

**A NEW METHODOLOGY
FOR IDENTIFYING INTERFACE RESIDUES
INVOLVED IN BINDING PROTEIN COMPLEXES**

By

Jong Cheol Jeong

Submitted to the Department of Electrical Engineering &
Computer Science and the Graduate Faculty of the University
of Kansas School of Engineering in partial fulfillment of
the requirements for the degree of Master's of Science

Thesis Committee:

Chairperson Dr. Xu-wen Chen

Dr. Luke Huan

Dr. Bo Luo

Date Defended:

The Thesis Committee for Jong Cheol Jeong
certifies that this is the approved version of the following thesis:

**A NEW METHODOLOGY
FOR IDENTIFYING INTERFACE RESIDUES
INVOLVED IN BINDING PROTEIN COMPLEXES**

Chairperson Dr. Xue-wen Chen

Date approved:

Acknowledgement

I would like to gratefully and sincerely thank my advisor, Professor Xue-wen Chen, for his effective guidance and encouragement that he has given me during my graduate studies at University of Kansas. Thanks also go to Dr. Luke Huan and Bo Luo for serving valuable advices as my committee.

I would especially like to thank my parents, Jae Wook Jeong and Sung Im Lee. Without their unconditional love and support for my entire life, nothing has been possible.

Most importantly, I truly give my deepest thanks to my wife Eunmi Kim. During this journey, her support, encouragement, patience and eternal love and devotion are the actual source of my endeavor and endurance. I know it is not enough just say thank to her, but I do know that she understands what I am thinking and what I really want to say to her as she always did.

Finally, I thank my two sons, Jayden Geonu Jeong and Joshua Myeongu Jeong although they may need several more years to read what I have written in this page. They are the source of my happiness and the most glorious and precious gift in my life. They make me laugh all the times.

A portion of this work was supported by the US National Science Foundation Award (IIS-0644366). The opinions, findings, or conclusions in this report are those of the authors and do not necessarily reflect the views of the National Science Foundation.

Abstract

Genome-sequencing projects with advanced technologies have rapidly increased the amount of protein sequences, and demands for identifying protein interaction sites are significantly increased due to its impact on understanding cellular process, biochemical events and drug design studies. However, the capacity of current wet laboratory techniques is not enough to handle the exponentially growing protein sequence data; therefore, sequence based predictive methods identifying protein interaction sites have drawn increasing interest. In this article, a new predictive model which can be valuable as a first approach for guiding experimental methods investigating protein–protein interactions and localizing the specific interface residues is proposed. The proposed method extracts a wide range of features from protein sequences. Random forests framework is newly redesigned to effectively utilize these features and the problems of imbalanced data classification commonly encountered in binding site predictions. The method is evaluated with 2,829 interface residues and 24,616 non-interface residues extracted from 99 polypeptide chains in the Protein Data Bank. The experimental results show that the proposed method performs significantly better than two other conventional predictive methods and can reliably predict residues involved in protein interaction sites. As blind tests, the proposed method predicts interaction sites and constructs three protein complexes: the DnaK molecular chaperone system, 1YUW and 1DKG, which provide new insight into the sequence–function relationship. Finally, the robustness of the proposed method is assessed by evaluating the performances obtained from four different ensemble methods.

Keywords: Protein-protein interactions, Protein Binding, Interface Residues, Machine Learning, Computational Biology, Random Forests, Protein Sequence Analysis, Properties of Amino Acids

TABLE OF CONTENTS

CHAPTER 1. INTRODUCTION.....	1
1.1 Motivation.....	1
1.2 Studies on protein-protein interactions.....	3
1.2.1 <i>Identifying protein-protein interactions</i>	4
1.2.2 <i>Defining domain-domain interactions</i>	7
1.2.3 <i>Identifying protein-protein interaction sites</i>	10
1.3 My Contributions.....	14
CHAPTER 2. RESEARCH BACKGROUND	17
2.1 Protein Data Bank (PDB).....	18
2.2 Dictionary of Protein Secondary Structure (DSSP).....	19
2.3 Features derived from amino acid sequences.....	22
2.4 Classifiers.....	31
2.5 Conventional Methods.....	36
CHAPTER 3. METHOD AND RESULTS.....	37
3.1 Methodology.....	37
3.1.1 <i>Extracting wide range features</i>	38
3.1.2 <i>Constructing an integrative method</i>	42
3.2 Experiments and Results.....	45
3.2.1 <i>Data sources</i>	45
3.2.2 <i>Evaluation criteria</i>	48
3.2.3 <i>Leave-One-Out Cross Validation (LOOCV)</i>	50

3.2.4	<i>Blind test</i>	67
3.2.5	<i>Further Evaluation on Random Forest Framework</i>	75
CHAPTER 4. CONCLUSION		88
4.1	Summary of Research	88
4.2	Future Work	91
REFERENCES		93

LIST OF FIGURES

Figure 2.1 Contents of PDB	18
Figure 2.2 Contents of DSSP	19
Figure 2.3 Position Specific Score Matrix (PSSM)	22
Figure 2.4 HSSP	25
Figure 2.5 Flow char of calculating HSSP	26
Figure 2.6 Sequence patch	36
Figure 2.7 Spatial patch	36
Figure 3.1 Schematic diagram of sliding window with size 10	38
Figure 3.2 Creating distance vector	40
Figure 3.3 Evaluation on Antibody-antigen.....	51
Figure 3.4 Evaluation on Enzyme.....	51
Figure 3.5 Evaluation on G-proteins.....	51
Figure 3.6 Evaluation on Protease-inhibitor	51
Figure 3.7 Evaluation on Large-prot.....	51
Figure 3.8 Evaluation on Miscellaneous.....	51
Figure 3.9 The ROC curves for three sequence-based predictors	53
Figure 3.10 Comparing accuracies based on the best balanced accuracy.....	55
Figure 3.11 Comparing accuracies based on the best overall accuracy.....	57
Figure 3.12 Predicted interface residues in 1IAI	58
Figure 3.13 Predicted results of a chain in 1GG2	59
Figure 3.14 Visualization of residues identified by three methods	62

Figure 3.15 Visualization of residues identified by three methods	63
Figure 3.16 Visualization of residues identified by three methods	64
Figure 3.17 Predicted results of chains in 2CIO	65
Figure 3.18 Visualization of residues in independent data identified by three methods	66
Figure 3.19 DnaK chaperone system	68
Figure 3.20 DnaK molecular chaperone system	70
Figure 3.21 Predicted results of 1YUW using the proposed method	72
Figure 3.22 Predicted results of three chains in 1DKG using proposed method	74
Figure 3.23 Comparing AUC on random forests	75
Figure 3.24 Comparing the offset of decision boundary	77
Figure 3.25 An example of Classification Pattern Matrix (CPM)	82
Figure 3.26 Comparing ROC among different ensemble methods	86
Figure 3.27 Comparing AUC among four different ensemble methods	87

LIST OF TABLES

Table 2.1 Defining interface residues	20
Table 2.2 Norminal Maximum Area.....	21
Table 2.3 The scale of physicochemical properties	29
Table 3.1 Categories of proteins and corresponding PDB IDs.....	46
Table 3.2 Statistics of the dataset.....	47
Table 3.3 Criterion functions	49
Table 3.4 Performances at the best balanced accuracy	55
Table 3.5 Performances at the best overall accuracy	56

Chapter 1. Introduction

1.1 Motivation

Identifying protein-protein interaction (PPI) pairs is one of the most challenging problem and its importance is getting increased in proteomics since protein functions can be characterized by PPIs and give key roles for revealing evolutionary lineages although interactions among proteins are astonishingly diverse and highly complex by means of analyzing patterns and principles governing these interactions [1-3].

Indeed, PPIs are strongly connected to developing new drugs by helping to identify or validate drug targets [4-5]. It is more interesting when one step moves further into the area of identifying protein binding sites. Compared to identifying PPIs identifying binding sites in PPIs can be directly used for treating diseases. For example, identifying binding sites can treat diabetic complications. Protein Kinase C (PKC) is implicated in the pathology of diabetic neuropathy and activated by increased levels of glucose which cause an increase in intracellular diacylglycerol(DAG). The production of DAG in membrane facilitates translocation of PKC from the cytosol to the plasma membrane and affects regulating neurite outgrowth and insulin resistance. An actin-binding site between C1 domains of PKC is important to mediate neurite outgrowth therefore by controlling the activation of various PKC isoforms can be considered as a treatment of diabetes [6-8].

Another example can be found in developing bioactive peptide drugs which are made with specific protein fragments and designed for interfering PPIs by enhancing the ability of interaction with target proteins to give a positive impact on body functions or conditions. The case of developing antiplatelet drugs[9] would be good example for this type of drug. Typical

antiplatelet drugs target to prevent or treat disease related to cardiovascular system such as Atherosclerosis or arteriosclerotic vascular disease (ASVD) which is caused by an artery wall thickens as the result of a build-up of fatty materials like cholesterol, Myocardial Infarction (MI) or Acute Myocardial Infarction (AMI) which is commonly known as a heart attack and caused by the interruption of blood supply to a part of the heart due to the blockage of a coronary artery and resulting in heart cells to die, and Stroke or Cerebrovascular Accident (CVA) which is the rapidly developing loss of brain functions due to disturbance in the blood supply to the brain due to the lack of blood flow caused by blood clot to an artery.

Above two examples might be good enough to get up an appetite about the study of identifying binding sites. However, current technologies to identify binding sites which will be explored in following chapters are often time consuming and have high computational complexity due to requiring wet laboratory work and handling three dimensional structures. Fortunately, current genome-sequencing projects show that the pool of protein sequences are rapidly increasing and make easy to access huge amount of protein sequences; therefore, there is a growing demand for developing advanced computational methods for predicting potential protein binding sites directly from protein sequences. Few studies have been conducted to predict interface residues by using amino acid sequences, but they are not enough to make reliable predictions yet and still remained in infancy. In this article, new methodology for reliably predicting protein interface residues involved in folding protein complexes is demonstrated. The proposed method is based on the analysis of protein sequence information derived from a wide variety of physicochemical properties and sequence profiles. We believe that the proposed method can help to reduce the burden of current technology and open new paradigm of identifying interface residues on protein-protein interactions.

1.2 Studies on protein-protein interactions

Studies to understand protein-protein interactions (PPIs) have drawn increasing interest since most cellular processes and biochemical events are triggered by interactions of proteins; therefore, understanding PPIs is a good starting point for revealing biological processes in entire cells such as elucidating cellular functions, the mechanisms of forming protein complexes, chemical reactions, and understanding signal transduction networks. Recent studies also show that understanding PPIs can help identifying pharmacological targets and rational drug design studies [9-12].

There are two main streams for discovering PPIs: experimental methods and *in silico* methods. Experimental methods simply mean that PPIs are directly discovered by biological experiments in wet laboratory through affinity chromatography[13], copurification[13], cross-linking[13], coimmunoprecipitation[5-7], yeast-two-hybrid[5-8], mRNA expression profile[14-15].

Compared to experimental methods, *in silico* methods often use the results of experimental methods as their principle sources. However, the results of *in silico* also can be feedback to the experimental methods as a starting point of forming hypotheses in biological experiments by validating functional hypotheses via design of restricted fragments for two-hybrid assays or specific mutagenesis [1, 5, 11].

Roughly speaking, *in silico* methods are categorized into three different groups: identifying protein-protein interaction, domain-domain interaction, and protein-protein interaction sites. Because their aims are similar as understanding cellular functions and processes, some of their principles are often overlapped but their approaches and specific aims are different in many cases.

To clarify the purpose of this study, three different categories of *in silico* methods are briefly reviewed and compared in this section.

1.2.1 Identifying protein-protein interactions

Identifying protein-protein interaction (PPI) is categorized in here as a type of the *in silico* methods and is mainly focused on pairs of proteins whether they are interact or not. High-throughput methods, yeast-two-hybrid (Y2H) and mRNA profiling are widely used techniques to verify PPIs. However, the reliability of the high throughput methods is still questionable due to producing many false positives [16-19] and each experimental assay can identify only a subset of PPIs; therefore, many efforts on *in silico* methods have devoted to improve the accuracy of experimental assays and complete PPI networks by discovering PPIs that have not been accessible to experimental methods[20]. These efforts can be categorized into two groups, genome and proteom based approaches.

Genome based approach

Genomic information is the most fundamental source of investigating cellular processes and inferring protein functions. It is well known that interacting proteins tend to co-evolve [21-24]. Several studies have been conducted on this rationale by comparing phylogenetic profiles that describe the pattern of the presence or absence of a given genes across the related organisms, so that similar profiles between proteins are likely to be functionally related each other [21-22, 25].

In the same manner, correlated gene neighbors are often used for discovering functional linkage of proteins such that two genes are found to be neighbors in different organisms then they are likely to encode functionally related proteins and tend to physically interact [26-27]. In a similar passion of the correlated gene neighbors, the codon compositions can be used for

predicting PPIs such that similar patterns of codons appeared in protein pairs tend to likely interact. Najafabadi *et al.* [28] Jansen *et al.* [29] introduced a PPI prediction method by using Naïve Bayesian Network upon the likelihood ratios between interact and non-interact codons, and co-expression and co-localization profiles.

Gene fusion or Rosetta stone method is a simple and powerful PPI inference method. The principle is that if genes are fused in an ancestor genome, in other words, the fused gene is homologous to two separate genes in another species then two proteins likely interact [19, 21].

Although it is true that genome based method often results in high accuracy on predicting PPI pairs and give precise analysis of PPIs, in order to achieve unbiased results, this approach has to be conducted on organisms with complete genomes[24]. Therefore, this type of methods should be careful on the biased results caused by species with incomplete genomes. The other skepticism of using genomic data is that some organisms make such incredible versatile use of few genes to produce protein. For example, human has about 90,000 different proteins and 25,000 genes [30]. The huge difference between the number of proteins and genomes shows that humans are very sophisticate using genes to produce actual proteins. In the other side, this makes problems more complicated to reproduce and/or explore the nature of protein-protein interactions since proteins are much more dynamic than the gene such that proteins are often changed during their development and interactions by regulating and supporting each other in response to external stimuli [16].

Proteom based approach

Because of such limitations on genomic information and versatile usage of genes in higher organism, many studies have been conducted on directly tackling PPI itself by using protein sequences and integrating databases [31-35].

However, both theoretically and physically, genes and proteins are closely related each other; therefore, the rationales used in genome based approaches are often translated into protein based approaches and visa versa.

For example, co-evolution theory used in a genome based approach is interpreted as measuring correlated mutations between interacting proteins. The difference between two methods are that the genome based approach analyzed co-evolution rate by investigating the presence or absence of genes in related organism based on phylogenetic tree [17-18, 21], but proteom based approaches analyze the correlated mutations by calculating multiple sequence alignments. This means that the proteom based approaches consider the interacting proteins undergo a process of co-evolution, so measuring similarity between two residues at the aligned positions can reveal the co-evolution tendency of interacting proteins [31, 36]. One of the issues found in conventional methods of co-evolutionary theory is that although the majority of proteins in nature is composed with multiple domains, conventional methods [31, 36] only validated this approach upon those proteins having only two domains; therefore, based on the conventional methods, the reliability of correlated mutations on proteins with multiple domain pairs is still questionable [37-38].

Several papers show that investigating physicochemical properties like hydrophobicity and hydrophilicity in protein sequences can be used as good predictors of PPIs [29-30, 35-38] but these studies focused on specific proteins or particular molecular system, so their general performances are not yet validated.

One of the papers that describes the usage of proteome databases for identifying PPIs is published by Sprinzak et al.[35]. In this paper, they introduced sequence-signatures extracted from InterPro database [39] as an indicator of identifying significant sequence-signature pairs for

protein interactions by calculating log-odd value between the observed frequency of a sequence signature pair and background frequency of a sequence signature pair.

Although many efforts on defining PPIs have been conducted, the information getting from this type of methods are very limited and not enough to understand the nature of biomolecular interactions and cellular processes since this type of study only provides a set of proteins that directly interact with their partners. In order to apply the information of PPIs into advanced technologies like designing new drugs and tracing transduction networks which requires the specific binding sites and/or functional relationships; therefore, such methods that gives more precise and sophisticated information about PPIs are required [38, 40].

1.2.2 Defining domain-domain interactions

The methods grouped into this category can potentially solve the problems remained in previous section; identifying the sites or locations of protein bindings. The principles of domain-domain interactions (DDIs) are that certain domains, which are critical to recognize molecules, are the key factors of forming protein complexes and defining protein functions; therefore, understanding interacting domains can identify potential PPI sites and protein functions simultaneously [1, 21, 41-42]. The effects from identifying DDIs are very significant; nevertheless a few methods have been introduced.

