4	11.77	3.33	0.4	38.97
251s	TASSER-VMT	39	29.6	21.25	9.42	3.91	0.27	41.71
452s	FALCON_EnvFold	39	28.02	19.78	11.14	3.35	0.4	34.31
335s	FALCON_TOPO	39	27.92	19.56	11.19	3.43	0.39	34.11
381s	FALCON_MANUAL	39	28.02	19.69	10.94	3.33	0.39	34.51
414s	FALCON_MANUAL_X	39	27.86	19.61	11.26	3.37	0.39	34.18
479s	RBO_Aleph	36	25.04	17.69	10.78	1.69	0.37	33.44
410s	Pcons-net	39	27.66	19.83	14.88	2.97	0.37	32.84
022s	3D-Jigsaw-V5_1	37	27.57	19.43	9.6	3.06	0.33	32.98
133s	IntFOLD3	39	28.6	19.88	17.02	3.35	0.39	35.68
117s	raghavagps-tsppred	39	27.59	20.32	22.89	3.58	0.37	31.97
073s	SAM-T08-server	25	18.94	13.33	6.99	1.89	0.23	21.82

To evaluate the overall performance of MULTICOM in CASP11 TBM human targets relative to other server predictors and to explore any possible relationship between target difficulty and accuracy, we first investigated the median accuracy of first models submitted by MULTICOM and other server predictors against the number of residues in domain. **Figure 8.2** shows the evaluation as judged by six different quality metrics. The lack of correlation between target length and accuracy might indicate the presence reliable template(s) irrespective of sequence length and

Table 8.3: The average scores of the best of top five models submitted by MULTICOM (bold) and top 25 performing server predictors.

Gr.	Name	Num	GDT-HA	GDC	Mol	LDDT	RMSD	Sph-Gr
290	MULTICOM	39	41	31.1	6.85	2.99	0.53	54.48
277s	Zhang-Server	39	40.02	29.89	6.76	2.93	0.5	52.47
499s	QUARK	39	39.69	29.38	6.92	2.99	0.49	52.88
184s	BAKER-ROSETTASERVER	39	37.69	28.56	8.38	1.93	0.49	50.44
038s	nns	39	37.56	28.07	8	2.91	0.49	50.29
420s	MULTICOM-CLUSTER	39	34.98	26.1	10.08	2.91	0.44	43.26
216s	myprotein-me	39	34.05	25.89	10.26	2.45	0.42	44
041s	MULTICOM-NOVEL	39	34.76	25.93	9.96	3.24	0.43	43.34
008s	MULTICOM-CONSTRUCT	39	34.38	25.84	10.67	2.9	0.42	41.33
251s	TASSER-VMT	39	32.67	24.07	8.87	3.89	0.29	44.01
050s	RaptorX	39	32.74	23.59	8.78	2.42	0.44	41.55
346s	HHPredA	39	29.78	21.34	10.77	4.28	0.36	36.01
212s	FFAS-3D	39	32.09	22.93	9.87	3.3	0.39	39.7
279s	HHPredX	39	31.86	23.16	11.69	4.27	0.38	38.9
300s	PhyreX	39	31.17	22.55	10.09	3.53	0.37	39.43
454s	eThread	39	29.68	20.48	10.98	3.65	0.37	37.58
479s	RBO_Aleph	36	27.41	19.89	10.31	1.65	0.38	34.95
452s	FALCON_EnvFold	39	29.97	21.19	10.65	3.37	0.4	35.48
335s	FALCON_TOPO	39	29.85	20.63	10.72	3.42	0.4	35.54
381s	FALCON_MANUAL	39	29.83	21.04	10.69	3.35	0.4	35.72
073s	SAM-T08-server	27	21.18	15.17	7.27	2	0.24	25.43
414s	FALCON_MANUAL_X	39	29.8	21.14	10.5	3.35	0.41	35.61
237s	chuo-fams-server	39	30.11	21.96	14.46	3.96	0.36	32.79
410s	Pcons-net	39	29.46	21	14.53	2.91	0.38	34.66
466s	RaptorX-FM	14	8.47	5.17	3.71	1.25	0.07	9.11

the predictors' ability to select them accordingly.

To gain additional insight in target difficulty, we examined the percentage of sequence identity between the target and best template present in Protein Data Bank after optimal structural superposition (as provided by CASP11 assessors at http://www.predictioncenter.org/download_area/CASP11/templates/). In **Figure 8.3**, we report the accuracy of first models submitted by MULTICOM and the median performance of server predictors against the percentage of sequence identity for each of the six quality metrics. Once again, no systematic pattern can be observed from between the target difficulty and performance.

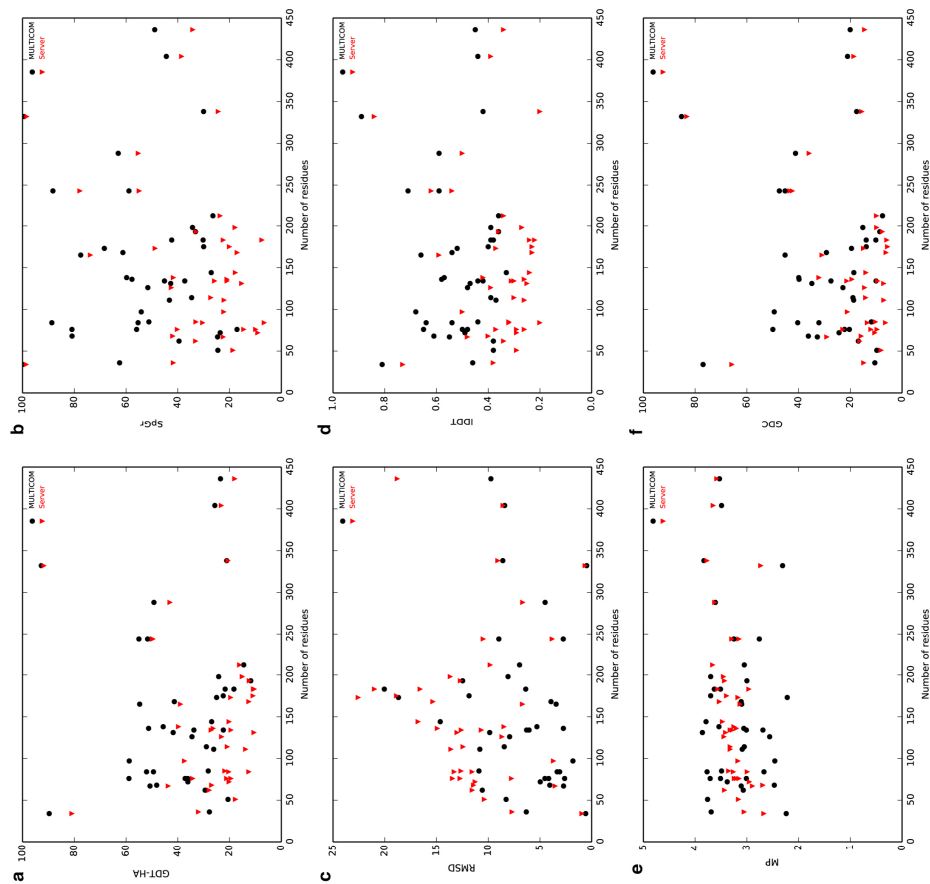


Figure 8.2: Performance of MULTICOM and server predictors with respect to number of residues in domain

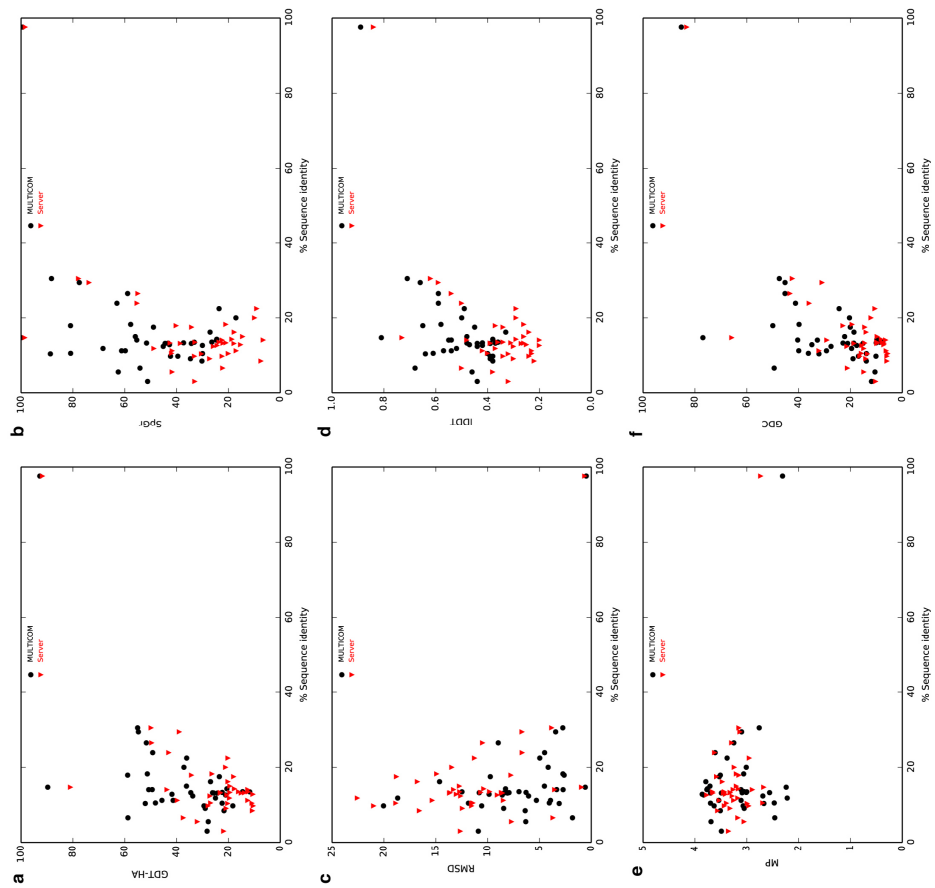


Figure 8.3: Performance of MULTICOM and server predictors with respect to difficulty of target

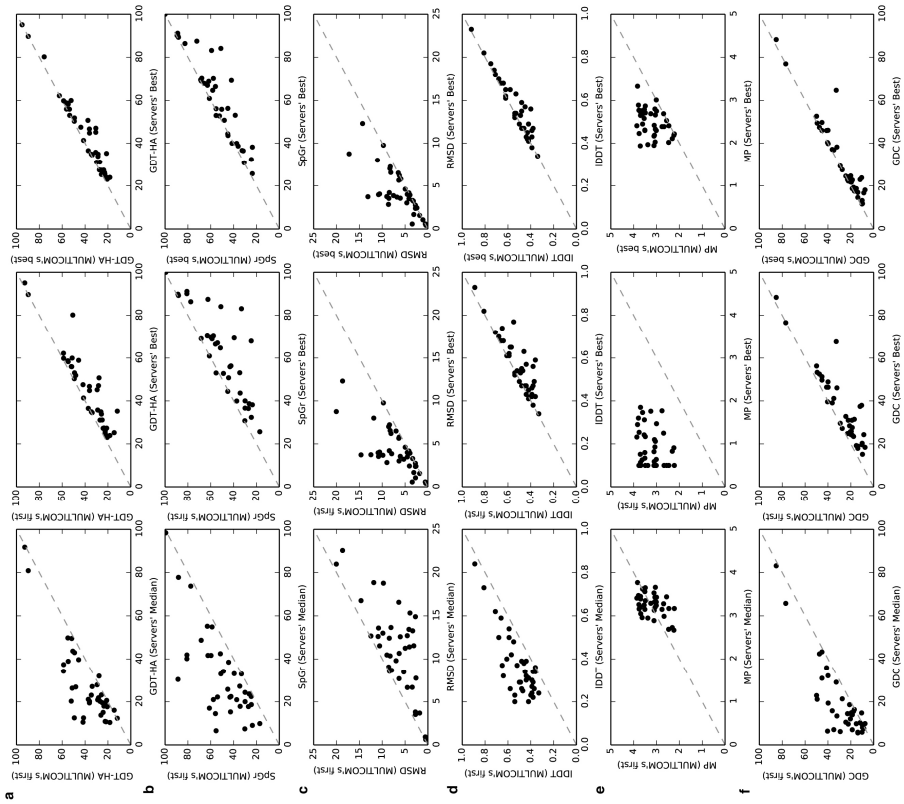


Figure 8.4: Accuracy of MULTICOM compared to other server predictors

In **Figure 8.4**, we examined the accuracy of the first models and the best of top five models submitted by MULTICOM and compared it with that of the server predictors. The comparison between the first models submitted by MULTICOM and the best server models (middle panels of **Figure 8.4**) indicates the ability of MULTICOM to often select good models from model pool. Furthermore, when the best of top five models submitted by MULTICOM are considered, MULTICOM’s performance of selecting some good models is even better (rightmost panels of **Figure 8.4**). This suggests that the massive integration of diverse protein quality assessment methods used in MULTICOM facilitates in selecting good models from the hundreds of alternative models generated by server predictors. MULTICOM’s performance in MolProbity was significantly worse than other quality metrics (**Figure 8.4e**), highlighting somewhat lack of physical reasonableness and enhanced stereochemistry in the submitted models. The problem may be caused by the poor quality of side chains and backbone atoms in the models, which could be corrected by using SCWRL [1] to repack the side chains, and using a physically-realistic all-atom MD/Monte Carlo simulation to refine the model.

To study the distribution and degree of accuracy on a per target basis and to understand the diversity of MULTICOM’s five submitted models, we calculated Z-score for each of the six quality metrics considering all predictors and analyzed the quartile plots of Z-scores by highlighting the five models submitted by MULTICOM. For several targets, MULTICOM’s performance was comparable with the best prediction submitted by any predictor. Moreover, the diversity between the five models submitted by MULTICOM indicates the effectiveness of using clustering together with model selection. Two representative examples are shown in **Figure 8.5** for CASP11 targets T0853-D1 and T0830-D1. For target T0853-D1, the first submitted model (highlighted in red) proved to be the best as judged by GDT-HA while the five submitted models were quite diverse covering different aspects of model quality. A

close resemblance can be observed between the experimental structure and prediction (**Figure 8.4a**). On the other hand, the fifth submitted model turned out to be the best in terms of GDT-HA for target T0830-D1 while having lesser diversity between five submitted models. In both the cases, the best out of five models by MULTICOM achieved accuracy close to the best-submitted model by any predictor.