The simplest method for defining DDIs is calculating the probability of DDI pairs based on the frequency of DDIs appeared in PPI databases such that the higher frequency appearing a domain pairs in PPI databases the higher probability of interaction domain pairs [41-42].

More sophisticated design of domain interactions is proposed by Deng *et al* and is based on the probability theory. [43]. In their paper the Maximum Likelihood Estimation (MLE) algorithm was used for predicting interact domains and optimized the probabilities of domain-

domain interactions by using evolutionarily conserved domains defined in a protein-domain database such as Pfam [44]. The major deficiency in this method is that the low-propensity and high-specificity domain interactions may not be detected because the probability of DDI is based on the frequency of co-occurrence in interacting protein pairs. In fact, some domains need fidelity in cellular circuitry for interacting protein pairs, so their characteristics of binding sophisticated domain pairs are resulted in low appearance; therefore, this kind of specific domain pairs cannot be detected by simple MLE method even though the domains of binding tendencies are clear in the nature [45-46]. Domain Pair Exclusion Analysis (DPEA) improved the identification of low-propensity and high-specificity domain pairs by introducing E-score [45]. E-score is calculated by Expectation and Maximization (EM) algorithm. The first step of DEPA is similar to the MLE in Deng et al[43] such that DPEA estimates propensity of interacting domain pairs from PPI networks. At the second step, DEPA excludes the high propensity of interacting domain pairs estimated by the first step with a threshold, E-score. The final E-score is calculated by the log-odd ratios in which the numerator is the probability of an interacting protein pair given that two domains interact and the denominator is the probability of an interacting protein pair given that two domains do not interact. In other words, the E-score measures the evidence of two domains ever interact [45].

Upon the studies of DDIs, we have seen that this type of study highly depends on PPI networks; therefore, acquiring accurate PPI network is prerequisite to derive reliable results on identifying DDIs. Studies show that many PPI networks are based on Yeast-Two-Hybrid (Y2H) method and this method often contains noises especially the high number of false positives [16-19]. Due to the importance of PPI networks, many studies have been conducted for reducing the noise effects. Message passing method is based on the belief propagation algorithm and

introduced by minimizing false positive on DDI networks[40], and Guimaraes et al. [47] introduced a parsimony approach in which they reformulated the problem of predicting DDIs as an optimization problem such that the objective is to minimize false positives by minimizing the number of DDIs necessary to justify the underlying PPI networks. In other words, optimal PPIs are defined by justifying the minimum number of DDIs in interacting protein pairs.

Some of the principles used in defining PPIs are also applied into defining DDIs. For example, Gene fusion or Rosetta stone method [19, 21] introduced in defining PPIs are translated to identifying DDIs. In the scope of identifying DDIs, genes are interpreted as domains in proteins such that gene fusions are translated as domain fusions in proteins; therefore, the fused two domains are considered as homology if isolated forms of these domains are found in another species, so these two domains are considered as an interacting domain pair [42, 48-50]. Phylogenetic profile [18] and correlated mutation [31] based methods used in defining PPIs are also applied into analyzing DDIs. Jothi et al. [38] proposed co-evolutionary analysis of interacting domain pairs. The rationale of this method is that interacting protein pairs are likely to co-evolve in nature, so the changes on the binding surface of one protein can affect the interface of the other interacting partner protein. They were investigating the relative degrees of co-evolutionary domains of interacting proteins. The relative degrees of co-evolutionary domains were defined as a correlation coefficient of similarity matrix which was calculated from multiple sequence alignment of domains extracted from orthologs of interacting proteins.

Although conventional methods identifying DDIs can help to look closer into the nature of interacting proteins, there are still many questions that conventional methods cannot answer: *What makes proteins interact? What is the most important factor causing PPIs?* and *What residues in a protein pairs are directly involved in PPI?* Indeed, most conventional methods [35,

38, 43, 45, 47] assume that DDIs are independent and identically-distributed (IID), but studies show that in practice, PPIs are often affected by multiple environmental factors such as shape and electrostatic complementarity, hydrogen bond, temperature, acidity and basicity [2, 12, 35-36, 53-59]. Due to the over simplified hypothesis of the model, conventional methods based on IID assumptions have inevitable problems on PPI identification: conflicted results of same structure but different functions and/or different interactions [35, 42, 60-64]; therefore, there is an increasing interest for developing new methods that can effectively reflect multiple factors of protein bindings to PPI identifications.

1.2.3 Identifying protein-protein interaction sites

For the next step after discovering PPI pairs and DDIs, identifying specific interaction sites on PPI and/or DDI pairs are often considered as interesting tasks in advanced technologies like pharmacology [9-12] since previously discussed methods for identifying PPI and DDI pairs only tells the possible interactions and/or broad definitions of binding sites. In other words, conventional methods for identifying PPIs and DDIs are not enough to reveal the nature of interactions in depth; therefore, a new approach to get more specific information about binding proteins including specific interaction sites between PPI pairs is a new and interesting challenge. With increasing demand of developing such methods, several methods have been proposed in different aspects. Roughly speaking, the efforts for identifying PPI sites can be categorized into two main streams: *structure based methods* using known protein structure information and *amino acid sequence-based method* using all possible information of proteins except structure information [1, 65]. In this section, different categories are compared and discussed together with brief introductions of each method.

Structure based method

The advantage of using protein structure information on identifying protein-protein interaction sites is that the most enriched information upon the current technology can be used for exploring the nature of protein complexes, so this makes possible to analyze protein interactions with various approaches originated from different fields of studies. For convenience of explanation, structure-based methods are grouped into three different categories: *geometric*, *energetic* and *machine learning-based* approaches.

The most distinct difference between structure based methods and others is the property of visualization. This means that their geometric properties can be visualized and used for discovering PPI sites by using complementarities of shape and size, close packing, and the absence of steric hindrance between potentially interacting protein pairs. This *geometric-based method* is actively used in docking methods such that the geometric properties are used for the fitness function defining the best match between potential proteins pairs [66-72]. In a figurative sense, the geometric or docking method is identifying interacting protein pairs by playing mosaic puzzles composed of proteins having various geometric characteristics.

Energetic methods are more focused on atomic level of protein structure rather than the entire structure of protein, so they search energetically stable protein complexes [51-52]. The stable complexes are derived from calculating energies within atoms of a protein itself and/or between atoms of potential interacting protein pairs such as the electrostatic potential energy, Coulombic field, van der Waals' interaction, and total interaction energy. However, calculating energy on atomic level is time intensive process, so energetic approaches often integrate geometric properties and other protein databases to reduce the processing time by narrowing down the possible interaction pairs and/or interface sites [53-57].

Geometric and energetic methods often require high resolution protein structure to get promising results, and both of them are time-consuming and require high computational complexity due to the exploration of three dimensional protein space or coordinates; therefore, different ways of using structure information are demanded. Several groups have proposed *machine learning-based* approaches for identifying PPI sites [58-62]. Compared to geometric and energetic methods, machine learning methods often use summarized structure information such as solvent accessibility of residues and geometric locations of residues. This information is used for identifying surface residues, and the identified surface residues are used for narrowing down the size of potential interfaces on protein complexes since interior or buried residues are rarely involved in PPIs. Once surface residues are identified, patches are defined by grouping neighboring surface residues. At the final step, features of interacting sites are extracted from patches by investigating the group of amino acid residues on biochemical, genetic, and theoretical properties; therefore, the derived features of protein complexes are applied into machine learning algorithms as training dataset.

Amino acid sequence based method

Although there have been significant efforts to analyze protein structures, time consuming and expensive experimental technologies limit the population of known protein structures. Compared to protein structures, protein sequences requiring relatively lower cost on their discovery lead the significant difference between the number of known protein structures and sequences. By October 23, 2011, there are 532,792 manually annotated protein sequences in Uniprot/SwissProt [63] and only 76,669 known protein structures in PDB [64].

Enriched protein sequences together with tools and databases for analyzing protein sequences inspire identifying protein interaction sites directly from amino acid sequences. Compared to the machine learning-based method using structure information, the limited-information about PPI pairs makes problem difficult, especially on deriving features for characterizing interface residues and others. However, studies showed possibilities that PPI sites can be identified directly from protein sequences, and the related studies were introduced below.

Chothia and Janin [65-66] showed that non-polar residues are dominant to the contribution of interface area, and the hydrophobic free energies are correlated with the interface areas. Argos [67] and Janin *et al.* [68] had similar results such that hydrophobic residues were enriched on interface areas, and protein molecular weights were correlated with accessible surface area. Jones and Thornton [69-71] analyzed multiple categories of protein complexes based on surface patches which were the group of the geometrically closest neighboring surface residues. The results showed that the hydrophobic residues had a greater preference for the interfaces of homodimers than for those of heterocomplexes. Kini *et al.* [9, 72-73] examined 1,600 PPIs and found that proline was the most commonly appeared residue by locating one or two residues away from interaction sites.

There are many evidences of possibility to identify PPI sites directly from protein sequences; nevertheless, several methods have been proposed. Eisenberg *et al.* [74-75] characterized membrane α -helix proteins from soluble proteins by plotting mean hydrophobic moment versus the mean hydrophobicity. In their work, the mean hydrophobicity was defined as the average of all of the hydrophobicities of the amino acids in the helix, and the mean hydrophobic moment is a measure of the amphiphilicity of the helix. De Loof *et al.* extended Eisenberg's method for predicting the receptor binding domains in apolipoprotein E and in the

low density lipoprotein apolipoprotein B-E receptor [76]. The method is further redefined by Gallet *et al.* by analyzing hydrophobicity distribution and amino acid frequencies in known interaction sites for identifying protein-DNA and protein-calcium ions interacting sites based on linear stretches of amino acid sequences [1]. Most recently, more sophisticated machine learning approaches are applied to predict interaction sites. Yan *et al.* [77] employed support vector machines (SVMs) as classifiers and features of individual target residues are extracted from a group of amino acid neighbors in a sequence. Wang *et al.* [78] also applied SVMs with features from spatial sequence and evolutionary conservation scores based on a phylogenetic tree.

As a result, although many researches have proved that the sequence based method is highly potential to efficiently characterize interaction sites from others, the current sequence based method is still in its infancy, in terms of both the accuracy and the usage of sequence features. These deficiencies encouraged the study of identifying interaction sites by using amino acid sequence information only.

1.3 My Contributions

There are three main reasons why such small number of methods compared to other methods has been proposed on sequence-based identifying interface sites although many studies showed that the amino acid sequence itself only has high potential to be a good indicator of defining interaction sites: (i) the biological properties that are responsible for PPIs are not fully understood, so this yields the difficulty of extracting informative features that are common to all interacting sites, (ii) there is no generally available systematic approaches to convert experimentally proven informative factors into computationally preferable data representations such that there are several features which are responsible for certain levels of PPIs like hydrophobicity and proline appearance, but due to the diversity of the length of amino acid

sequences, it is not easy to represent and integrate these commonly known features into computationally preferable data format, and (iii) more precisely the number of interface residues in a protein is much smaller than that of non-interacting residues, which leads to a very challenging problem, the so-called imbalanced data classification problem in the view of machine learning.

Through this article, the solutions of three major problems described above in identifying interface residues in PPIs are proposed. The proposed method is based on machine learning algorithm, especially focused on random forests [79-80] paradigm. The solution of first problem, the lack of informative features is investigated through integration of well known features using an ensemble method. The basic assumption on this approach is that the combinations of partially effective features together can generate effective rules for general problems. This assumption is widely used in machine learning society [98-101]; therefore, through this article the way to effectively integrating previously known sequence-based features are introduced and evaluated.

Next, the systematic approaches building computationally preferable data formats are suggested as a solution of the second problem. The proposed solutions can be varied upon the property of features. In this article three different types of features are discussed: features from physicochemical properties, geometric features, and evolutionary profiles. The proposed solutions show that under these three different features, almost all amino acid sequence-based features can be formatted toward computationally favorable dataset.

The solution of the third problem, the issue of imbalanced dataset is investigated through reinterpreting the principle of random forests: producing many biased trees with random sample and features and minimizing the bias to get optimal prediction by assembling trees in a certain way. The proposed method generalizes this principle by learning models with controlled

sampling mechanism. The results showed that proposed method could derive robust models on highly imbalanced dataset.

In summary, the thesis contributes the area of identifying PPI sites in four ways. First, the effective way to integrate features is proposed and the proposal can change the direction of PPI site identification by using a group of features instead of finding a unique global feature.

Second, the systematic approaches converting experimentally known factors into computationally preferable data format are guided. With the guidelines, almost all of features derived from amino acid sequence properties can be transformed to machine learning preferable datasets by considering current Amino Acid Index [81] containing 544 amino acid properties.

Third, the effective way to handle imbalanced dataset is proposed by generalizing the principles of random forests.

Finally, this thesis shows that the proposed method outperforms conventional protein-protein interaction site prediction methods.

Chapter 2. Research Background

Machine learning-based protein-protein interface residue prediction method is proposed. The proposed method is composed of several steps including integration of physicochemical properties of amino acids, proteome databases and software for identifying true class of interface residues among protein-protein interactions. Due to the integration of multiple sources, this chapter reviews each feature, database and software required in the proposed method. The reviews are described through three subsections: protein data bank (PDB), dictionary of protein secondary structure (DSSP) and features derived from amino acid sequences.

To help understanding the comparisons between the conventional and proposed method in Chapter3, two machine learning-based conventional methods are introduced and related technical issues and background studies are also discussed in this chapter.

2.1 Protein Data Bank (PDB)

Protein Database Bank (PDB) [64] is the most well known protein structure database and publicly available through <http://www.pdb.org>. This database provides the coordinates of atoms which are elements of individual proteins determined by crystallographic processes like nuclear magnetic resonance (NMR) and X-ray crystal structure determination; therefore this information makes possible to reveal the actual binding sites or/and calculate interface residues from protein complexes by calibrating these coordinates.

In the proposed method, PDB is used for identifying the locations of individual residues whether they are located at the surface or inside of protein. To do this, the coordinates of individual chains are split or merged depends on the sequence similarity, and then files containing the coordinates of atoms belonging to the selected chains are used as input files of DSSP which is describe in next section.

Amino acid				coordinate			Temperature Factor			
Atom				X	Y	Z				
ATOM	1	N	ALA A	2	-0.467	15.970	2.964	1.00	9.41	N
ATOM	2	CA	ALA A	2	-0.551	16.338	1.545	1.00	9.71	C
ATOM	3	C	ALA A	2	-0.460	15.063	0.726	1.00	9.00	C
ATOM	4	O	ALA A	2	0.575	14.778	0.135	1.00	8.92	O
ATOM	5	CB	ALA A	2	0.529	17.358	1.168	1.00	10.05	C
ATOM	6	H1	ALA A	2	-1.165	15.264	3.181	1.00	9.76	H
ATOM	7	H2	ALA A	2		15.594	3.212	1.00	8.93	H
ATOM	8	H3	ALA A	2		16.791	3.534	1.00	9.77	H
ATOM	9	HA	ALA A	2	-1.523	16.792	1.352	1.00	9.77	H
ATOM	10	HB1	ALA A	2		16.945	1.353	1.00	9.77	H
ATOM	11	HB2	ALA A	2	0.441	17.616	0.113	1.00	10.93	H
ATOM	12	HB3	ALA A	2	0.404	18.263	1.763	1.00	10.16	H
ATOM	13	N	LYS A	3	-1.555	14.299	0.706	1.00	9.02	N
ATOM	14	CA	LYS A	3	-1.545	12.880	0.379	1.00	8.43	C
ATOM	15	C	LYS A	3	-0.764	12.126	1.465	1.00	6.54	C
ATOM	16	O	LYS A	3	-0.047	12.723	2.272	1.00	5.78	O
ATOM	17	CB	LYS A	3	-0.994	12.569	-1.017	1.00	9.05	C
ATOM	18	CG	LYS A	3	-1.407	13.513	-2.151	1.00	10.58	C

Figure 2.1 Contents of PDB

This figure shows the contents of an actual PDB file describing atom information. The meaning of each column is denoted and pointed by using arrows

2.2 Dictionary of Protein Secondary Structure (DSSP)

Dictionary of Protein Secondary Structure (DSSP) [82] is designed for making standard secondary structure alignment of all entries in the PDB, and it is publicly available through <http://swift.cmbi.ru.nl/gv/dssp/>. Some important features of DSSP are shown below.