In addition to assessing the overall performance, we specifically examined how massively integration of diverse protein quality assessment methods helps in improving the ranking of template-based models compared to any individual QA method and explored how average accuracy of the pool of model impacted model selection. **Figure 8.6** presents the GDT-HA of the top model selected by each of the single QA and MULTICOM with respect to the median GDT-HA score of the ensemble of server predictors. The overall accuracy of MULTICOM is observed to be better than individual QA methods. Several additional interesting insights can be observed. For example, when the median GDT-HA scores are very high, several clustering-based methods display relatively poor performance compared to single model QA methods. One explanation for this could be that the presence of an easily identifiable template and relatively straightforward target-template alignment, causing almost all the server methods to perform similarly. This results in less diversity in the model ensemble and subsequently affects the performance of clustering-based QA techniques that favor average-quality models (i.e. the center of a model cluster).

Table 8.4 shows the comparison for the top 1 model selected by MULTICOM and each QA method based on GDT-HA score. As we can see from the table, in terms of average GDT-HA, and also Z-score on all targets, MULTICOM gets the best performance. In addition, we do a Wilcoxon signed ranked sum test on the top 1 model's Z-score difference between MULTICOM and each QA method, and the p-value is shown in the table. The QA method QApr, ModelEva, and Proq2 actually

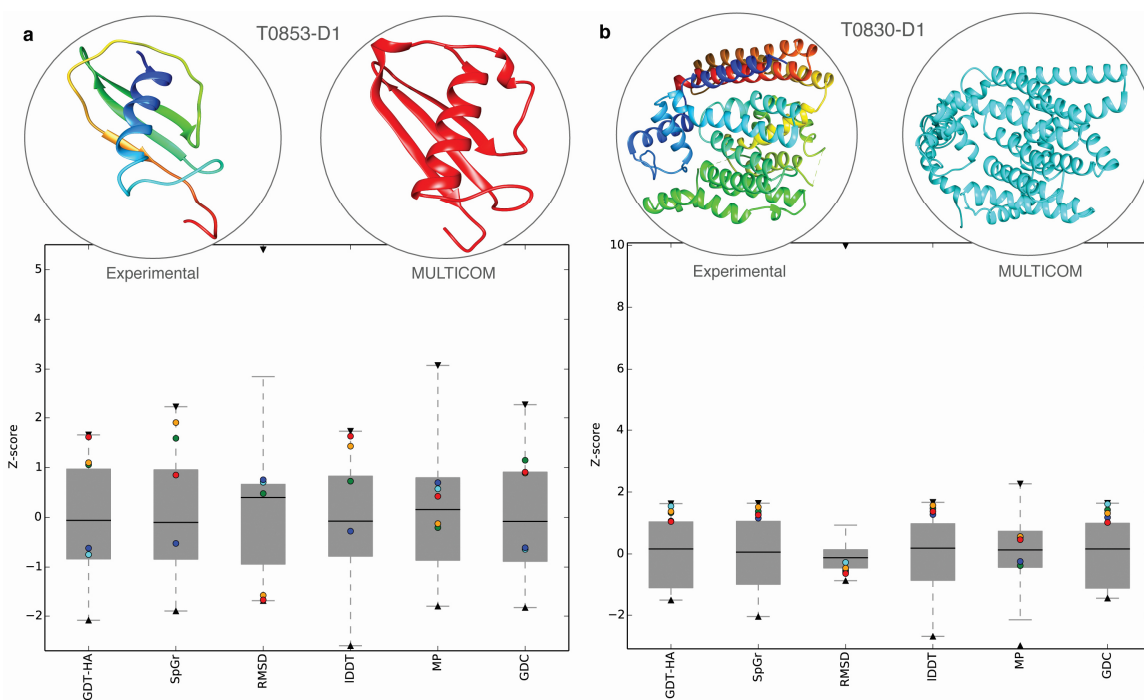


Figure 8.5: Case study for CASP11 targets T0853-D1 and T0830-D1.

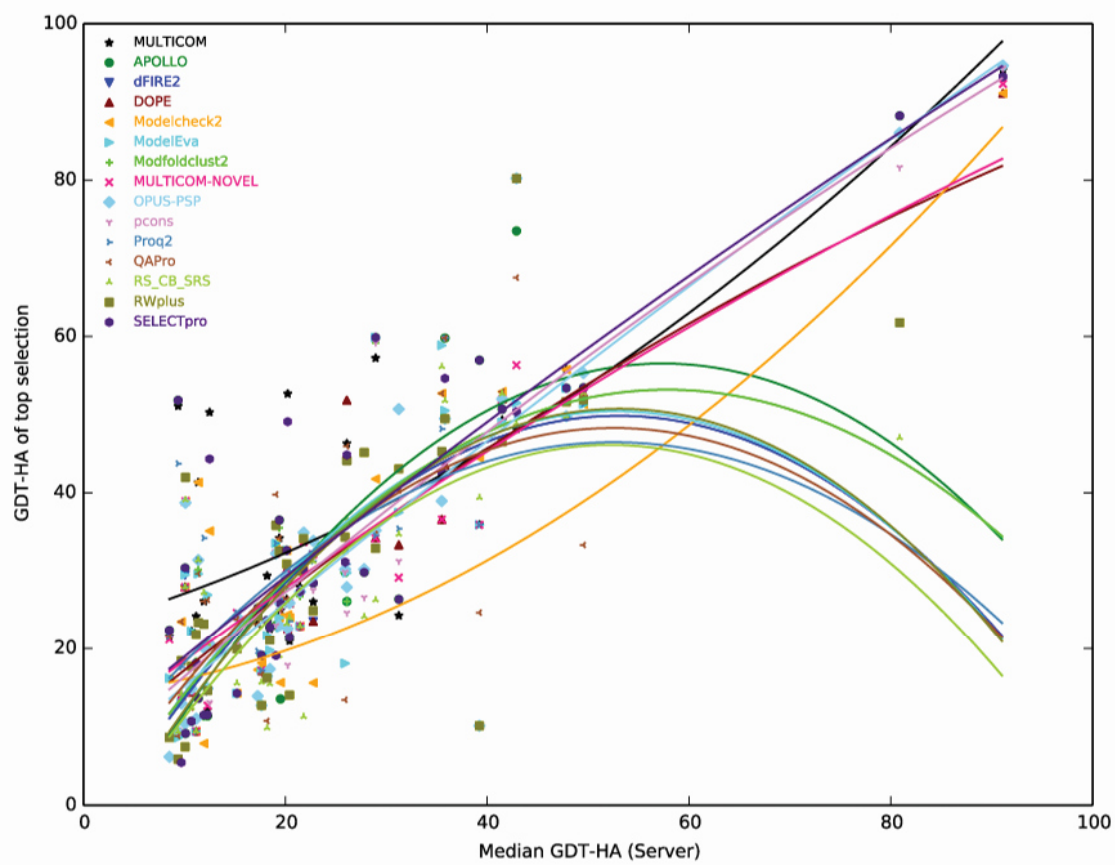


Figure 8.6: Comparison of MULTICOM with individual QA methods.

perform very well on these TBM targets, and the difference between MULTICOM and them is not very significant given the confidence level 0.05. However, MULTICOM is significantly different with other QA methods based on the selected top 1 model’s Z score, suggesting Z-score is a more sensitive measure of the difference in model quality.

Table 8.4: Comparison of MULTICOM with each QA method on the average GDT-HA score and Z-score of the top models selected, and the significant of each QA method.

QA score name on all human targets	Ave. GDT-HA score on all	Ave. Z score on all	p-value of Z score diff.
MULTICOM	36.3	1.417	-
SELECTpro	33	0.889	0.0159
Proq2	31.8	1.158	0.0558
Modelcheck2	31.8	0.959	0.0208
MULTICOM-NOVEL	31.4	0.936	0.0059
Pcons	31.1	0.681	0.0125
ModelEva	31.1	1.086	0.0829
APOLLO	30.9	0.83	0.0463
Modfoldclust2	30.9	0.888	0.0425
QApr	30.9	1.117	0.195
Dope	30.8	0.835	0.0061
Dfire2	30.4	0.997	0.0224
OPUS-PSP	29.9	0.635	0.0016
RWplus	29.8	0.932	0.0161
RF_CB_SRS	27.6	0.489	0.0017

To investigate MULTICOM’s ability to rank the models, we studied the GDT-HA score of a model with respect to its ranking by MULTICOM on a per target basis. In **Figure 8.7**, we present two typical example of MULTICOM’s ranking. For target T0822-D1, shown in **Figure 8.7a**, the majority of the models has GDT-HA score less than 0.15 GDT-HA score and was ranked low by MULTICOM, while few models have GDT-HA score more than 0.25 and were usually ranked higher. MULTICOM was able to select the better model compared to other QA methods, although it missed the best model myprotein-me`TS4 in the server model pool. In case of target T0838-D1,

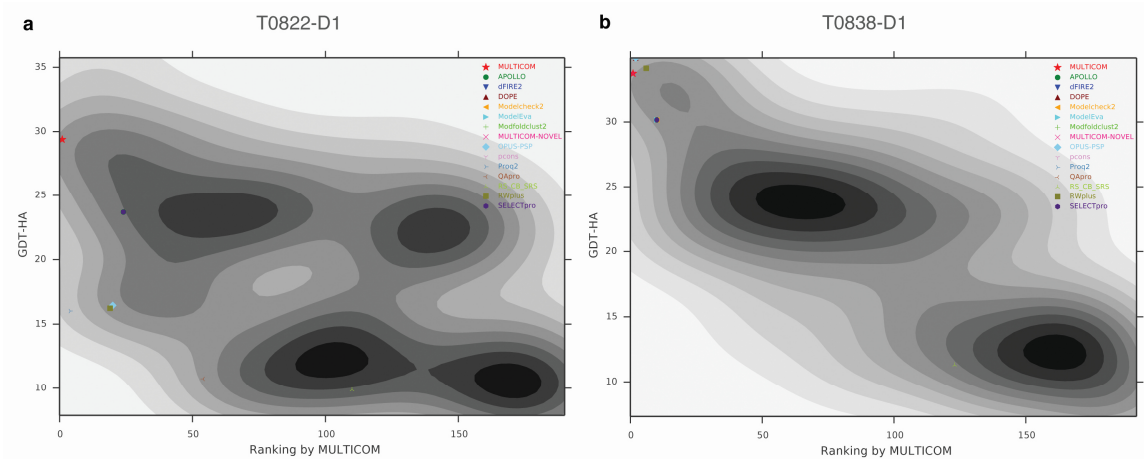


Figure 8.7: Landscape of MULTICOM's ranking.

reported in **Figure 8.7b**, clear convergence to the optimal model can be observed as shown by distinct inverted funnel shaped ranking landscape. Even though in this case MULTICOM was neither able to pick the best model myprotein-me¹TS1 in the server model pool, nor performed better than all the other QA methods. However, the performance of MULTICOM and the optimal QA methods (OPUS-PSP or DOPE) were comparable.

8.5 Conclusions

We conducted a comprehensive analysis of our CASP11 human tertiary structure predictor MULTICOM on template-based targets. Our experiment demonstrates that the massive integration of diverse, complementary quality assessment methods is a promising approach to address the significant challenge of ranking protein models and improves the accuracy and reliability of template-based modeling. In order to further improve the template-based modeling, on one hand more accurate tertiary structure prediction methods need to be developed to generate a large portion of good structural models, and on the other hand more sensitive model quality assessment methods need to be included to reliably select good models from a pool of models that may only contain a few good models.

Bibliography

- [1] G.G. Krivov, M.V. Shapovalov, and R.L. Dunbrack Jr. Improved prediction of protein sidechain conformations with scwrl4. *Proteins: Structure, Function, and Bioinformatics*, 77(4):778–795, 2009.
- [2] B. Mauroy, M. Filoche, E. R. Weibel, and B. Sapoval. An optimal bronchial tree may be dangerous. *Nature*, 427:633–636, 2004.
- [3] Jilong Li, Renzhi Cao, and Jianlin Cheng. A large-scale conformation sampling and evaluation server for protein tertiary structure prediction and its assessment in casp11. *BMC bioinformatics*, 16(1):337, 2015.
- [4] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.
- [5] Michael KK Leung, Andrew Delong, Babak Alipanahi, and Brendan J Frey. Machine learning in genomic medicine: A review of computational problems and data sets. *Proceedings of the IEEE*, 104(1):176–197, 2016.
- [6] E. Lieberman-Aiden, N.L. Van Berkum, L. Williams, M. Imakaev, T. Ragoczy, A. Telling, I. Amit, B.R. Lajoie, P.J. Sabo, and M.O. Dorschner. Comprehensive mapping of long-range interactions reveals folding principles of the human genome. *Science*, 326(5950):289–293, 2009.

- [7] Renzhi Cao and Jianlin Cheng. Deciphering the association between gene function and spatial gene-gene interactions in 3d human genome conformation. *BMC genomics*, 16(1):880, 2015.
- [8] Yuxiang Jiang, Tal Ronnen Oron, Wyatt T Clark, Asma R Bankapur, Daniel D’Andrea, Rosalba Lepore, Christopher S Funk, Indika Kahanda, Karin M Ver-spoor, and Asa Ben-Hur. An expanded evaluation of protein function prediction methods shows an improvement in accuracy. *arXiv preprint arXiv:1601.00891*, 2016.
- [9] Renzhi Cao and Jianlin Cheng. Integrated protein function prediction by mining function associations, sequences, and protein-protein and gene-gene interaction networks. *Methods*, 93:84–91, 2016.
- [10] Renzhi Cao, Zheng Wang, and Jianlin Cheng. Designing and evaluating the multicom protein local and global model quality prediction methods in the casp10 experiment. *BMC Structural Biology*, 14(1):13, 2014.
- [11] Renzhi Cao and Jianlin Cheng. Protein single-model quality assessment by feature-based probability density functions. *Scientific Reports*, 6:23990, 2016.
- [12] Renzhi Cao, Debswapna Bhattacharya, Badri Adhikari, Jilong Li, and Jianlin Cheng. Large-scale model quality assessment for improving protein tertiary structure prediction. *Bioinformatics*, 31(12):i116–i123, 2015.
- [13] Renzhi Cao, Debswapna Bhattacharya, Badri Adhikari, Jilong Li, and Jianlin Cheng. Massive integration of diverse protein quality assessment methods to improve template based modeling in casp11. *Proteins: Structure, Function, and Bioinformatics*, 2015.
- [14] DT Jones and LJ McGuffin. Assembling novel protein folds from super-secondary structural fragments. *Proteins*, 53(S6):480 – 485, 2003.