In figure 2.2, residue ID, chain ID, and residue name are corresponding to those of PDB. DSSP assigns a secondary structure of each residue from seven different states: H, B, E, G, I, T, and S which are alpha helix, residue in isolated beta-bridge, extended strand participating in beta ladder, 3-helix, 5-helix, H-bonded turn, and bend respectively. BP1 and BP2 denote residue ID of first and second bridge partner. ACC shows the number of water molecules in contact with each residue and is often used for defining surface or/and interface residues in protein complexes [36-37, 94-95, 104-105]. Hydrogen bond denotes the states of hydrogen bond together with its energy.

#	RESIDUE	AA	STRUCTURE	BP1	BP2	ACC	Hydrogen bond			
							N-H-->O	O-->H-N	N-H-->O	O-->H-N
1	B	G		0	0	114	0, 0.0	2, -2.4	0, 0.0	0, 0.0
2	B	P	+	0	0	108	0, 0.0	2, -0.2	0, 0.0	0, 0.0
3	B	R	+	0	0	239	-2, -2.4	0, 0.0	1, -0.0	0, 0.0
4	B	L	+	0	0	179	1, -0.2	2, -2.0	-2, -0.2	4, -0.5
5	B	S	+	0	0	79	1, -0.2	-1, -0.2	2, -0.1	4, -0.0
6	B	R	S >	0	0	257	-2, -2.0	3, -1.8	2, -0.1	-1, -0.2
7	B	L	G >	0	0	87	-3, -0.5	3, -2.4	1, -0.3	-2, -0.1
8	B	L	G >	0	0	95	-4, -0.5	3, -1.7	1, -0.3	-1, -0.3
9	B	S	G <	0	0	111	-3, -1.8	-1, -0.3	1, -0.3	-2, -0.2
10	B	Y	G <	0	0	164	-3, -2.4	-1, -0.3	-4, -0.1	-2, -0.2
11	B	A	<	0	0	131	-3, -1.7	-2, -0.2	-4, -0.0	-3, -0.1

Figure 2.2 Contents of DSSP

This figure shows the actual DSSP output and the meaning of each column is denoted and pointed by arrows.

In fact, ACC values are commonly used for defining true class labels whether a residue is interface or not [36-37, 94-95, 104-105]. The input of DSSP program is PDB files, so to get expected results, the manipulations of PDB file is required. The common procedure to identify interface residues is shown below.

Table 2.1 Defining interface residues

Step 1. Calculating Solvent Accessible Area (ASA) using DSSP for both unbound molecule (MASA) and complexes (CMASA)

Step 2. Calculating relative ASA (RASA)

$$RASA = \frac{MASA}{\text{Norminal Maximum Area}} \times 100$$

Step 3. Interface residue is satisfying both conditions:

- i. $RASA \geq 25\%$
 - ii. $MASA - CMASA \geq 1\text{\AA}^2$
-

The value of MASA and CMASA is obtained by manipulating PDB files such that each target chain or group of target chains in a PDB file is stored separately and then these files are run on DSSP. At the output of DSSP, the values in ACC column are used as MASA and CMASA. For the experiment, measuring strict sequence alignment among separated chains is conducted, and sequences with less than 30% sequence similarity are selected to avoid bias effects occurred by homologous sequences. RASA also can be used for defining surface and interior residues, and the norminal maximum area (NMA) used for the experiments were retrieved from the study of Rost *et al.* [83] and shown table 2.2.

Table 2.2 Norminal Maximum Area

AA	A	B	C	D	E	F	G	H	I	K	L	M
NMA	106	160	135	163	194	197	84	184	169	205	164	188
AA	N	P	Q	R	S	T	V	W	X	Y	Z	
NMA	157	136	198	248	130	142	142	227	180	222	196	

As a difference from surface residues, defining interface residues require two conditions: (i) interface residues must be a subset of surface residues and (ii) in the complex form the interface residues must be buried with a certain rate. These conditions are very intuitional since interface residues contact to other molecules, so in order to make contact, the residues must be located in the surface of molecules. The meaning of 25% of RASA is that in order to be a surface residue, its solvent accessibility which is the measure of contacting water molecules has to remain greater than 25% of its maximum solvent accessibility, so these areas are considered to be potential binding sites. Once molecules bound each other then the contact residues are prevented for contacting water molecular. In other words, the residues are buried by contacting molecules; therefore the solvent accessibility of each residue between MASA and CMASA must be different.

2.3 Features derived from amino acid sequences

Position Specific Score Matrix (PSSM)

Position specific score matrix (PSSM) was originally designed for identifying distantly related proteins by using a group of sequences previously aligned by structural or sequence similarity [84]. PSI-BLAST [85] is the most commonly used application to detect remotely related homologous proteins or DNA by using PSSM profiles. Although the formation of PSSM can be varied on the purpose of applications, the principles are very much the same.

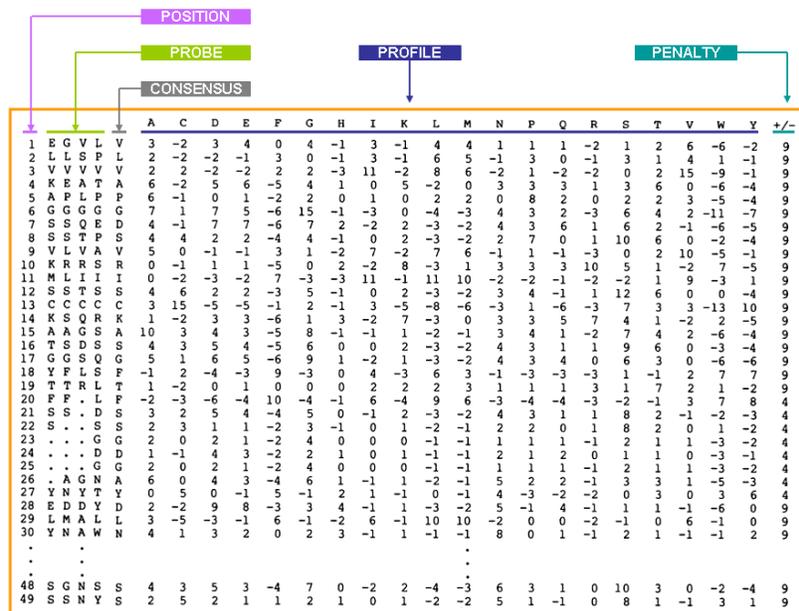


Figure 2.3 Position Specific Score Matrix (PSSM)
This figure shows PSSM profile for an immunoglobulin variable-region

Figure 2.3 shows the original PSSM introduced by Gribskov et al. [84]. The PSSM profile consisted of five sections: *Position* indicates the index of each amino acid residues which is defined by the results of multiple sequence alignment by sequentially increasing the index of amino acids including gaps. *Probe* is a group of sequences identified as functionally related proteins based on sequence or structural similarity. *Consensus* is a sequence of amino acid

residues which are the most conserved residues at each position among the aligned probes by the means of mutation of each residue at that position. It is different from the common appearance of amino acid residue at each position such that a consensus residue at each position is defined by selecting a residue belonging to the highest score in the profile column. *Profile* is composed of 20 columns which correspond to 20 amino acids and N rows which is the length of the multiple alignments of probes. The 20 columns of each row specify the conservation score for each residue finding at that position. The details of calculating the score are shown below.

The score of an amino acid a corresponding to one of the columns in profile section at the position p which is the index of amino acid at first column in PSSM profile, $M(p,a)$ is defined by the summation of the ratio between the frequency of appearing any of 20 amino acid at position p and number of probes multiplied by the value of Dayhoff's mutation matrix [86] between amino acid a and all 20 amino acids.

$$\mathbf{M}(p,a) = \sum_{i=1}^{20} \mathbf{w}(p,i) \times \mathbf{Y}(a,i)$$

here,

$$\mathbf{w}(p,i) = \frac{\text{the frequency of appearing amino acid } i \text{ at position } p}{\text{the number of probes}} \quad (2.1)$$

$\mathbf{Y}(a,i)$ = the value of Dayhoff's mutation matrix between amino acid a and i

At the last column, *Penalty* specifies the position-dependent penalties for insertion and deletions of the corresponding probe residues because insertions and deletions in families of aligned homologous sequences are more likely to be appeared in regions between segments of regular secondary structure than within them. Later, this penalty is used for profile-sequence alignment for specifying opening and gap extension penalties. The new penalty is shown below.

$$\begin{aligned}
 PEN &= P_{score} \times (OPEN + EXN \times L) \\
 P_{score} &= \frac{YMAX}{OPEN + EXN + LMAX(p)}
 \end{aligned}
 \tag{2.2}$$

Here, PEN is the new penalty for profile-sequence alignment, $OPEN$ and EXN are opening and gap extension penalties respectively, P_{score} is the penalty score appeared in PSSM, $YMAX$ is the highest score in Dayhoff's matrix, and $LMAX(p)$ is the longest gap in the probe that includes position p . Although PSSM is a strong feature to identify the relationships of a sequence to others, the variant length of each sequence makes difficult to directly import PSSM into machine learning favorable dataset. The techniques of importing PSSM are described later.

A database of homology-derived secondary structure of proteins (HSSP)

One of the main features used in the proposed method is PSSM produced by a database of homology-derived secondary structure of proteins (HSSP), so the details of HSSP are discussed in this section.

HSSP was introduced by Sander *et al.* [87] and the principle of this database is quantifying the relation between sequence similarity, structure similarity, and alignment length by aligning proteins of known structures which are discovered as homologies based on the threshold curve. This database is publically available through <http://mrs.cmbi.ru.nl/mrs-web/> and all entries in Protein Data Bank (PDB) [64] are available in this database. As a part of HSSP, a position specific score matrix is appeared in the section of *SEQUENCE PROFILE AND ENTROPY*, and it contains nine different information at each position (Figure 2.4).

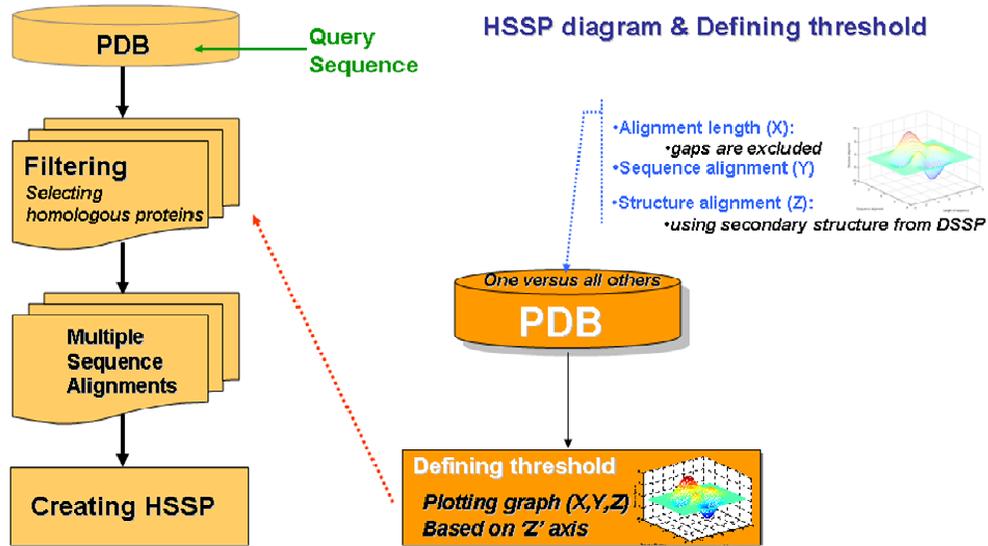


Figure 2. 5 Flow char of calculating HSSP

Each point in 3D space is calculated by comparing one versus all others in PDB. Once all points are plotted in 3D space, two variables, sequence similarity and the length of aligned sequences, are defined by user's perspectives about good structure similarity; therefore, any sequences having above the threshold of similarity score and alignment length are considered as structural homology.

After the threshold of alignment score and length is defined, HSSP is now ready to process the input query sequences. The process of HSSP consists of three steps.

At the first step, when query sequence is given, HSSP searches all PDB entries to get potential structure homology. To find the structural homology, HSSP is aligning the query sequence to all sequences in PDB. At the next step, the selected homologous sequences are filtered based on the predefined threshold value such that the sequences having above the threshold of similarity scores and aligned length are selected and others are removed. At the final step, the remaining sequences are assigned as probes, and the multiple sequence alignments are

processed. PSSM profile used in the proposed method is created from the result of multiple sequence alignments.

Now let's keep continuing on introducing rest columns in HSSP output-file shown in figure 2.5. *ENTROPY* and *RELENT* are respectively the sequence variability and its normalized value at each position of multiple sequence alignments. In detail, entropy and its normalization is calculated by the summation of negative logarithm of the frequencies of 20 amino acids at that position multiplied by the negative frequencies of 20 amino acids themselves. The equation is defined at the equation (2.3) [87]. Here, $S(i)$ is the entropy of position i , R is a residue among 20 amino acids, and f_{R_i} is the frequency of amino acid R at position i . Normalized entropy, $RELENT(i)$, is calculated with divided by maximum entropy such that the range of $S(i)$ is $0 \leq S(i) \leq \ln 20$, so the normalized entropy is the value divided by $\ln 20$. The equation is defined at the equation (2.4).

$$S(i) = -\sum_R^{20} f_{R_i} \ln f_{R_i} \quad (2.3)$$

$$RELENT(i) = \frac{S(i)}{\ln 20} \quad (2.4)$$

WEIGHT is evolutionary conservation score at each position, and calculated with the summation of a mismatch rate multiplied by a mutation rate between all possible pairs of homologous sequences, and the final value is defined with the summation divided by the summation of mismatch rate between all possible pairs of homologous sequences.

$$weight(i) = \frac{\sum_{k,l} w_{kl} \mathbf{sim}(R_{ik}, R_{il})}{\sum_{k,l} w_{kl}} \quad (2.5)$$

Here, k and l denotes sequence pair, therefore the number of all possible pairs among N sequences is $N(N-1)/2$ pairs. w_{kl} is the fraction of amino acid mismatches over the alignment length L , so the more similar sequence has the lower weight.

$$w_{kl} = 1 - \frac{1}{L} \sum_i^L \delta(R_{ik}, R_{il}) \quad (2.6)$$

$Sim(R_{ik}, R_{il})$ is mutation similarity of residue R between sequence k and l at i^{th} position defined by Dayhoff's matrix [86]. $\delta(R_{ik}, R_{il})$ denotes the delta function such that if between two sequences k and l , two residues at position i are identical then this function returns values 0 and others 1.

Physicochemical properties

Many studies have supported that the physicochemical properties of amino acids could be used for understanding the mechanisms of protein-protein interactions [1, 35-37, 54, 87-93]. Among the published physicochemical properties of amino acids, nine different physicochemical properties were selected as major features of this study: hydrophobicity, hydrophobic moments, hydrophilicity, hydrophilic moments, propensity, propensity moment, isoelectric point, isoelectric moment and mass.

Hydrophobicity is the scale of a physical property on a molecule that is repelled from water, and this hydrophobic molecule tends to be non-polar, so it prefers other neutral molecules and nonpolar solvents. Due to its distinct characteristics, hydrophobicity is often considered as one of the most important features of protein bindings and used them for analyzing and defining PPIs [1, 87-93]. The scale of the hydrophobicity used in this study was referred from Eisenberg *et al.* [74] and shown in the Table 2.3 and denoted as **HPO**.

Hydrophilicity is the scale showing a physical property of a molecule that can transiently bond with water or polar solvents through hydrogen bond [89]. The scales are appeared in the Table 2.3 and denoted as **HPI**.

Propensity is the relative frequency of different amino acid residues in the interface of complexes defined by Jones and Thornton [69], and the natural logarithms of these scales are shown below and denoted as **PP**.

Isoelectric point (pI) is the pH at which a particular molecule or surface of electrical charges are equilibrium. The scales of isoelectric point are shown in the Table 2.3 and denoted as **pI**.

The *mass* of amino acid residues are also considered as an important feature in this study, and the scales are shown in the Table 2.3 and denoted as **MS**. In table 2.3, **AA** column contains 20 amino acid residues.