- [15] KT Simons, C Kooperberg, E Huang, and D Baker. Assembly of protein tertiary structures from fragments with similar local sequences using simulated annealing and bayesian scoring functions. *Journal of Molecular Biology*, 268(1):209–225, 1997.
- [16] J Cheng, J Eickholt, Z Wang, and X Deng. Recursive protein modeling: a divide and conquer strategy for protein structure prediction and its case study in casp9. *J Bioinform Comput Biol*, 10(3), 2012. doi:10.1142/S0219720012420036.
- [17] A Zemla, C Venclovas, J Moult, and K Fidelis. Processing and analysis of casp3 protein structure predictions. *Proteins*, 37(S3):22 – 29, 1999.
- [18] A Zemla, C Venclovas, J Moult, and K Fidelis. Processing and evaluation of predictions in casp4. *Proteins*, 45(S5):13 – 21, 2002.
- [19] Y. Yang and Y. Zhou. Ab initio folding of terminal segments with secondary structures reveals the fine difference between two closely related allatom statistical energy functions. *Protein Science*, 17(7):1212–1219, 2008.
- [20] J Lee, D Lee, H Park, EA Coutsias, and C Seok. Protein loop modeling by using fragment assembly and analytical loop closure. *Proteins*, 78(16):3428 – 3436, 2010.
- [21] P Liu, F Zhu, DN Rassokhin, and DK Agrafiotis. A self-organizing algorithm for modeling protein loops. *PLoS Comput Biol*, 5(8):e1000478, 2009.
- [22] J Zhang, Q Wang, B Barz, Z He, I Kosztin, Y Shang, and D Xu. Mufold: a new solution for protein 3d structure prediction. *Proteins*, 78(5):1137 – 1152, 2010.

- [23] Natalia Naumova, Maxim Imakaev, Geoffrey Fudenberg, Ye Zhan, Bryan R Lajoie, Leonid A Mirny, and Job Dekker. Organization of the mitotic chromosome. *Science*, 342(6161):948–953, 2013.
- [24] Liam J McGuffin and Daniel B Roche. Rapid model quality assessment for protein structure predictions using the comparison of multiple models without structural alignments. *Bioinformatics*, 26(2):182–188, 2010.
- [25] Tung BK Le, Maxim V Imakaev, Leonid A Mirny, and Michael T Laub. High-resolution mapping of the spatial organization of a bacterial chromosome. *Science*, 342(6159):731–734, 2013.
- [26] Stefan Grob, Marc W Schmid, and Ueli Grossniklaus. Hi-c analysis in arabidopsis identifies the knot, a structure with similarities to the flamenco locus of drosophila. *Molecular cell*, 55(5):678–693, 2014.
- [27] SF Altschul, TL Madden, AA Schaffer, J Zhang, Z Zhang, W Miller, and DJ Lipman. Gapped blast and psi-blast: a new generation of protein database search programs. *Nucleic Acids Res*, 25(17):3389 – 3402, 1997.
- [28] Songling Li and Dieter W Heermann. Transcriptional regulatory network shapes the genome structure of saccharomyces cerevisiae. *Nucleus*, 4(3):216–228, 2013.
- [29] M Ashburner, CA Ball, JA Blake, D Botstein, H Butler, JM Cherry, AP Davis, K Dolinski, SS Dwight, and JT Eppig. Gene ontology: tool for the unification of biology. *Nature genetics*, 25(1):25–29, 2000.
- [30] Z Du, L Li, CF Chen, PS Yu, and JZ Wang. G-sesame: web tools for go-term-based gene similarity analysis and knowledge discovery. *Nucleic Acids Research*, 37(Web Server issue):W345, 2009.

- [31] L.J. McGuffin. The modfold server for the quality assessment of protein structural models. *Bioinformatics*, 24(4):586–587, 2008.
- [32] Z Wang, J Eickholt, and J Cheng. Apollo: a quality assessment service for single and multiple protein models. *Bioinformatics*, 27(12):1715 – 1716, 2011.
- [33] P. Larsson, M.J. Skwark, B. Wallner, and A. Elofsson. Assessment of global and local model quality in casp8 using pcons and proq. *Proteins: Structure, Function, and Bioinformatics*, 77(S9):167–172, 2009.
- [34] A Ray, E Lindahl, and B Wallner. Improved model quality assessment using proq2. *BMC bioinformatics*, 13(1):224, 2012.
- [35] A Kryshtafovych, A Barbato, K Fidelis, B Monastyrskyy, T Schwede, and A Tramontano. Assessment of the assessment: evaluation of the model quality estimates in casp10. *Proteins*, 82(Suppl 2):112 – 26, 2013. doi:10.1002/prot.24347.
- [36] Predrag Radivojac, Wyatt T Clark, Tal Ronnen Oron, Alexandra M Schnoes, Tobias Wittkop, Artem Sokolov, Kiley Graim, Christopher Funk, Karin Verspoor, and Asa Ben-Hur. A large-scale evaluation of computational protein function prediction. *Nature methods*, 10(3):221–227, 2013.
- [37] Badri Adhikari, Debswapna Bhattacharya, Renzhi Cao, and Jianlin Cheng. Confold: Residueresidue contactguided ab initio protein folding. *Proteins: Structure, Function, and Bioinformatics*, 83(8):1436–1449, 2015.
- [38] R Liithy, JU Bowie, and D Eisenberg. Assessment of protein models with three-dimensional profiles. *Nature*, 356:83 – 85, 1992.

- [39] Pascal Benkert, Marco Biasini, and Torsten Schwede. Toward the estimation of the absolute quality of individual protein structure models. *Bioinformatics*, 27(3):343–350, 2011.
- [40] Z. Wang, X.C. Zhang, M.H. Le, D. Xu, G. Stacey, and J. Cheng. A protein domain co-occurrence network approach for predicting protein function and inferring species phylogeny. *PLoS ONE*, 6(3):e17906, 2011.
- [41] Jilong Li, Debswapna Bhattacharya, Renzhi Cao, Badri Adhikari, Xin Deng, Jesse Eickholt, and Jianlin Cheng. *The MULTICOM Protein Tertiary Structure Prediction System*, volume 1137, pages 29–41. Springer New York, New York, NY, 2014.
- [42] Andriy Kryshchak, Alessandro Barbato, Bohdan Monastyrskyy, Krzysztof Fidelis, Torsten Schwede, and Anna Tramontano. 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.
- [43] B Rost. Protein structure prediction in 1d, 2d, and 3d. *Encyclopaedia Comput Chem*, 3:2242 – 2255, 1998.
- [44] Taeho Jo and Jianlin Cheng. Improving protein fold recognition by random forest. *BMC bioinformatics*, 15(Suppl 11):S14, 2014.
- [45] Hou J. Eickholt J. & Cheng J. Jo, T. Improving protein fold recognition by deep learning networks. *Sic. Rep*, 5:17573, 2015.
- [46] Bernhard Schölkopf and Alexander J Smola. *Learning with kernels: Support vector machines, regularization, optimization, and beyond*. MIT press, 2002.