Table 2.3 The scale of physicochemical properties

AA	HPO	HPI	PP	pI	MS	AA	HPO	HPI	PP	pI	MS
R	-1.76	-0.5	0.27	5.405	156.2	Y	0.02	-2.3	0.66	5.705	163.2
K	-1.1	3	-0.36	5.61	128.2	C	0.04	-1	0.43	6.31	103.1
D	-0.72	3	-0.38	5.945	115.1	G	0.16	0	-0.07	6.065	57
Q	-0.69	0.2	-0.11	5.65	128.1	A	0.25	3	-0.17	6.11	71.1
N	-0.64	0.2	0.12	5.43	114.1	M	0.26	-1.3	0.66	5.705	131.2
E	-0.62	3	-0.13	5.785	129.1	W	0.37	-3.4	0.83	5.935	186.2
H	-0.4	-0.5	0.41	5.565	137.1	L	0.53	-1.8	0.4	6.035	113.2
S	-0.26	0.3	-0.33	5.7	87.1	V	0.54	-1.5	0.27	6.015	99.1
T	-0.18	-0.4	-0.18	5.595	101.1	F	0.61	-2.5	0.82	5.755	147.2
P	-0.07	0	-0.25	6.295	97.1	I	0.73	-1.8	0.44	6.04	113.2

To project an amino acid sequence into a vector form, each amino acid in a protein sequence is converted to corresponding scale values. Next, by using the sequence of scale values along with original amino acid sequences, the average $\langle H_i \rangle$ and moments $\langle \mu_{Hi} \rangle$ of scale values

were calculated by sliding the window through the target sequences. The size of window is defined by external users.

At the calculation of moments, the property, *mass*, was excluded; therefore, total nine different feature values were created at each amino acid in the target sequence from five different physicochemical properties (i.e. five $\langle H_i \rangle$ values and four $\langle \mu_{H_i} \rangle$ values).

Although more details on generating sequence features of the proposed method are discussed in Chapter 3, the simplified forms of equations are shown below.

$$\langle H_i \rangle = \frac{1}{2N+1} \sum_{n=-N}^N h_n^{(i)} \quad (2.7)$$

$$\langle \mu_{H_i} \rangle = \frac{1}{2N+1} \left[\left(\sum_{n=-N}^N h_n^{(i)} \sin(\delta\theta) \right)^2 + \left(\sum_{n=-N}^N h_n^{(i)} \cos(\delta\theta) \right)^2 \right] \quad (2.8)$$

Here, N is one side of window size; therefore $(2N+1)$ is total window size including the center amino acid. $h_n^{(i)}$ is the scale value of an amino acid in the window, and $\delta\theta$ is the step function of the gyration angle between two consecutive residues in the sequence. In this paper, we used parameters $N=5$ and $\delta\theta=100^\circ$ found by Gallet *et al.*[10].

As a part of *geometric property*, the distance of 20 amino acids centered at each residue on the protein sequence is also considered and this feature is reinterpretation of Kini and Evans's analysis [9, 72-73]. They examined 1,600 PPI sequences and found that proline residues appear within four residues on either side, usually one or two residues away from the binding site. Inspired by this idea, shortest distance from the target residue to 20 amino acid residues were examined and used as a geometric feature in this study.

2.4 Classifiers

Support Vector Machine (SVM)

Data classification is a common problem in machine learning such that by considering binary classification problems, some data points are belonging to one of the classes, and the classifier is to decide whether a data point is belonging to one of the classes. Compared to other classifiers, Support Vector Machine (SVM) defines hyperplane with maximal margin. Maximum-margin hyperplane is one of the possible hyperplanes that can be defined on the given data points. Let's assume that there are N data points with a binary classification problem then the number of possible cases of hyperplanes on this space is 2^N ; therefore, to get the optimal solution of the given problem, it is important to define what the best hyperplane is. One of the reasonable choices to select the best hyperplane is choosing the hyperplane that separates two classes with maximum-margin. In other words, the hyperplane that maximize the distance between the nearest points of each class is chosen. More details on theoretical background of SVMs are shown below.

For a linearly separable problem, SVMs define discriminant function $g(\mathbf{x}_i) = \mathbf{w} \cdot \mathbf{x}_i + w_0$ from the given dataset $\mathcal{D} = \{\mathbf{x}_i, y_i\}$ where \mathbf{w} is a weight vector that will be defined by SVM, $\mathbf{x}_i \in \mathcal{R}^n$ is data points, w_0 is a bias term, and y_i are corresponding class labels $y_i \in \{\pm 1\}$, $i = 1, \dots, m$. The discriminant function satisfies following constraints.

$$\begin{aligned} g(\mathbf{x}_i) &> 0, \text{ if } y_i = 1 \\ g(\mathbf{x}_i) &< 0, \text{ if } y_i = -1 \end{aligned} \tag{2.9}$$

For linearly non-separable problems, slack variable ξ_i which regulates the deviation of a data point from optimal hyperplane [90] is introduced and accordingly the discriminant function is defined by $y_i(\mathbf{w} \cdot \mathbf{x}_i + w_0) \geq 1 - \xi_i$, $\xi \geq 0$, and the maximum margin ρ is defined by:

$$\frac{y_i g(\mathbf{x}_i)}{\|\mathbf{w}\|} \geq \frac{1}{\|\mathbf{w}\|} = \rho, i = 1, \dots, m \quad (2.10)$$

Equation (2.10) shows that minimizing \mathbf{w} induces maximizing ρ . SVMs are maximum margin classifier, so SVMs are designed for minimizing both \mathbf{w} and ξ_i as following equation.

$$\begin{aligned} \Phi(\mathbf{w}, \xi) &= \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^m \xi_i \\ \text{subject to : } & y_i (\mathbf{w} \cdot \mathbf{x}_i + w_0) \geq 1 - \xi_i, \xi_i \geq 0 \end{aligned} \quad (2.11)$$

To solve this optimization problem, the method of Lagrange multipliers is applied and the function is defined as following equation.

$$\begin{aligned} L(\mathbf{w}, w_0, \xi_i; \boldsymbol{\alpha}, \boldsymbol{\beta}) &= \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^m \xi_i \\ &\quad - \sum_{i=1}^m \alpha_i [y_i (\mathbf{w}^t \mathbf{x}_i + w_0) - 1 + \xi_i] \\ &\quad - \sum_{i=1}^m \beta_i \xi_i \\ \text{with } & \alpha_i \geq 0, \beta_i \geq 0, \xi_i \geq 0 \end{aligned} \quad (2.12)$$

The solution of equation (2.12) has to minimize \mathbf{w} , w_0 , and ξ_i , and maximize $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$; therefore, three conditions are defined from this equation.

$$\begin{aligned} \frac{\partial L}{\partial \mathbf{w}} &= \mathbf{w} - \sum_{i=1}^m \alpha_i y_i \mathbf{x}_i = \mathbf{0} \\ \therefore \mathbf{w} &= \sum_{i=1}^m \alpha_i y_i \mathbf{x}_i \end{aligned} \quad (2.13)$$

$$\frac{\partial L}{\partial w_0} = \sum_{i=1}^m \alpha_i y_i = 0 \quad (2.14)$$

$$\begin{aligned}
\frac{\partial L}{\partial \xi_i} &= C - \alpha_i - \beta_i = 0 \\
\text{where } \alpha_i &\geq 0, \beta_i \geq 0, \xi_i \geq 0 \\
\frac{\partial L}{\partial \xi_i} &= C - \beta_i = \alpha_i \geq 0 \\
\alpha_i &= C \text{ if and only if } \beta_i = 0 \\
\therefore 0 &\leq \alpha_i \leq C
\end{aligned} \tag{2.15}$$

With these conditions, optimization can be solved in a dual problem and the final dual equation is defined by:

$$\begin{aligned}
\psi(\mathbf{a}) &= \sum_{i=1}^m \alpha_i - \frac{1}{2} \sum_{i,j=1}^m y_i y_j \alpha_i \alpha_j (\mathbf{x}_i^t \cdot \mathbf{x}_j) \\
\text{subject to} & \\
i) \sum_{i=1}^m \alpha_i y_i &= 0 \\
ii) 0 &\leq \alpha_i \leq C
\end{aligned} \tag{2.15}$$

The linear SVMs which is inner product of data points $(\mathbf{x}_i^t \cdot \mathbf{x}_j)$ can be readily extend to nonlinear SVMs by applying the kernel trick such that replace the inner product in linear SVMs to a nonlinear kernel function $K(\mathbf{x}_i, \mathbf{x}_j)$, which satisfies Mercer's theorem [90].

For SVM implementation, two versions of SVM are commonly used and they are free for academic usages; SVM^{light} [91] which can be downloaded from <http://svmlight.joachims.org/> and libSVM [92] which can be downloaded from <http://www.csie.ntu.edu.tw/~cjlin/>. For the experiments in this thesis, SVM^{light} is used.

Random forests

Random forests [79-80] is an ensemble method that integrates multiple versions of predictors, and several studies show that this method can lead better accuracies [98, 116-118] than non-ensemble methods. The idea behind this method is that multiple versions of predictors can extend the space of hypotheses. Consequently this can increase the possibility to contain the optimal hypothesis for the given problem, and integrated predictors can reduce the risk of choosing wrong hypothesis. To get the advantage on this principle, random forests manipulated Classification and Regression Trees (CART) such that each tree is growing without pruning and by using randomly selected samples and random variables. The number of variables evaluated at each splitting node is a subset of entire feature space and defined by $\text{int}(\log_2 M + 1)$. M is the total number of variables in the given dataset.

However, the original random forests framework is not suitable for the problem of predicting interface residues in this study because the number of interface residues is much smaller than non-interface residues in protein complexes [71-72, 93]. In other words, the difference between positive and negative samples sizes which are interface and non-interface residues respectively are huge. This problem is very challenging and commonly known as imbalanced data classification problem [98, 118-123]. To handle this problem, guided random sampling approach is used in this study such that each tree is growing with same number of positive and negative samples with replacement from the original training set; therefore, with the guided sampling, the imbalanced problem can be simply corrected into a normal problem. This approach does not mean to change the principle of random forests, but it can be considered as a special case of the original random selection on random forests framework. For the experiments,

2/3 of positive samples and the same number of negative samples are randomly selected to grow trees.

The high dimensional feature space produced by the proposed method is also a challenging problem since most classification methods suffer from the curse of dimensionality. Fortunately, in contrast to the Occam's razor, the random subspace feature selection method used in random forests framework can take advantage of the high dimensionality and can improve accuracy as the complexity of feature space is grown [94].

In practice, features are separated into three groups to increase the diversity of trees in the forests and explore the feature space more efficiently. More details on generating features and constructing random forests model for the experiments will be discussed in Chapter 3.

2.5 Conventional Methods

Most recently, more sophisticated machine learning methods are applied to predict protein-protein interaction sites. Yan *et al.* [77, 95] applied broad concepts of patch analysis [69-71] in which a patch consisting with 11 consecutive residues in a amino acid sequence as shown in figure 2.6 is considered for deriving features from PSSM.

Wang *et al.* [78] used similar strategy to Yan's method to predict protein interaction sites. There were two main differences between Wang and Yan's method. Instead of using neighboring consecutive residues, Wang defined a patch as 11 spatially neighboring amino acid residues (Figure 2.7). In other words, Wang used structural information of protein complexes to define 10 geometrically closest neighborhoods of the target amino acids. In addition to using PSSM, Wang incorporated the evolutionary conservation score derived from phylogenetic tree as a new feature.

More differences were found in the way training a classifier[90]. Although Yan and Wang both used SVM as primary classifier, Wang incorporated ensemble method with SVM such that five different SVM models were trained with 5-fold cross validation, and the final decision was made by majority vote of these models. Details of both methods are described in the next chapter including creating features and training models.

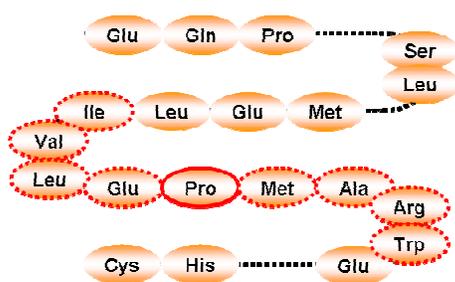


Figure 2.6 Sequence patch

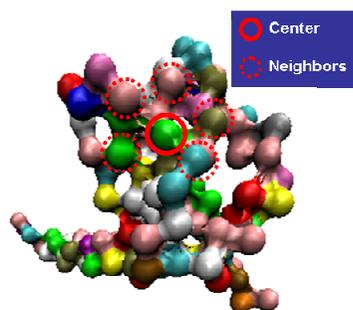


Figure 2.7 Spatial patch

Chapter3. Method and results

3.1 Methodology

Although identifying protein interaction sites have significant impact on understanding cellular systems and designing new drugs, conventional methods requiring protein structure information are not suitable for handling currently existing huge number of proteins due to the lack of structure information and extensive computational complexity. Indeed, the high cost of producing structure information is another limitation of structure-based methods. Compared to structure-based methods, sequence-based machine learning algorithm is more efficient by means of costs and computational complexity, thus it has drawn increasing interest.

The proposed method extracts wide range features from protein sequences without using any structure information and uses random-forests framework to effectively utilize these features and to handle imbalanced data classification problem commonly encountered in binding site predictions. The details of features and random-forests-based integrative model are discussed in this chapter together with experiments and results.

3.1.1 Extracting wide range features

The proposed method extracts three different groups of features from a protein sequence: *physicochemical property and evolutionary conservation score, amino acid distances, and position specific score matrix (PSSM)*. Each group of features is kept as individual feature vector instead of one long feature vector in order to emphasize the characteristics of individual feature group established by the distributions of group members. For the convenience, physicochemical property and evolutionary conservation score, amino acid distances, and PSSM are called Group I, Group II, and Group III respectively.

Group I consists of *nine physicochemical properties and an evolutionary conservation score*: hydrophobicity, hydrophobic moment, hydrophilicity, hydrophilic moment, propensity, propensity moment, isoelectric point, isoelectric moment and mass [36, 112]. Hydrophobicity and hydrophobic momentum were originally introduced for identifying membrane α helix proteins from soluble proteins [5, 6], and later they were used for predicting protein binding sites in the apolipoprotein E sequence [10, 76]. The values of this category are calculated by sliding a window centered at target amino acid along with the given sequence (Figure 3.1).

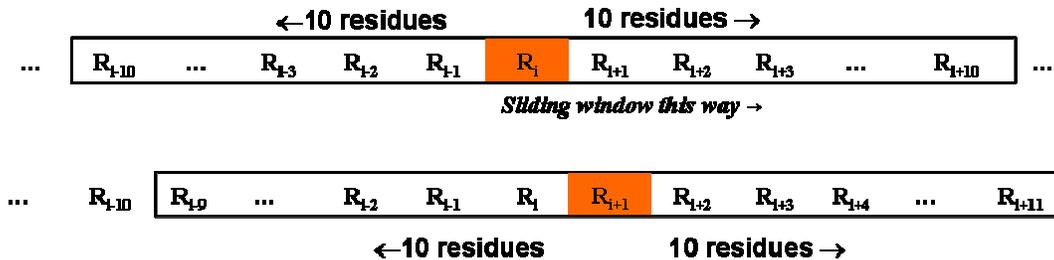


Figure 3.1 Schematic diagram of sliding window with size 10

The equations of calculating hydrophobicity and hydrophobic moment are shown below.

$$\langle H_i \rangle = \frac{1}{2N+1} \sum_{n=-N}^N h_n^{(i)} \quad (3.1)$$

$$\langle \mu_{H_i} \rangle = \frac{1}{2N+1} \left[\left(\sum_{n=-N}^N h_n^{(i)} \sin(\delta\theta) \right)^2 + \left(\sum_{n=-N}^N h_n^{(i)} \cos(\delta\theta) \right)^2 \right]^{\frac{1}{2}} \quad (3.2)$$

Here, N is a half of the window size or a half of the amino acids defining the window, so $(2N+1)$ is the actual size of a window including the center amino acid i . $h_n^{(i)}$ is the hydrophobicity scale of a amino acid. More precisely $h_n^{(i)}$ represents n amino acids away from the target amino acid i . $\delta\theta$ is the step function of the gyration angle between two consecutive residues in the sequence. Gallet *et al.* [10] found that parameters, $N=5$ and $\theta=100^\circ$, gave the most successful results, so same parameter values are used for the experiments in this study. The values of hydrophobicity are taken from the scale developed by Eisenberg *et al.* [74] (see table 2.3 for details).

Rest seven more features in *Group I* are also generated by following same procedures of hydrophobicity and hydrophobic moment: hydrophilicity, hydrophilic moment, propensity, propensity moment, isoelectric point, isoelectric moment and mass (actual scales of each properties are shown in table 2.3). Notice that mass does have moment!

One more feature in Group I, the evolutionary conservation score, is extracted from HSSP (see the Chapter2 '*A database of homology-derived secondary structure of proteins*' for more details on HSSP); therefore, *Group I* belonging to 10 feature values per a residue: nine physicochemical features and an evolutionary conservation score.

Group II consists of *amino acid distance*: the shortest distance from the target residue to each of 20 amino acids is calculated. For example, in figure 3.2, a feature vector of target residue, Methionine (M) is calculated by filling out the distant vector of 20 amino acids such that a residue, R is five residues away from the target residue M on left side when the distance from the target residue M to M itself is considered as 0. Once both sides of distance vectors, right and left side of a residue M, are filled out, two vectors are compared and then a smaller distance value except -1 which means that the corresponding residue is not existed is chosen for the final distance. This will create a row vector for each residue having a size of 20.

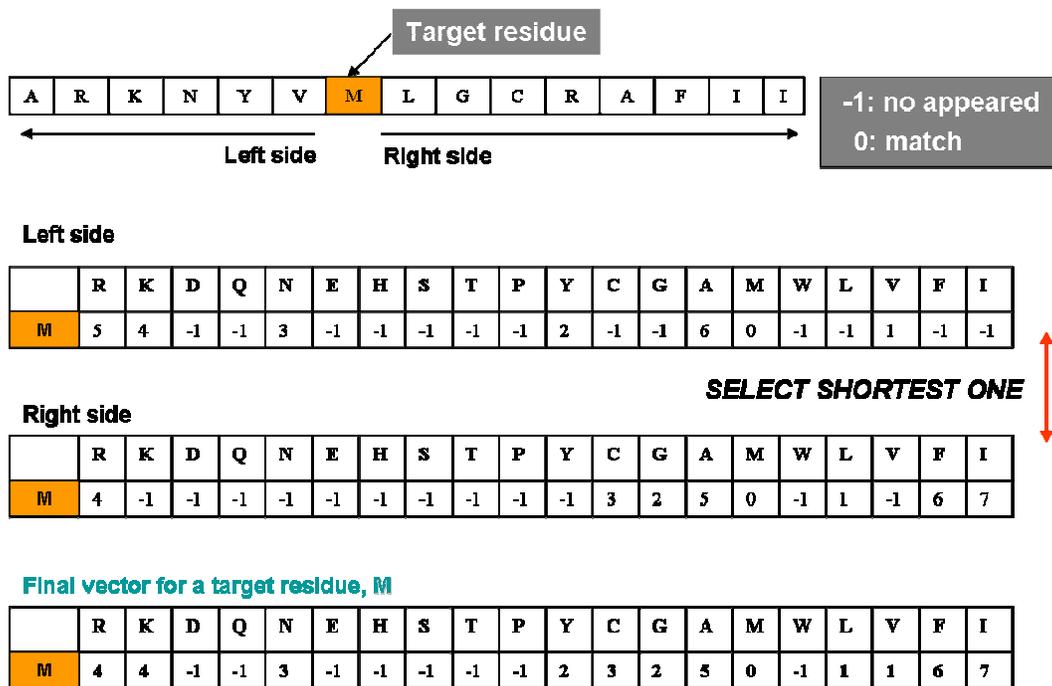


Figure 3.2 Creating distance vector

Group III consists of *position specific score matrix (PSSM)*: PSSM calculated by HSSP is used such that each row vector consisting of 20 scores in PSSM is concatenated according to the order of residues in a window. This makes a final feature vector with a size 420 (20×21) per a residue in a protein sequence (see chapter 2.3 for more details of PSSM and HSSP).

To get the final feature vector used in this study, *Group I* and *Group II* requires one more step such that each feature vectors in both groups are conducting sliding window again. At this final step, window size 21 centered on target residue is used such that as shown in figure 3.1 the feature vector of current residue R_i is considered as a target residue, and the final feature vector for R_i is a concatenated vector with feature vectors of 10 residues in both sides of the target residue. Notice that PSSM feature, *Group III* does not require this windowing method since the feature vector itself was created by concatenation.

Consequently, the size of final feature vectors for each residue is 210, 420, and 420 for *Group I*, *II* and *III* such that *Group I* consists of a vector with 10 feature values and a sliding window requires 21 neighborhood residues including the target residue itself, so the final feature vector are resulted in a vector consisting of 210 (10×21) values. The *Group II* consists of a vector with 20 feature values, so sliding window with 21 residues produces a vector consisting of 420 (20×21). The last *Group III* does not conduct sliding window; therefore it remains a vector consisting of 420 values.

3.1.2 Constructing an integrative method

There are two issues that the proposed method is not trivial to directly using conventional classifiers: having multiple feature groups and imbalanced datasets.

The proposed method has multiple groups of features that represent different characteristics of protein sequences; therefore, the ideal classifier requires having full access on exploring the feature spaces in order to retrieve various and rich characteristics of protein sequences. However, most classifiers are very restricted on exploring feature spaces including SVM, decision tree and NaïveBayes. Although exploring feature space under SVM is more flexible than others by means of using kernel tricks, the principles are still bounded by the projection of features rather than reorganizing or reformulating the feature itself.

The second issue is that identifying protein interface residues is a highly imbalanced problem. For example, the dataset used in this study has 2829 positive samples (interface residues), and 24616 negative samples (non-interface residues including surface and interior residues). In other words, the number of negative samples is much bigger than that of positive samples, and studies show that imbalanced dataset is likely to lead a highly biased model resulted in poor and unreliable predictions.

In contrast to other classifiers, a model of random forests is obtained by manipulating both feature space and data samples, and studies show that random forests often outperforms other conventional classifiers [96-99]; therefore, the proposed method manipulates the principles of random forests to solve the problems mentioned above. The details are described below.

To explore multiple features efficiently, two approaches can be considered: merging datasets and integrating models. Merging datasets is the simplest solution such that multiple feature groups are merged into a single dataset. However, this approach can under/over estimate

a typical feature due to the significance of data distribution, instead of reflecting biological relevance toward interface residues. In other words, under the strategy of merging dataset the features consisting of the highest dimension could be more significantly affected toward building the final model instead of weighting individual feature groups systematically. It is not a biologically meaningful but this also contradicts the assumption that the binding sites or interfaces residues are defined by various factors instead of a dominant feature. Compared to simply merging dataset, the second approach, integrating models produced by individual feature groups, sounds more biologically relevant and closer to the fundamental assumption of this study since this method focuses on the models built from different factors or feature groups; therefore, the problem now can be reformulated from finding a dominant feature to identifying the optimal relationships among the feature groups. In other words, the proposed method focuses on revealing the relationships among the given groups of features to build a robust model for identifying interface residues, instead of focusing on a dominant feature which is critical to define interface residues. Although several techniques exist [100-103] for integrating models, the comparisons related to this study have not been reported. Indeed, integrating methods are often time consuming and computationally intensive tasks; therefore, comparing the performance of ensemble methods and investigating new method are of interest.

As mentioned above, due to the favorable characteristics of random forests toward the problems of identifying interface residues consisting with multiple feature groups and imbalanced dataset, the proposed method borrowed the principles of random forests by manipulating the learning algorithm such that the final model for predicting interface residues is built up with majority vote from the pool of multiple trees produced by random forests.

In this study, random-forest resolves the imbalanced problem by correcting it into a normal balanced problem such that samples training each tree is controlled by randomly selecting 2/3 of positive samples (interface residues or minority class) and the same number of negative samples (non-interface residues or dominant class). Although this sounds little awkward, it actually does not violate the principles of random forests since the guided sampling can be considered as special case of random samplings as well as keeping the randomness of the sampling.

For the second issue, multiple feature groups are handled at each splitting or decision node while a tree is growing such that at each splitting node in a tree, the best feature is evaluated by randomly chosen 100 features from a feature group, and each tree in the random-forest is grown without pruning. At the last step, the pool of majority vote is made by producing 100 trees from a feature group as following random data and feature sampling. For the prediction of a new instance, each tree learned from multiple feature groups makes a decision as a vote, and the final prediction for the instance is determined by majority votes among the pool of votes made by entire trees grown with multiple feature groups.

3.2 Experiments and Results

3.2.1 Data sources

In order to validate the proposed method, a set of 70 protein-protein heterocomplexes previously used in the studies of Chakrabarti and Janin [104] and Yan *et al.* [77, 95] are used.

To minimize the effects from biased dataset, redundant and peptide sequences are filtered out as following: proteins with less and equal to 30% sequence identity measured by Smith-Waterman algorithm[88] are selected and molecules with fewer than 10 residues were removed from original dataset. Some proteins which are not available in HSSP[87] and DSSP[82] programs are also omitted.

After following the stringent filtering steps, total 54 heterocomplexes composed with 99 polypeptide chains were remained and downloaded from PDB. These 54 heterocomplexes were grouped into six categories: antibody-antigen, protease-inhibitor, enzyme, large-protease, G-proteins, and miscellaneous. The details of each category are described below.

Antibodies also known as immunoglobulins are gamma globulin proteins which are the class of protein in the blood or other bodily fluids of vertebrate and used by the immune system to identify and neutralize or inactivate foreign objects which are called antigen. Although the general structure of antibodies are very similar each other, the tip of antibodies is extremely variable, so these variable sites make possible to bind numerous antigens [89, 105].

A protease also termed peptidase or proteinase is a type of cellular enzymes that conducts proteolysis. In other words, a protease breaks down or digests proteins. Protease inhibitors are molecules that inhibit the function of protease [89].

Enzymes are proteins that catalyze chemical reactions together with substrates which is a molecule that binds to the active sites of enzyme and later is converted into products [89]. In the dataset, this category contains several different types of enzymes (e.g. Ribonuclease, Ribonuclease inhibitor, Porcine pancreatic alpha-amylase, and Bean lectine-like inhibitor etc.)

A large-protease is selected from protease based on their surface area which is greater than 2000\AA^2 , and this dataset contain serine protease and hydrolase.

G-proteins (guanine nucleotide-binding proteins) are signal transducers that transmit chemical signals outside the cell, and causing changes inside the cell [106].

Miscellaneous category contains different types of proteins from all others (i.e. viral protein, hormone, protein complexes etc.)

Table 3.1 Categories of proteins and corresponding PDB IDs

Antibody-antigen	1AO7_A, 1AO7_B, 1AO7_D, 1AO7_E, 1DVF_AB, 1DVF_CD, 1IAI_LH, 1IAI_MI, 1JH1_A, 1KB5_AB, 1KB5_LH, 1NCA_LH, 1NCA_N, 1NFD_ABCD, 1NFD_EFGH, 1NMB_LH, 1NMB_N, 1NSN_LH, 1NSN_S, OSP_LH, 1OSP_O, 1QFU_A, 1QFU_B, 1QFU_H, 1QFU_L, 1YQV_LH, 2JEL_LH, 2JEL_P, 3HFM_LH
Protease-inhibitor	1ACB_E, 1ACB_I, 1AVW_A, 1AVW_B, 1CHO_I, 1FLE_E, 1FLE_I, 1HIA_ABXY, 1HIA_IJ, 1MCT_A, 1STF_E, 1STF_I, 1TGS_I, 1TGS_Z, 2SIC_I, 2SNI_E, 2SNI_I, 3SGB_E, 4CPA_I
Enzyme	1BRS_ABC, 1BRS_DEF, 1DFJ_E, 1DFJ_I, 1DHK_A, 1DHK_B, 1FSS_A, 1FSS_B, 1GLA_F, 1GLA_G, 1UDI_E, 1UDI_I, 1YDR_E, 1YDR_I
Large-protease	1BTH_PQ, 1DAN_LH, 1DAN_TU, 1TBQ_LHJK, 1TBQ_RS, 1TOC_ABCDEFGH, 1TOC_RSTU, 4HTC_I
G-proteins	1AGR_AD, 1AGR_EH, 1GG2_A, 1GG2_B, 1GG2_G, 1GOT_A, 1GOT_B, 1GOT_G, 1GUA_A, 1GUA_B, 1TX4_A, 1TX4_B, 2TRC_P
Miscellaneous	1AK4_AB, 1ATN_A, 1ATN_D, 1DKG_AB, 1EFN_AC, 1FC2_C, 1FC2_D, 1HWG_A, 1HWG_BC, 1IGC_A, 1IGC_LH, 1SEB_ABEF, 1YCS_A, 1YCS_B, 2BTF_A, 2BTF_P

The 99 poly peptide chains consist of total 27445 residues. Among 27445 residues, 13774 surface residues and 2829 interface residues are identified by DSSP same as used in other literatures[69-70, 77-78, 107-108] such that residues which correspond to the value of relative solvent accessible surface area (RASA) greater or equal to 25% are defined as surface residues (see the section 2.2 for details), and residues are defined as interface residues if they satisfy two conditions: i) interface residues have to be subset of surface residues and ii) the difference of accessible surface areas (ASA) between unbound molecular and bounded complex has to be great than 1\AA^2 . Thus 2829 residues are defined as interface residues (positive class), and other 24616 residues including non-binding surface residues and non-surface residues are defined as non-interface residues (negative class). The ratio between positive to negative samples is about 1:9; therefore, the given problem is apparently an imbalanced classification problem [96, 99, 109-113] and known to be very challenging problem in machine learning area. Indeed, the length of sequence is significantly varied such that the minimum and maximum length of a sequence in this study is 20 (1YDR_I) and 1148 (1TOC_ABCDEFGH) respectively. The average number of interface residues in a protein is 29 amino acid residues and this is about 1% of residues in a protein. The statistic detail of the dataset is shown below.

Table 3.2 Statistics of the dataset

Measure Category	Ave. Pos.	Ave. Neg.	Min Length	Max Length	Neg / Pos	Pos / Surf	Pos / All
1	25.69 ±14.7	304.83 ±193.5	85	884	13.87±9.98	0.19±0.13	0.11±0.08
2	16.47 ±6.5	145.36 ±105.5	36	460	9.08±5.96	0.25±0.13	0.15±0.10
3	24.14 ±9.6	244.64 ±172.3	20	532	11.16±9.48	0.27±0.20	0.18±0.21
4	74.13 ±49.6	326.38 ±326.3	61	1148	4.33±3.32	0.40±0.18	0.27±0.15
5	36.23 ±18.2	214.00 ±163.6	71	692	6.79±4.76	0.36±0.23	0.21±0.19
6	23.06 ±13.7	265.38 ±160.2	58	746	12.73±7.97	0.19±0.10	0.10±0.06
Overall	28.58 ±23.3	249.15 ±187.6	20	1148	10.68±8.35	0.25±0.17	0.15±0.14

1: antibody-antigen; 2: protease-inhibitor; 3: enzyme; 4: large-protease; 5: G-protein; 6: miscellaneous

In the table 3.2, the category index corresponds to antibody-antigen, protease-inhibitor, enzyme, large-protease, G-protein, and miscellaneous proteins from 1 to 6, and the statistical categories appeared as measure are defined as the average number of positive (interface) residues (Ave. Pos.), the average number of negative (non-interface) residues (Ave. Neg), the minimum length of a protein sequence in the corresponding category (Min Length), the maximum length of a protein sequence in the corresponding category (Max Length), the ratio between the number of negative and positive residues (Neg/Pos), the ratio between the number of positive and surface residues (Pos/Surf), and the ratio between the number of positive and total residues in a sequence (Pos/All). The statistics of the dataset shows that the number of interface residues per a protein sequence is less than 5% of total residues in a protein sequence. This verifies again that the given problem is significantly imbalanced.

3.2.2 Evaluation criteria

Due to the special status of the given dataset, the evaluation method and criteria are carefully designed. In order to make the evaluation more realistic, the proposed method is evaluated and compared with other conventional method based on the concept of leave-one-out cross validation (LOOCV) and several criterion functions are used.

Table 3.3 defines the given criterion functions. In this definition, true positive (TP) is the number of true interface residues that are predicted correctly; true negative (TN) is the number of true non-interface residues that are predicted correctly; false positive (FP) is the number of true non-interface residues that are predicted to be interface residues; and false negative (FN) is the number of true interface residues that are predicted to be non-interface residues.

Table 3.3 Criterion functions

Overall accuracy: $\frac{TN + TP}{TN + FP + FN + TP}$	Sensitivity: $\frac{TP}{FN + TP}$
Balanced accuracy: $\sqrt{\text{Sensitivity} \times \text{Specificity}}$	Specificity: $\frac{TN}{FP + TN}$
Correlation coefficient (CC): $\frac{TP \times TN - FP \times FN}{\sqrt{(TP + FN)(TP + FP)(TN + FP)(TN + FN)}}$	

The overall accuracy is the ratio of the number of correctly predicted residues in both positive and negative cases to the total number of residues in the dataset, so this measures overall performance of a model. Although the overall accuracy is the one of most commonly used evaluation criteria, overall accuracy alone will not be sufficient to evaluate the performance of the proposed system since the given problem is highly imbalanced. Let's assume a model that always predicts a residue as a non-interface residue no matter what actual inputs are. Surprisingly, the overall accuracy of the model will be around 90%. This happens because the number of negative residues is about nine times bigger than positive residues. However, is it really good model although this model does not predict any interface residue at all? Of course it is not a good model; therefore, to avoid over/under estimation of the imbalanced problem, balanced accuracy and receiver operating characteristic (ROC) curves are added as important criteria. The balanced accuracy is related to the product of sensitivity and specificity, so high accuracies on both positive and negative residues together will receive high score from this evaluation function. ROC curves are drawn in terms of true positive rate and false positive rate which are sensitivity and 1-specificity respectively. The last evaluation function is correlation coefficient and this ranges from -1 to +1. The value, -1, denotes a worst possible classifier, +1 indicates a best possible classifier, and 0 imply a random classifier.