- [47] Troy Hawkins, Meghana Chitale, Stanislav Luban, and Daisuke Kihara. Pfp: Automated prediction of gene ontology functional annotations with confidence scores using protein sequence data. *Proteins: Structure, Function, and Bioinformatics*, 74(3):566–582, 2009.
- [48] Guoli Wang and Roland L Dunbrack. Pisces: a protein sequence culling server. *Bioinformatics*, 19(12):1589–1591, 2003.
- [49] Elaine R Mardis. A decade/’s perspective on dna sequencing technology. *Nature*, 470(7333):198–203, 2011.
- [50] B Wallner and A Elofsson. Identification of correct regions in protein models using structural, alignment, and consensus information. *Protein Sci*, 15(4):900 – 913, 2009.
- [51] Jian Zhang and Yang Zhang. 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, 2010.
- [52] Haiyou Deng, Ya Jia, and Yang Zhang. 3drobot: automated generation of diverse and well-packed protein structure decoys. *Bioinformatics*, page btv601, 2015.
- [53] Karolis Uziela and Björn Wallner. Proq2: Estimation of model accuracy implemented in rosetta. *Bioinformatics*, page btv767, 2016.
- [54] Hongyi Zhou and Jeffrey Skolnick. Goap: a generalized orientation-dependent, all-atom statistical potential for protein structure prediction. *Biophysical journal*, 101(8):2043–2052, 2011.

- [55] Yinghao Wu, Mingyang Lu, Mingzhi Chen, Jialin Li, and Jianpeng Ma. Opusca: A knowledge-based potential function requiring only c positions. *Protein Science*, 16(7):1449–1463, 2007.
- [56] Y. Yang and Y. Zhou. Specific interactions for ab initio folding of protein terminal regions with secondary structures. *Proteins: Structure, Function, and Bioinformatics*, 72(2):793–803, 2008.
- [57] Jesse Eickholt and Jianlin Cheng. Predicting protein residue–residue contacts using deep networks and boosting. *Bioinformatics*, 28(23):3066–3072, 2012.
- [58] Geoffrey E Hinton. Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800, 2002.
- [59] Troy Hawkins, Stanislav Luban, and Daisuke Kihara. Enhanced automated function prediction using distantly related sequences and contextual association by pfp. *Protein Science*, 15(6):1550–1556, 2006.
- [60] Meghana Chitale, Troy Hawkins, Changsoon Park, and Daisuke Kihara. Esg: extended similarity group method for automated protein function prediction. *Bioinformatics*, 25(14):1739–1745, 2009.
- [61] Meghana Chitale, Ishita K Khan, and Daisuke Kihara. In-depth performance evaluation of pfp and esg sequence-based function prediction methods in cafa 2011 experiment. *BMC bioinformatics*, 14(Suppl 3):S2, 2013.
- [62] Ishita K Khan, Qing Wei, Meghana Chitale, and Daisuke Kihara. Pfp/esg: automated protein function prediction servers enhanced with gene ontology visualization tool. *Bioinformatics*, page btu646, 2014.

- [63] Domenico Cozzetto, Daniel WA Buchan, Kevin Bryson, and David T Jones. Protein function prediction by massive integration of evolutionary analyses and multiple data sources. *BMC bioinformatics*, 14(Suppl 3):S1, 2013.
- [64] Marco Falda, Stefano Toppo, Alessandro Pescarolo, Enrico Lavezzo, Barbara Di Camillo, Andrea Facchinetti, Elisa Cilia, Riccardo Velasco, and Paolo Fontana. Argot2: a large scale function prediction tool relying on semantic similarity of weighted gene ontology terms. *BMC bioinformatics*, 13(Suppl 4):S14, 2012.
- [65] Paolo Fontana, Alessandro Cestaro, Riccardo Velasco, Elide Formentin, and Stefano Toppo. Rapid annotation of anonymous sequences from genome projects using semantic similarities and a weighting scheme in gene ontology. *PLoS One*, 4(2):e4619, 2009.
- [66] Christopher S Funk, Indika Kahanda, Asa Ben-Hur, and Karin M Verspoor. Evaluating a variety of text-mined features for automatic protein function prediction with gostruct. *Journal of biomedical semantics*, 6(1):9, 2015.
- [67] Patrik Koskinen, Petri Törönen, Jussi Nokso-Koivisto, and Liisa Holm. Pannzer: high-throughput functional annotation of uncharacterized proteins in an error-prone environment. *Bioinformatics*, 31(10):1544–1552, 2015.
- [68] Liang Lan, Nemanja Djuric, Yuhong Guo, and Slobodan Vucetic. Ms-knn: protein function prediction by integrating multiple data sources. *BMC bioinformatics*, 14(Suppl 3):S8, 2013.
- [69] Valerio Mariani, Marco Biasini, Alessandro Barbato, and Torsten Schwede. lddt: a local superposition-free score for comparing protein structures and models using distance difference tests. *Bioinformatics*, 29(21):2722–2728, 2013.

- [70] Kliment Olechnovič, Eleonora Kulberkytė, and eslovas Venclovas. Cadscore: A new contact area differencebased function for evaluation of protein structural models. *Proteins: Structure, Function, and Bioinformatics*, 81(1):149–162, 2013.
- [71] Akshay Yadav and Valadi Krishnamoorthy Jayaraman. Structure based function prediction of proteins using fragment library frequency vectors. *Bioinformatics*, 8(19):953, 2012.
- [72] F. Eisenhaber, B. Persson, and P. Argos. Protein structure prediction: recognition of primary, secondary, and tertiary structural features from amino acid sequence. *Critical reviews in biochemistry and molecular biology*, 30(1):1–94, 1995.
- [73] J. Cheng. A multi-template combination algorithm for protein comparative modeling. *BMC Structural Biology*, 8(1):18, 2008.
- [74] GG Krivov, MV Shapovalov, and RL Dunbrack. Improved prediction of protein side-chain conformations with scwrl4. *Proteins*, 77(4):778 – 795, 2009.
- [75] RD Page. Treeview: an application to display phylogenetic trees on personal computer. *Comp Appl Biol Sci*, 12:357 – 358, 1996.
- [76] Andy Liaw and Matthew Wiener. Classification and regression by randomforest. *R news*, 2(3):18–22, 2002.
- [77] Zheng Wang, Renzhi Cao, and Jianlin Cheng. Three-level prediction of protein function by combining profile-sequence search, profile-profile search, and domain co-occurrence networks. *BMC bioinformatics*, 14(Suppl 3):S3, 2013.
- [78] David T Jones, WR Taylor, and Janet M Thornton. A new approach to protein fold recognition. *nature*, 1992.