3.2.3 Leave-One-Out Cross Validation (LOOCV)

To compare the performance of the proposed method, two conventional methods are implemented. One of the methods is introduced by Yan *et al.* [77], and this method uses PSSM with 11 consecutive residues in a sequence. The other method is proposed by Wang *et al.* [78], and this method uses both PSSM and evolutionary conservation score derived from HSSP with 11 spatially neighboring residues. The details of these methods are discussed in chapter 2.5.

The methods used in this study are implemented by strictly following the procedures described in their papers. For the fair comparison, all the methods are trained and tested on the same datasets. To evaluate the methods more realistic, the first evaluation is conducted by deriving the concept of LOOCV such that one of the 99 polypeptide chains including interface residues and all other residues is used as test data and the remaining all other amino acid sequences from 98 chains are used as training data; this process is repeated 99 times and the final results are averaged over the test results. In other words, instead of validating each sample, individual chain or sequence is considered as a sample in ordinary LOOCV.

The results of leave-one-out test are shown from Figure 3.3 to 3.8 by averaging the results of each protein categories. As shown in the figures, the overall accuracy is not appropriate to evaluate imbalanced problems. For example, the results from Figure 3.3 to 3.8 cannot distinct the differences among three methods and they are all equal under the overall accuracies. However, the differences are very clear when other measurements such as sensitivity, precision, and balance accuracy are considered. Especially the distinctions obtained by changing the criteria are even clearer on Figure 3.4 and 3.8. In these categories, other two methods, Wang and Yan's method fail to classify these categories, but criteria with overall accuracies show that two methods outperform the proposed method NEW.

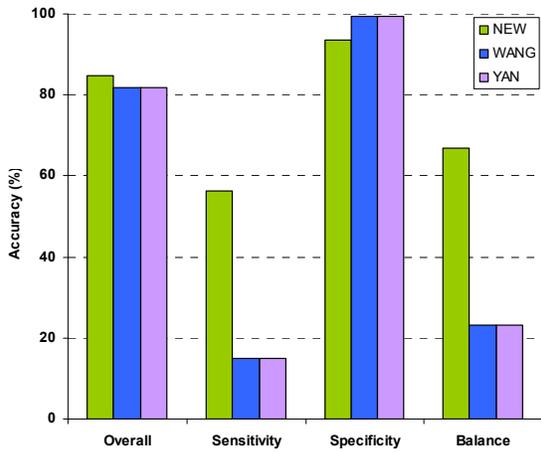


Figure 3.3 Evaluation on Antibody-antigen

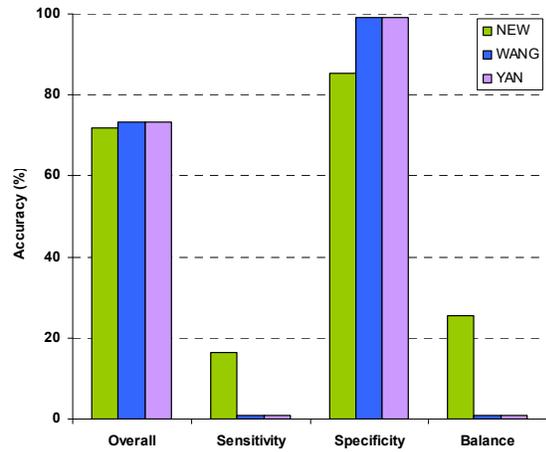


Figure 3.4 Evaluation on Enzyme

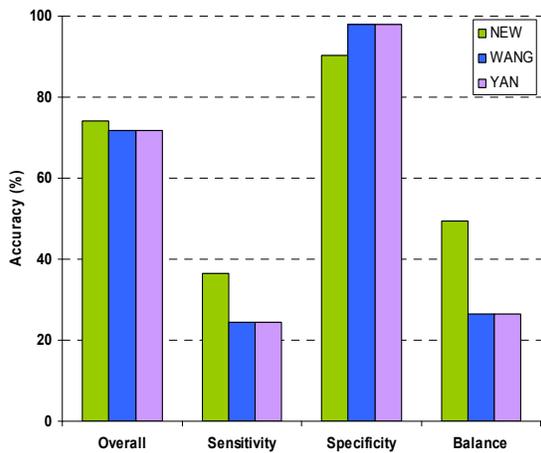


Figure 3.5 Evaluation on G-proteins

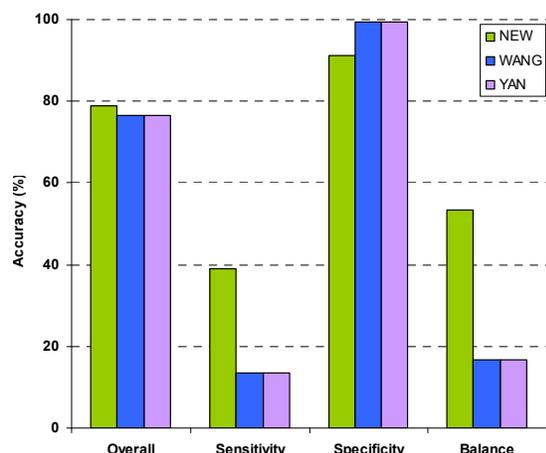


Figure 3.6 Evaluation on Protease-inhibitor

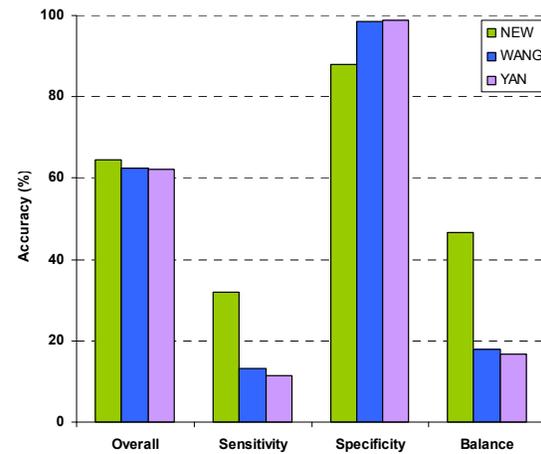


Figure 3.7 Evaluation on Large-protease

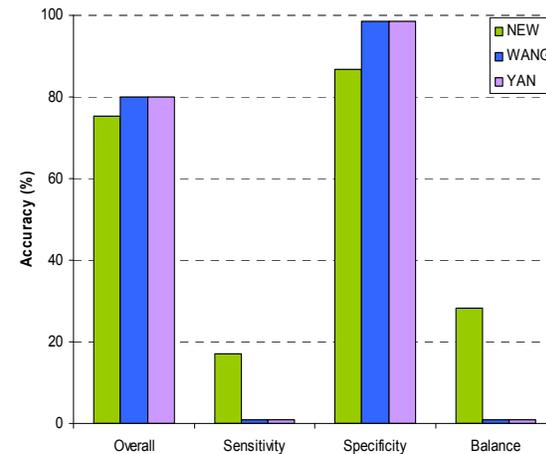


Figure 3.8 Evaluation on Miscellaneous

This happens since the number of negative samples is dominant to positive samples, so overall accuracy which considered the number of true positives and true negatives only is not suitable for imbalanced problem and the reasons are clear on these extreme cases.

It is reasonable that all three methods performs poor on two categories, Enzyme and Miscellaneous since these two categories consist with different types of proteins, and the number of proteins with same types in each category is rare, so this could lead the lack of training samples of specific proteins in these categories. It is even clear to compare the number of samples in each category and their balanced accuracy. Upon the evaluation of balanced accuracy, the performance is ordered by antibody-antigen (29) > protease-inhibitor (19) > G-protein (13) > Large-protease (8) from best to worst case. Here the numbers in parenthesis indicate the number of samples in the corresponding category and the order is proportional to the order of balanced accuracy among four categories. It is interesting that this relationship is true on NEW method only, and other methods Wang and Yan's method do not hold this relationship. Therefore, this implies that adding more samples on NEW method is more likely to improve the accuracy than other methods. To further explorer differences among three methods, the potentials of each method are compared with receiver operating characteristic (ROC) measurement.

Figure 3.9 shows the ROC curves of three sequence-based predictors: the proposed method is denoted as NEW; Yen's and Wang's method are denoted as YEN and WANG respectively. An ROC curve is a plot of the sensitivity versus (1- specificity) for a binary classifier in which the points in a plot are calculated by moving decision boundary. Sensitivity measures the capability of predicting positive samples (interface residues) correctly, and specificity determines the rate of false positives by calculating the number of incorrectly predicted interface residues among all true negative samples (non-interface residues).

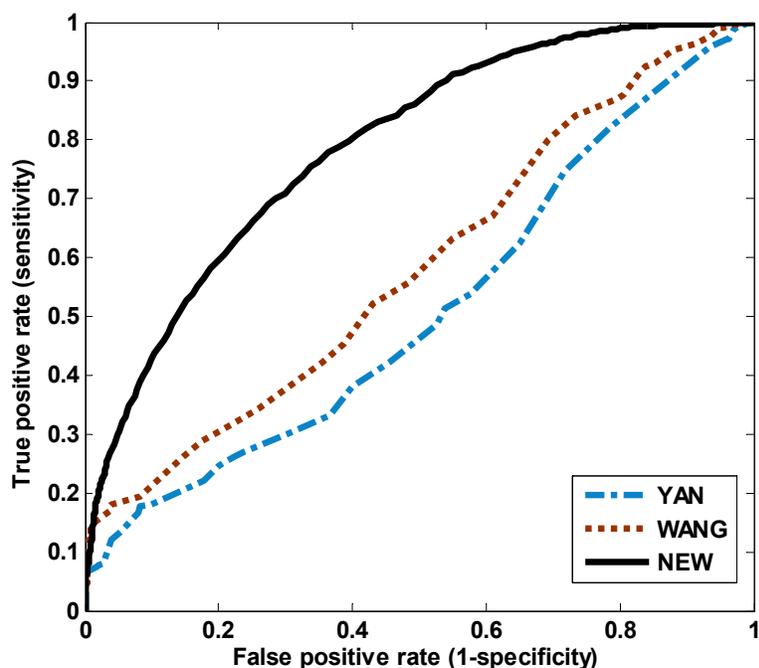


Figure 3.9 The ROC curves for three sequence-based predictors

As mentioned above, ROC is plotted by changing the decision boundary, more precisely the ROC curve of the proposed method is constructed by changing the threshold of the decision boundary for majority votes among decision trees. For example, a threshold at five implies that at least five more votes of binding sites than these of non-binding sites are necessary to classify a residue as interface residue. Otherwise, the residue is predicted as non-interface residue. In special case, it is possible that the majority votes always win if and only if the threshold is equal to zero. Therefore, varying the threshold of the majority votes will produce different values of sensitivity and specificity for the model trained with a dataset.

Similarly, the ROC for Yan's method is generated by changing the threshold of SVM which actually is the bias of SVM. Compared to Yan's methods, Wang's method is consist of five SVM models produced by five-fold cross-validations; therefore, this method requires changing two thresholds: the threshold of majority vote and the bias of SVM such that the same

bias of SVM for five models is increased or decreased and then sensitivity and specificity of the final decision is calculated by changing the threshold of the majority vote. This procedure is repeated until the ROC curve is complete. The results are similar with the direct comparisons on each category such that Figure 3.9 shows that the proposed method significantly outperforms both Yen's and Wang's method in terms of ROC curves: for example, with a false positive rate of 30% the sensitivities of Yan's, Wang's, and the proposed method are 30%, 39%, 73% respectively. Further comparison is conducted with another criterion function CC, which evaluates how predicted results correlate with true labels. With false rate of 30%, the CC values are 0.00, 0.06, and 0.28 for Yan's, Wang's and the proposed method respectively. With true positive rate 70%, the CC values are 0.02, 0.05 and 0.28 for Yan's, Wang's and the proposed method respectively. By considering the range of CC values which is from -1 (worst possible prediction) to 1 (best possible prediction), the zero value is corresponding to random guess, so the results show that Yan's and Wang's methods are close to the random predictors and the proposed method is significantly better than random guess. Thus the proposed method clearly outperforms other two methods.

As mentioned earlier, the balanced accuracies are reasonable measurement for imbalanced problems, so based on the best balanced accuracy in ROC curve, the results of each criterion function are compared among three methods. Table 3.4 shows that both Yan's and Wang's method have higher false positive rate and lower true positive rate than the proposed (NEW) method. Since the balanced accuracy is chosen, overall, sensitivity, specificity, and balanced accuracy are similar within each method, and this shows that the comparing the performance among methods by using balanced accuracy is more meaningful than other criterion functions. However, balanced accuracy itself may not fully revealing the differences or can

misleading the results; therefore, other criteria should be simultaneously considered together especially for evaluating imbalanced problems. The comparisons on the correlation coefficient at Table 3.4 also support that the proposed method outperforms other two methods. The difference between Yan’s and Wang’s methods is much clearer than the comparisons on the leave-one-out cross validation on data groups which are from Figure 3.3 to Figure 3.8.

Table 3.4 Performances at the best balanced accuracy

Criterion functions	YAN	WANG	NEW
True positive rate	0.51	0.52	0.71
False positive rate	0.54	0.43	0.28
Overall	46.6	56.5	71.9
Sensitivity	51.4	52.1	71.2
Specificity	46.1	56.9	72
* Balance	48.7	54.5	71.6
CC	-0.01	0.06	0.28

For the easy comparisons on performance and stability among three methods, a graph is drawn from Table 3.4. Figure 3.10 clearly shows that the difference between the proposed method and others, and this also shows the distinction between Yan’s and Wang’s method.

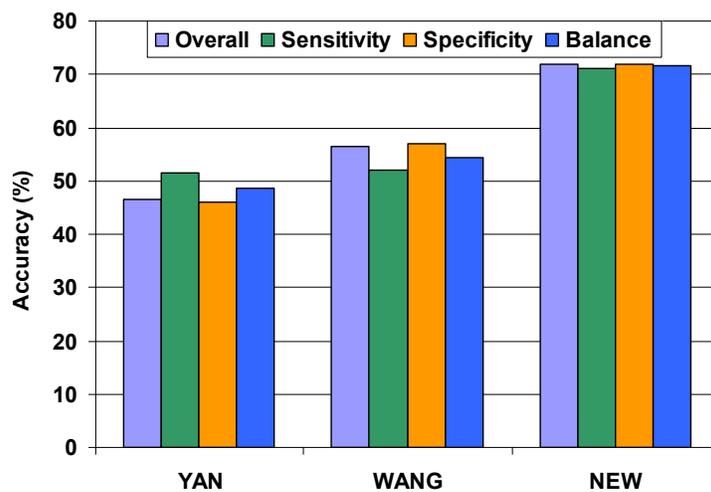


Figure 3.10 Comparing accuracies based on the best balanced accuracy

Table 3.5 and Figure 3.11 show the performances by selecting the highest overall accuracy at each method. Compared to the performances at the best balanced accuracy, three methods do not have distinct differences on overall and specificity. Indeed, the sensitivity and balanced accuracies are very poor compared to Figure 3.10 and Figure 3.11.

Table 3.5 Performances at the best overall accuracy

Criterion functions	YAN	WANG	NEW
True positive rate	0.07	0.15	0.21
False positive rate	0.01	0.01	0.01
* Overall	90.3	90.7	90.8
Sensitivity	6.7	14.6	20.9
Specificity	99.6	99.2	98.6
Balance	25.8	38.0	45.5
CC	0.19	0.28	0.33

Interestingly, CC values are increased although the sensitivity and balanced accuracies are decreased. With normal cases, CC values should be proportional to sensitivity and balanced accuracies, but the given problem is highly imbalanced, so that this can be happened and it is not surprising anymore. This means that predicting more samples to majority class (negative class) are the easiest way to improve both overall accuracy and correlation coefficients because the population of the minority class (positive class) is too small to affect majority class; therefore, both CC and overall are enforced by the predictions on majority class, and the results emphasize the accuracy of majority class. In other words, if a classifier considers 0-1 loss function then the classifier is likely to build a model that maximizes the overall accuracy because increasing the accuracy on minority class often leads more incorrect predictions on majority class than the number of correctly classified minority samples; therefore, in short, the probability of correctly predicting majority class is much easier than increasing correct predictions on both majority and minority class together, so the results shown in this study is very significant for improving the accuracy of predicting interface residues.

Indeed, maximizing overall accuracy is often resulted in sacrificing sensitivity and balanced accuracy. As shown in Figure 3.11, overall accuracy and specificity are significantly increased from Figure 3.10, but the sensitivity and balanced accuracies are decreased. The example can be found in comparing two figures Figure 3.10 and 3.11 such sensitivity in Figure 3.10 which is based on balanced accuracy is significantly decreased in Figure 3.11 in which the model is designed for predicting more samples toward negative samples by considering the maximum overall accuracy; therefore, this example proves again that a model which only focuses on overall accuracy does not help to improve generalization on imbalanced problem.

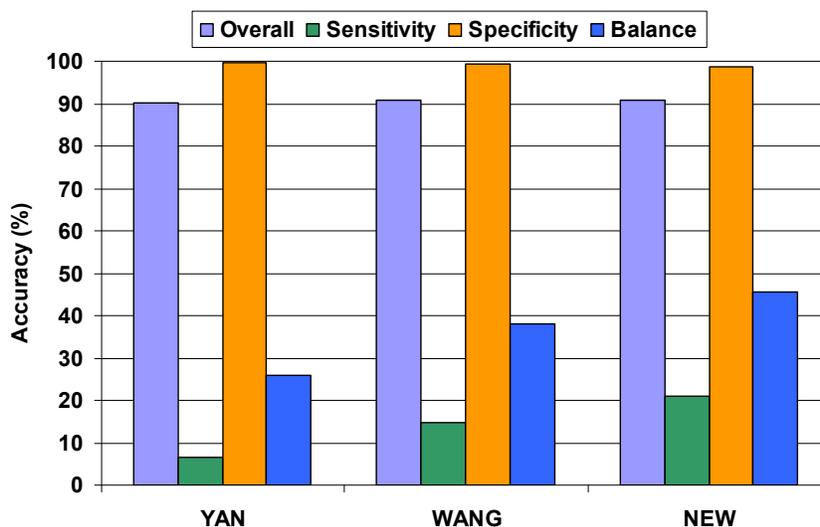


Figure 3.11 Comparing accuracies based on the best overall accuracy

For further investigations on the proposed and conventional methods, VMD software [114] and Jmol [115] is used to visualize the difference between the proposed methods and others such that the results presented in each figure are produced by three different methods with actually locations of predicted interface residues. The same models used for the experiment in one-leave-out cross-validation are applied; therefore, threshold and other parameters are remained same such that the final predictions of the proposed method are made with majority

votes, Yan's method choose the SVM parameters by cross validation upon arbitrarily chosen a series of values, and Wang's method defines SVM parameters for each five model same as Yan's method and majority votes are used for making final predictions.

Figure 3.12 shows the results of predicted interface residues on an idiotype-anti-idiotype Fab complex [116] (PDB ID: 1IAI) which is one of the antibody-antigen pairs. Figure 3.12 (a) shows binding structure of four chains I, M, H, and L which are cyan, yellow, purple, and white color respectively. In this example the complex is considered as binding chain LH and MI.

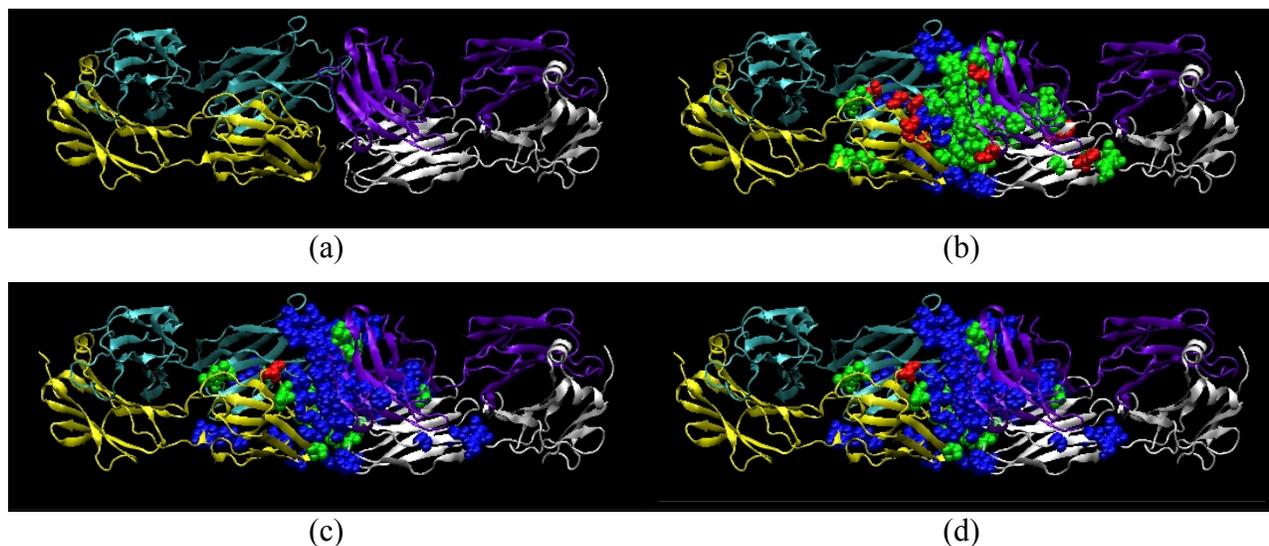


Figure 3.12 Predicted interface residues in 1IAI: **(a)** four chains: white(L), purple(H), yellow(M), and cyan(I) **(b)** predictions of the proposed method, **(c)** Wang's method, **(d)** Yan's method. Spheres with blue, red, and green are false negatives (FN), false positives (FP), and true positives (TP).

Figure 3.12 (b) depicts the prediction results based on the proposed method. Spheres denote atoms of each predicted interface residue: the green color spheres denote true positives (TP) which are the atoms of true interface residues that are predicted correctly, the blue color spheres denote false negatives (FN) which are the atoms of true interface residues that are predicted to be non-interface residues, and the red color sphere denote false positives (FP) which is the number of true non-interface residues that are predicted to be interface residues.

Figure 3.12 (c) and (d) show the prediction results of Wang's and Yan's method respectively. It is clear that the area of green color in Figure 3.12 (b) is much larger than both Figure 3.12 (c) and (d). This means that the proposed method predicted more TPs in this protein complex. Indeed, the area of blue color in Figure 3.12 (b) is much smaller than other two methods; therefore the figures show that the proposed method clearly outperforms other two methods in this example.

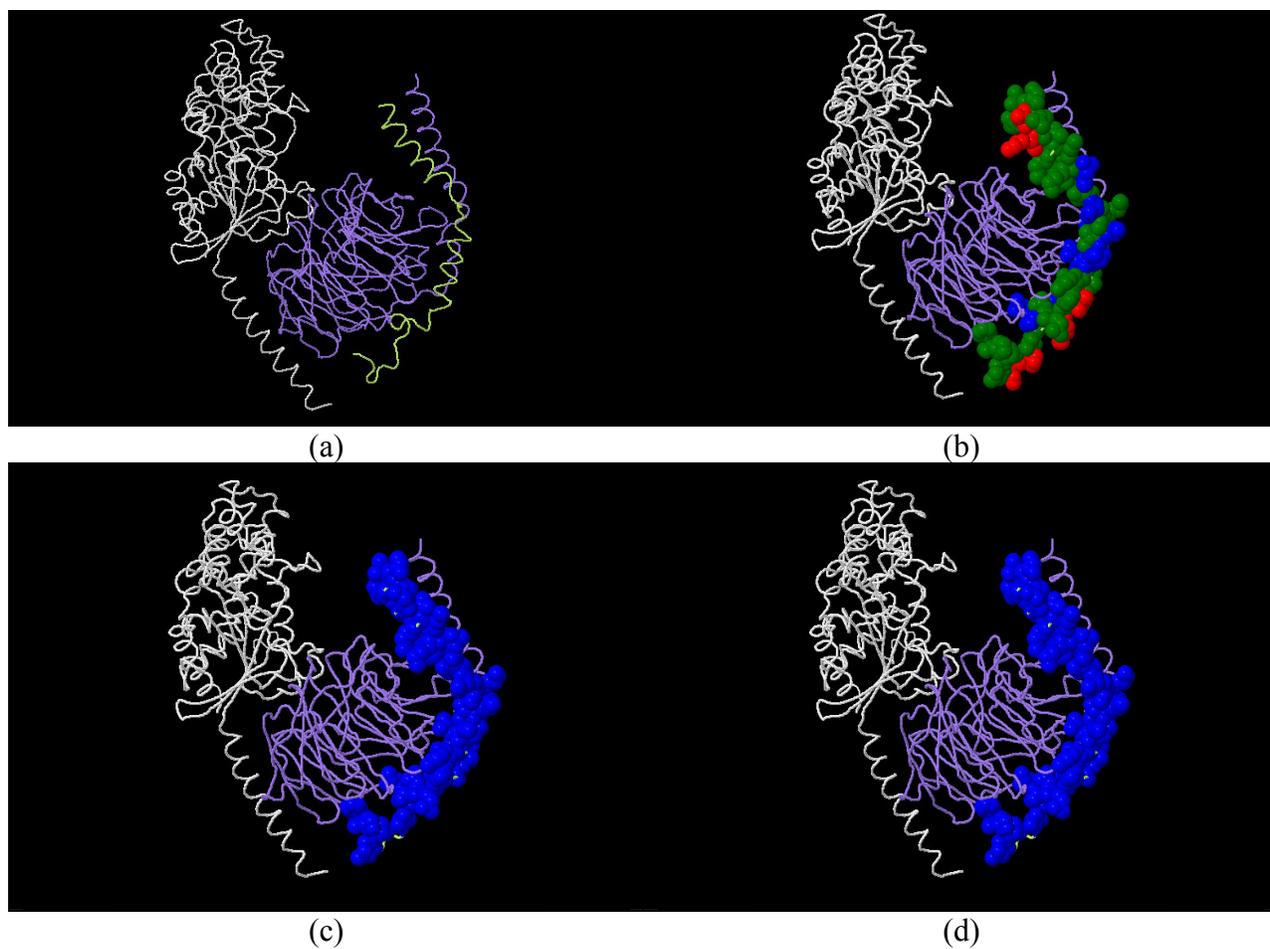


Figure 3.13 Predicted results of a chain in 1GG2 using (a) three chains: white(A), purple(B) and yellow(G), (b) the proposed method, (c) Wang's method, (d) Yan's method. Spheres with blue, red, and green are false negatives (FN), false positives (FP), and true positives (TP).

More results are shown in Figure 3.13. This example shows an extreme case among comparisons of three methods.

The predictions are made on the G protein heterotrimer $G_i \alpha 1 \beta 1 \gamma 2$ [117] (PDB ID: 1GG2). The definitions of colored spheres are same as the previous one such that green, blue, and red denotes TP, FN, and FP respectively. In this complex, it is assumed that the binding are made by a structure containing both chains A and B against to chain G. As shown in Figure 3.13 (c) and (d) two conventional methods completely lost their ability to predict interface residues and resulted in predicting all interface residues (TP) as non-interface residues (FN). It is interesting that two different methods Yan and Wang's method make similar prediction patterns as shown in Figure 3.12 and 3.13. This can be explained by considering the tendency of making decision boundary of SVM. SVM is very strong classifier and many papers show that SVM outperforms most cases due to the principles of maximal margin and structural risk minimization theory. However, the reported studies are evaluated based on balanced problems; therefore, it is of interest if SVM works well with imbalanced problems. The theoretical proof of generalization efficiency of SVM is clearly beyond the scope of this study, so instead of showing theoretical proof, the facts that observed from the results are reported here. If and only if the results shown in this study are considered and analyzed, it seems that the similar prediction patterns between Yan and Wang' are from the effect of samples in majority class which give more effect on defining support vectors; therefore, more weights are given toward majority class and the model is biased toward majority class. In addition, the marginal principle is very strong, so that this cannot be easily affected by simple sampling schemes as well as samples are selected from same distribution function. In other words, a model from SVM is generalized well so that the decision boundary is cannot be simply affected or changed. In short, SVM models are very stable so that

individual model and multiple models produced by difference samples are likely to be same. Although the data are not provided, the predictions of Wang's method among five models are correlated. Due to the lack of theoretical proof, instead of insisting the hypothesis is being true, the observed factors are reported here and the theoretical proof and deriving new classifier favorable toward imbalanced problems are left for the task of future study.

Let's come back to Figure 3.13 and compare three results, the figures clearly show that the proposed method, Figure 3.13 (a) outperforms Yan and Wang's method by identifying most true interface residues while Yan and Wang misclassify interface residues.

More results are selected from experiments based on LOOCV and shown in Figure 3.14, 3.15, and 3.16. Each row displays protein complexes first four letters denote PDB ID belonging to the protein complex followed by hyphen and alphabetic letters in which each letter lists involved chains of building a protein complex. Each column denotes overall structures of protein complexes and the types of residues identified by each method: first column shows individual chain compositions in the given protein complex, second column shows the status predicted residues identified by the proposed method and third and fourth column show residue status identified by Wang and Yan's method respectively.

Displaying schemes are same as previous ones: green, blue, and red spheres are TP, FN, and FP respectively. For the convenience of interpretation, the categories of reported complexes are given; *Antibody-antigen*: 1DVF, 1NMB, 1NCA, *Protease-inhibitor*: 1GTS, *large-protease*: 1TBQ, 1TOC, and *G-proteins*: 1GG2. The given lists of figures are only selected from the results and many figures which do not give notable differences based on visualization are removed. Actually, the difference within couple residues is not easily recognizable and some of them were blocked by neighbor residues in visualization.

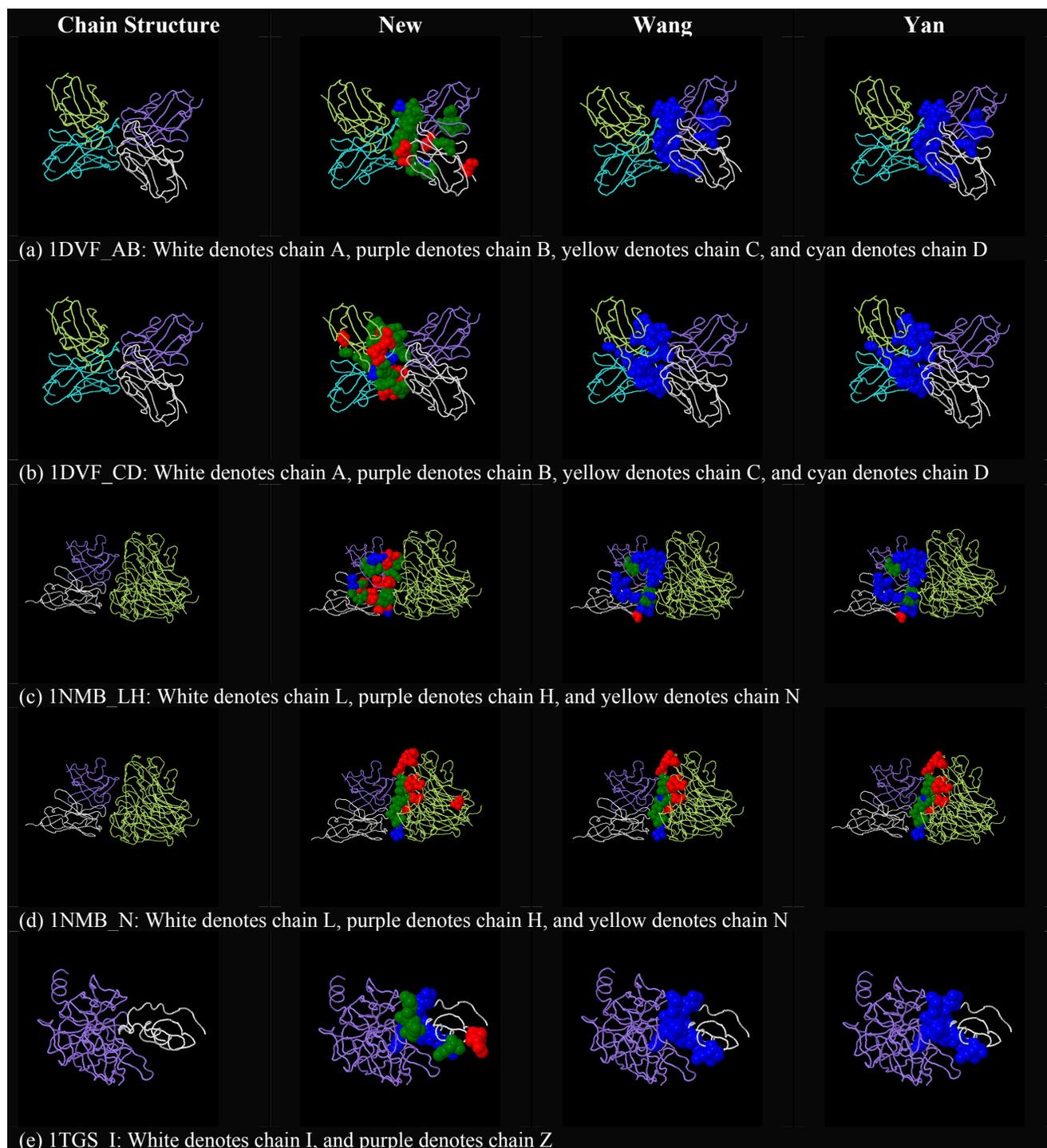


Figure 3.14 Visualization of residues identified by three methods: each row shows the protein complex used for predictions. The first column shows the chain structure of each protein complex, the second column denoted as New shows the predicted results of the proposed method, and the third and fourth column shows the predicted interface residues from Wan's and Yan's method denoted as Wang and Yan respectively. Spheres with blue, red, and green are false negatives (FN), false positives (FP), and true positives (TP) respectively.