- [79] Adam Zemla. Lga: a method for finding 3d similarities in protein structures. *Nucleic Acids Research*, 31(13):3370–3374, 2003.
- [80] L.J. McGuffin. Prediction of global and local model quality in casp8 using the modfold server. *Proteins: Structure, Function, and Bioinformatics*, 77(S9):185–190, 2009.
- [81] Q Wang, K Vantasin, D Xu, and Y Shang. Mufold-wqa: a new selective consensus method for quality assessment in protein structure prediction. *Proteins*, 79(SupplementS10):185 – 95, 2011. doi:10.1002/prot.23185.
- [82] Z Wang, AN Tegge, and J Cheng. Evaluating the absolute quality of a single protein model using structural features and support vector machines. *Proteins*, 75(3):638 – 647, 2009.
- [83] Minyi Shen and Andrej Sali. Statistical potential for assessment and prediction of protein structures. *Protein Science*, 15(11):2507–2524, 2006.
- [84] Renzhi Cao, Taeho Jo, and Jianlin Cheng. Evaluation of protein structural models using random forests. *arXiv preprint arXiv:1602.04277*, 2016.
- [85] Björn Wallner and Arne Elofsson. Can correct protein models be identified? *Protein Science*, 12(5):1073–1086, 2003.
- [86] Liam J McGuffin, Kevin Bryson, and David T Jones. The psipred protein structure prediction server. *Bioinformatics*, 16(4):404–405, 2000.
- [87] Pascal Benkert, Silvio CE Tosatto, and Dietmar Schomburg. Qmean: A comprehensive scoring function for model quality assessment. *Proteins: Structure, Function, and Bioinformatics*, 71(1):261–277, 2008.
- [88] László Kaján and Leszek Rychlewski. Evaluation of 3d-jury on casp7 models. *BMC bioinformatics*, 8(1):304, 2007.

- [89] Y. Zhang and J. Skolnick. Spicker: A clustering approach to identify nearnative protein folds. *Journal of computational chemistry*, 25(6):865–871, 2004.
- [90] J. Cheng, AZ Randall, MJ Sweredoski, and P. Baldi. Scratch: a protein structure and structural feature prediction server. *Nucleic acids research*, 33(suppl 2):W72–W76, 2005.
- [91] Wolfgang Kabsch and Christian Sander. Dictionary of protein secondary structure: pattern recognition of hydrogenbonded and geometrical features. *Biopolymers*, 22(12):2577–2637, 1983.
- [92] Y Zhang and J Skolnick. Scoring function for automated assessment of protein structure template quality. *Proteins: Structure, Function, and Bioinformatics*, 57(4):702–710, 2004.
- [93] Roderic DM Page. Treeview. *An application to display phylogenetic trees on personal computer. Comp Appl Biol Sci*, 12:357–358, 1996.
- [94] Renzhi Cao, Debswapna Bhattacharya, Badri Adhikari, Jilong Li, and Jianlin Cheng. Massive integration of diverse protein quality assessment methods to improve template based modeling in casp11. *Proteins: Structure, Function, and Bioinformatics*, 2015.
- [95] Renzhi Cao, Zheng Wang, Yiheng Wang, and Jianlin Cheng. Smoq: a tool for predicting the absolute residue-specific quality of a single protein model with support vector machines. *BMC bioinformatics*, 15(1):120, 2014.
- [96] Dmitry Rykunov and András Fiser. Effects of amino acid composition, finite size of proteins, and sparse statistics on distancedependent statistical pair potentials. *Proteins: Structure, Function, and Bioinformatics*, 67(3):559–568, 2007.

- [97] M Kalman and N Ben-Tal. Quality assessment of protein model-structures using evolutionary conservation. *Bioinformatics*, 26(10):1299 – 1307, 2010.
- [98] Avinash Mishra, Satyanarayan Rao, Aditya Mittal, and B Jayaram. Capturing native/native like structures with a physico-chemical metric (pcsm) in protein folding. *Biochimica et Biophysica Acta (BBA)-Proteins and Proteomics*, 1834(8):1520–1531, 2013.
- [99] Eshel Faraggi, Tuo Zhang, Yuedong Yang, Lukasz Kurgan, and Yaoqi Zhou. Spine x: improving protein secondary structure prediction by multistep learning coupled with prediction of solvent accessible surface area and backbone torsion angles. *Journal of computational chemistry*, 33(3):259–267, 2012.
- [100] Matthew Jacobson and Andrej Sali. Comparative protein structure modeling and its applications to drug discovery. *Annu. Rep. Med. Chem*, 39(85):259–274, 2004.
- [101] Thorsten Joachims. Optimizing search engines using clickthrough data. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 133–142. ACM, 2002.
- [102] Chao Wang, Haicang Zhang, Wei-Mou Zheng, Dong Xu, Jianwei Zhu, Bing Wang, Kang Ning, Shiwei Sun, Shuai Cheng Li, and Dongbo Bu. Falcon@ home: a high-throughput protein structure prediction server based on remote homologue recognition. *Bioinformatics*, page btv581, 2015.
- [103] Shuai Cheng Li, Dongbo Bu, Jinbo Xu, and Ming Li. Fragmenthmm: A new approach to protein structure prediction. *Protein Science*, 17(11):1925–1934, 2008.
- [104] Y. Zhang. I-tasser server for protein 3d structure prediction. *BMC bioinformatics*, 9(1):40, 2008.

- [105] J Peng and J Xu. Raptorx: exploiting structure information for protein alignments by statistical inference. *Proteins*, 79(S10):161 – 171, 2011.
- [106] Wenyu Zhang, Jiajia Chen, Yang Yang, Yifei Tang, Jing Shang, and Bairong Shen. A practical comparison of de novo genome assembly software tools for next-generation sequencing technologies. *PLoS One*, 6(3):e17915, 2011.
- [107] Debswapna Bhattacharya and Jianlin Cheng. De novo protein conformational sampling using a probabilistic graphical model. *Scientific Reports*, 5, 2015.
- [108] Tong Liu, Yiheng Wang, Jesse Eickholt, and Zheng Wang. Benchmarking deep networks for predicting residue-specific quality of individual protein models in casp11. *Scientific Reports*, 6:19301, 2016.
- [109] Debswapna Bhattacharya, Renzhi Cao, and Jianlin Cheng. Unicon3d: de novo protein structure prediction using united-residue conformational search via step-wise, probabilistic sampling. *Bioinformatics*, page btw316, 2016.
- [110] Will Y Zou, Xiaoyu Wang, Miao Sun, and Yuanqing Lin. Generic object detection with dense neural patterns and regionlets. *arXiv preprint arXiv:1404.4316*, 2014.
- [111] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George van den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, and Marc Lanctot. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016.
- [112] Bino John and Andrej Sali. Comparative protein structure modeling by iterative alignment, model building and model assessment. *Nucleic acids research*, 31(14):3982–3992, 2003.

- [113] Nazri Mohd Nawi, Meghana R Ransing, and Rajesh S Ransing. 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*, volume 1, pages 152–157. IEEE, 2006.
- [114] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H Witten. The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18, 2009.
- [115] Jackson Nowotny, Avery Wells, Oluwatosin Oluwadare, Lingfei Xu, Renzhi Cao, Tuan Trieu, Chenfeng He, and Jianlin Cheng. Gmol: An interactive tool for 3d genome structure visualization. *Scientific Reports*, 6:20802, 2016.
- [116] Dong Xu and Yang Zhang. Ab initio protein structure assembly using continuous structure fragments and optimized knowledgebased force field. *Proteins: Structure, Function, and Bioinformatics*, 80(7):1715–1735, 2012.
- [117] Qian Cong, Lisa N Kinch, Jimin Pei, Shuoyong Shi, Vyacheslav N Grishin, Wenlin Li, and Nick V Grishin. An automatic method for casp9 free modeling structure prediction assessment. *Bioinformatics*, 27(24):3371–3378, 2011.
- [118] Mingyang Lu, Athanasios D Dousis, and Jianpeng Ma. Opus-ppsp: an orientation-dependent statistical all-atom potential derived from side-chain packing. *Journal of molecular biology*, 376(1):288–301, 2008.
- [119] Debswapna Bhattacharya and Jianlin Cheng. 3drefine: Consistent protein structure refinement by optimizing hydrogen bonding network and atomi-level energy minimization. *Proteins: Structure, Function, and Bioinformatics*, 81(1):119–131, 2013.

- [120] Marcin Pawlowski, Michal J Gajda, Ryszard Matlak, and Janusz M Bujnicki. Metamqap: a meta-server for the quality assessment of protein models. *BMC bioinformatics*, 9(1):403, 2008.
- [121] C.B. Anfinsen, E. Haber, M. Sela, and FH White Jr. The kinetics of formation of native ribonuclease during oxidation of the reduced polypeptide chain. *Proceedings of the National Academy of Sciences of the United States of America*, 47(9):1309, 1961.
- [122] CA Floudas. Computational methods in protein structure prediction. *Biotechnology and bioengineering*, 97(2):207–213, 2007.
- [123] M. Shah, S. Passovets, D. Kim, K. Ellrott, L. Wang, I. Vokler, P. LoCascio, D. Xu, and Y. Xu. A computational pipeline for protein structure prediction and analysis at genome scale. *Bioinformatics*, 19(15):1985, 2003.
- [124] B.G. Fox, C. Goulding, M.G. Malkowski, L. Stewart, and A. Deacon. Structural genomics: from genes to structures with valuable materials and many questions in between. *Nature methods*, 5(2):129–132, 2008.
- [125] C.M.R. Lemer, M.J. Rooman, and S.J. Wodak. Protein structure prediction by threading methods: evaluation of current techniques. *Proteins: Structure, Function, and Bioinformatics*, 23(3):337–355, 1995.
- [126] J. Moult, J.T. Pedersen, R. Judson, and K. Fidelis. A large-scale experiment to assess protein structure prediction methods. *Proteins: Structure, Function, and Bioinformatics*, 23(3):ii–iv, 1995.
- [127] Yang Zhang. Progress and challenges in protein structure prediction. *Current opinion in structural biology*, 18(3):342–348, 2008.

- [128] James U Bowie, Roland Luthy, and David Eisenberg. A method to identify protein sequences that fold into a known three-dimensional structure. *Science*, 253(5016):164–170, 1991.
- [129] Morten Källberg, Haipeng Wang, Sheng Wang, Jian Peng, Zhiyong Wang, Hui Lu, and Jinbo Xu. Template-based protein structure modeling using the raptorx web server. *Nature protocols*, 7(8):1511–1522, 2012.
- [130] Pascal Benkert, Michael Künzli, and Torsten Schwede. Qmean server for protein model quality estimation. *Nucleic acids research*, page gkp322, 2009.
- [131] D Eisenberg, R Luthy, and JU Bowie. Verify3d: assessment of protein models with three-dimensional profiles. *Methods Enzymol*, 277:396–404, 1997.
- [132] Yang Zhang. Interplay of i-tasser and quark for template-based and ab initio protein structure prediction in casp10. *Proteins: Structure, Function, and Bioinformatics*, 82(S2):175–187, 2014.
- [133] Christopher M Dobson, Andrej ali, and Martin Karplus. Protein folding: a perspective from theory and experiment. *Angewandte Chemie International Edition*, 37(7):868–893, 1998.
- [134] David Shortle, Kim T Simons, and David Baker. Clustering of low-energy conformations near the native structures of small proteins. *Proceedings of the National Academy of Sciences*, 95(19):11158–11162, 1998.
- [135] Hongyi Zhou and Yaoqi Zhou. Distance-scaled, finite ideal-gas reference state improves structure-derived potentials of mean force for structure selection and stability prediction. *Protein Science*, 11(11):2714–2726, 2002.

- [136] Jingfen Zhang, Zhiquan He, Qingguo Wang, Bogdan Barz, Ioan Kosztin, Yi Shang, and Dong Xu. *Prediction of protein tertiary structures using MU-FOLD*, pages 3–13. Springer, 2012.
- [137] Renzhi Cao, Debswapna Bhattacharya, Badri Adhikari, Jilong Li, and Jianlin Cheng. Large-scale model quality assessment for improving protein tertiary structure prediction. *23rd International Conference on Intelligent Systems for Molecular Biology (ISMB), Bioinformatics (accepted)*, 2015.
- [138] Jingfen Zhang and Dong Xu. Fast algorithm for populationbased protein structural model analysis. *Proteomics*, 13(2):221–229, 2013.
- [139] Zheng Wang, Jesse Eickholt, and Jianlin Cheng. Multicom: a multi-level combination approach to protein structure prediction and its assessments in casp8. *Bioinformatics*, 26(7):882–888, 2010.
- [140] David T Jones. Protein secondary structure prediction based on position-specific scoring matrices. *Journal of molecular biology*, 292(2):195–202, 1999.
- [141] Wolfgang Kabsch. A solution for the best rotation to relate two sets of vectors. *Acta Crystallographica Section A: Crystal Physics, Diffraction, Theoretical and General Crystallography*, 32(5):922–923, 1976.
- [142] Zheng Wang, Allison N Tegge, and Jianlin Cheng. Evaluating the absolute quality of a single protein model using structural features and support vector machines. *Proteins: Structure, Function, and Bioinformatics*, 75(3):638–647, 2009.
- [143] Domenico Cozzetto, Andriy Kryshchak, Krzysztof Fidelis, John Moult, Burkhard Rost, and Anna Tramontano. Evaluation of templatebased models in casp8 with standard measures. *Proteins: Structure, Function, and Bioinformatics*, 77(S9):18–28, 2009.

- [144] Yuanpeng J Huang, Binchen Mao, James M Aramini, and Gaetano T Montelione. Assessment of templatebased protein structure predictions in casp10. *Proteins: Structure, Function, and Bioinformatics*, 82(S2):43–56, 2014.
- [145] Andriy Kryshtafovych, Bohdan Monastyrskyy, and Krzysztof Fidelis. Casp prediction center infrastructure and evaluation measures in casp10 and casp roll. *Proteins: Structure, Function, and Bioinformatics*, 82(S2):7–13, 2014.

VITA

Renzhi Cao was born in JiXi county of Anhui Province, China. He received his Bachelor's degree from Anhui Normal University at 2008, and Masters degree from University of Science and Technology of China at 2011. He started his Ph.D studies in the Department of Computer Science at University of Missouri-Columbia at fall of 2011. He is champion winner of Computer Science Annual Programming Contest for all students in University of Missouri-Columbia at 2013 and 2014, and he plays a vital role for MULTICOM which ranks 3rd among all 143 participant groups in the 11th Critical Assessment of Techniques Protein Structure Prediction (CASP11) competition at 2014.

He is interested in developing and applying machine learning, data mining techniques to address the biomedical problems. He worked on protein quality assessment for protein structure prediction when he started his Ph.D studies, and he is also interested in a lot of other bioinformatics problems, such as human genome conformation data analysis, and protein function prediction.