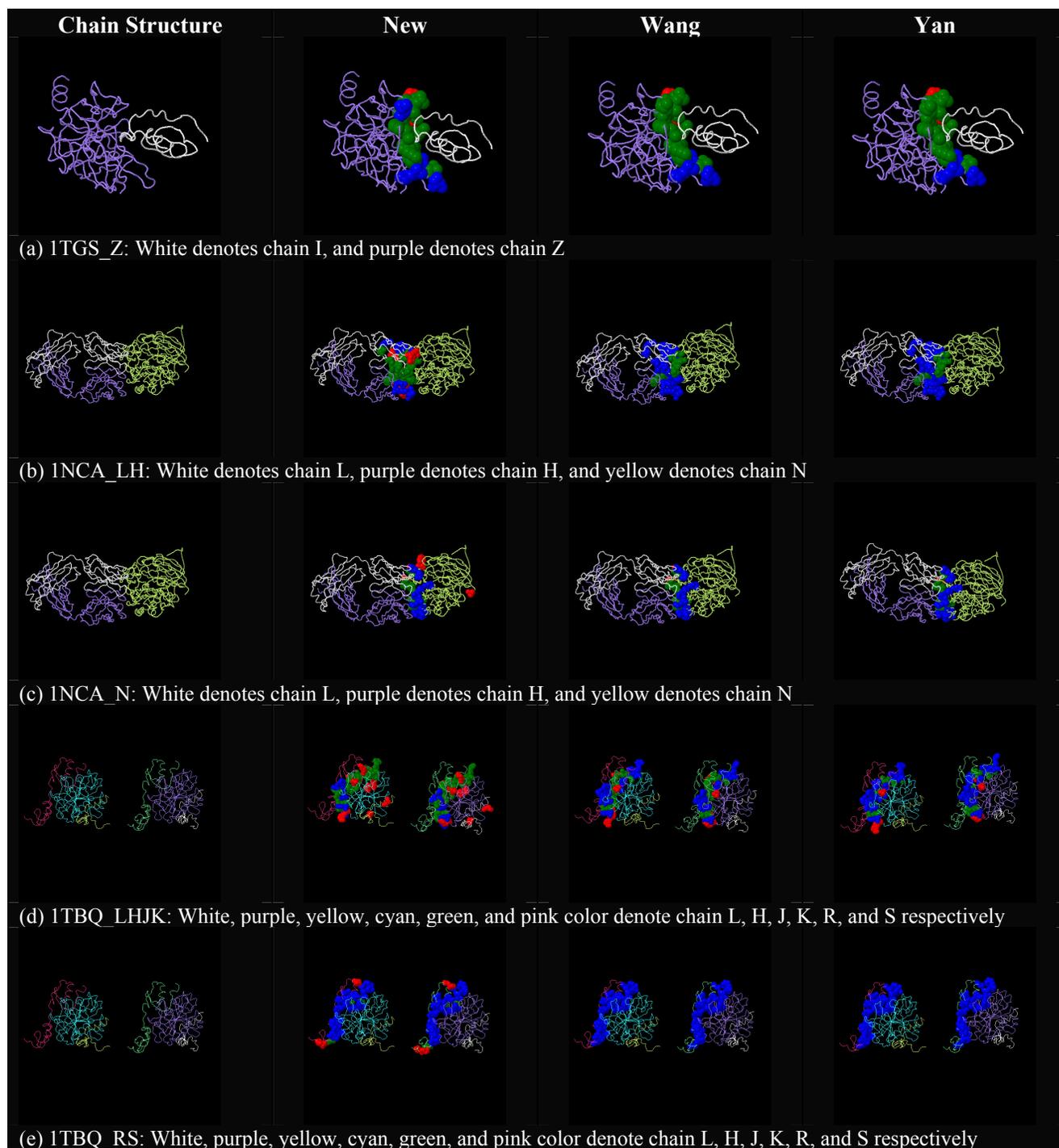


Figure 3.15 Visualization of residues identified by three methods: each row shows the protein complex used for predictions. The first column shows the chain structure of each protein complex, the second column denoted as New shows the predicted results of the proposed method, and the third and fourth column shows the predicted interface residues from Wan's and Yan's method denoted as Wang and Yan respectively. Spheres with blue, red, and green are false negatives (FN), false positives (FP), and true positives (TP) respectively.

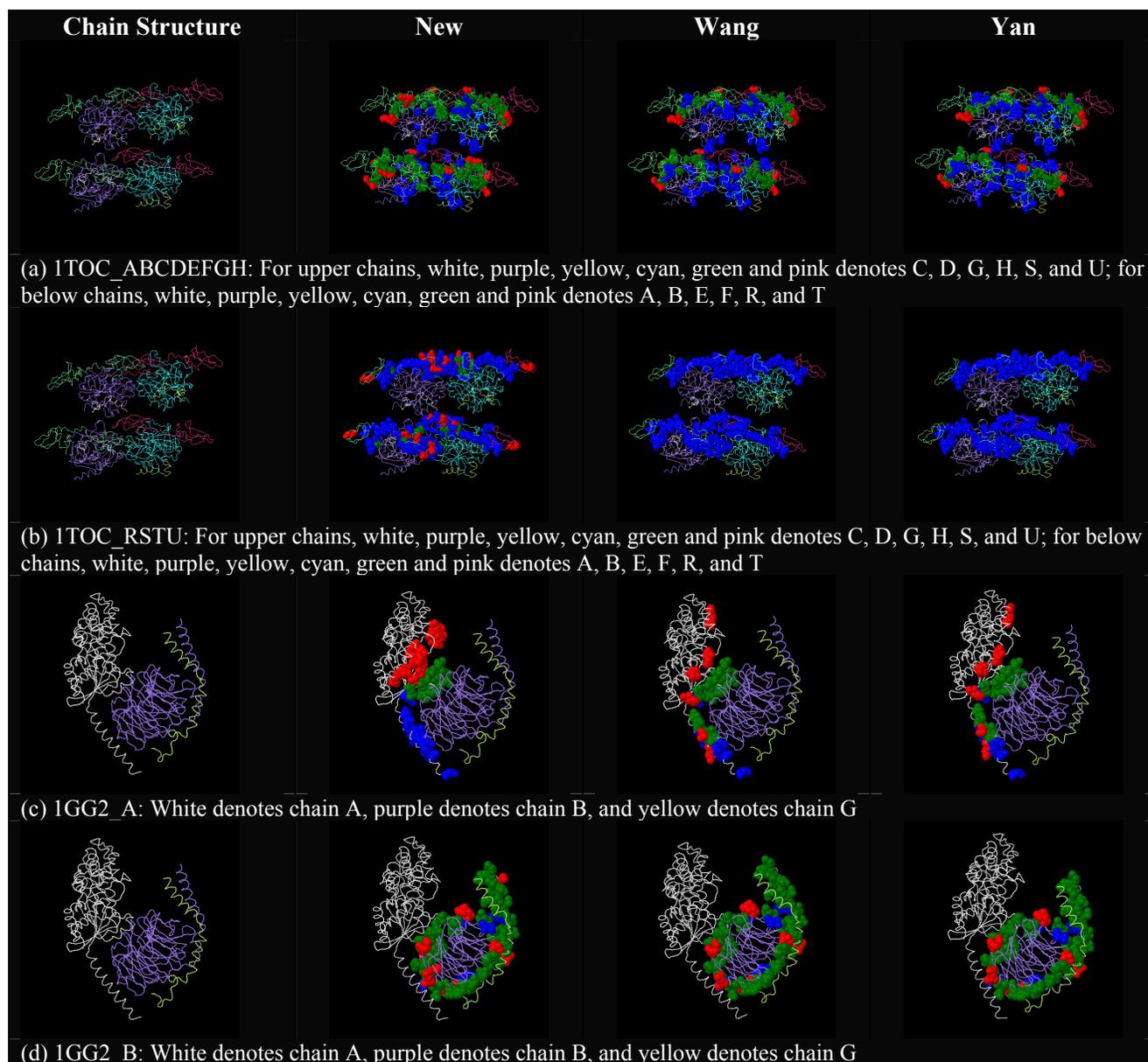


Figure 3.16 Visualization of residues identified by three methods: each row shows the protein complex used for predictions. The first column shows the chain structure of each protein complex, the second column denoted as New shows the predicted results of the proposed method, and the third and fourth column shows the predicted interface residues from Wan’s and Yan’s method denoted as Wang and Yan respectively. Spheres with blue, red, and green are false negatives (FN), false positives (FP), and true positives (TP) respectively.

Although some of the results, Figure 3.15 (e), and Figure 3.16(c) are not easy to compare their excellences between proposed method and others due to the compensation between the number of TP and FP such that at the Figure 3.16 (c) Wang and Yan’s method predicted more

true interface residues than proposed method, but by considering the total number of predictions toward true interface residues Wang and Yan predicted more FP than the proposed method; therefore, if the precision is considered as criteria then the proposed method actually outperform other two methods. Indeed, final results show that the proposed method clearly outperforms other methods.

Further comparisons in identified interface residues among three methods are conducted by using a cysteine protease inhibitor (PDB ID: 2CIO). This experiment is different from previous cases such that the structure of this complex was not available when the model was originally trained; therefore, it was not included in the group of 99 chains and considered as real testing data or independent data. To get the model used for predicting these independent datasets, all 99 chains are used as training samples. Figure 3.17 shows the differences among three different methods: Yan's method, Figure 3.17 (d) missed many interface residues compared to Wang's method, Figure 3.17 (c). The proposed method, Figure 3.17 (a) outperformed Wang's method; therefore, the proposed method performs the best among three methods. More results of the third, independent dataset are shown Figure 3.18.

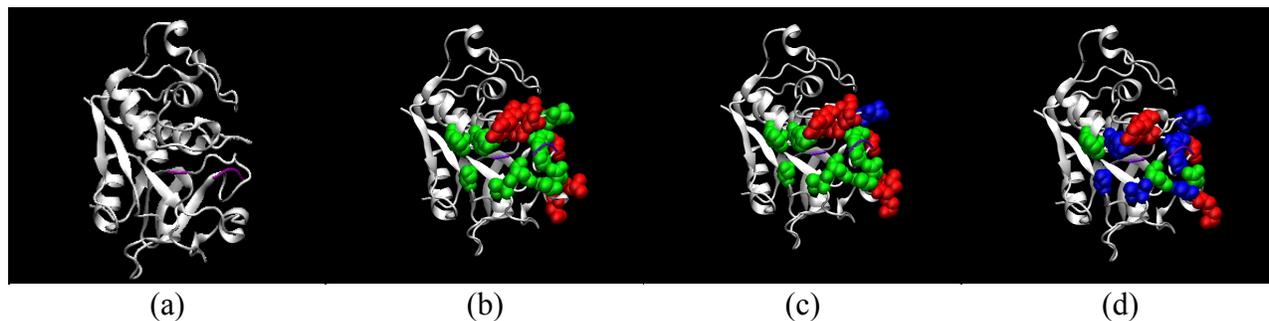


Figure 3.17 Predicted results of chains in 2CIO: (a) two chains: white(A), purple(B), (b) the proposed method, (c) Wang's method, (d) Yan's method. Spheres with blue, red, and green are false negatives (FN), false positives (FP), and true positives (TP).

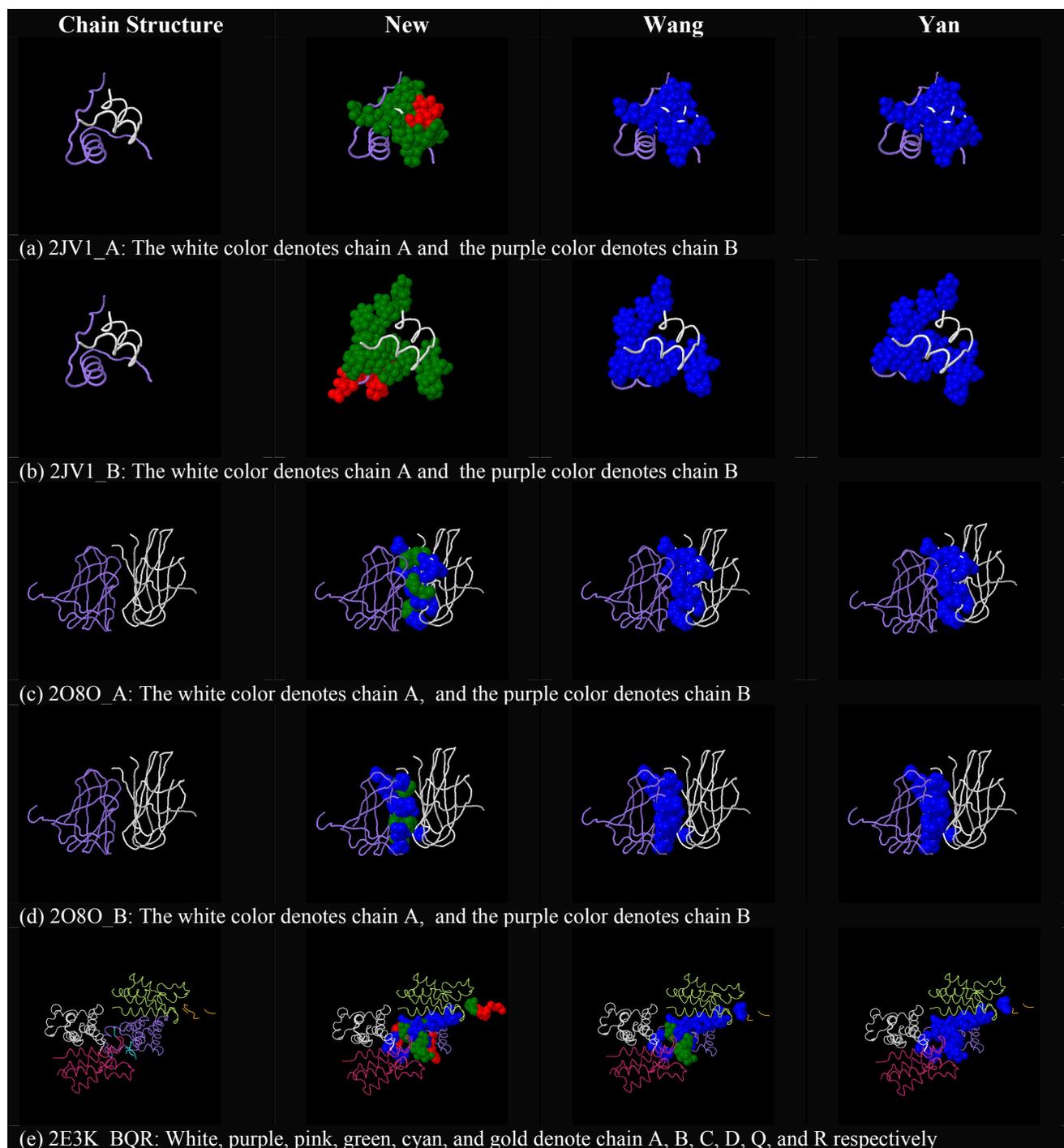


Figure 3.18 Visualization of residues in independent data identified by three methods: each row shows the results of predicted interface residues from three different methods. The first column shows the chain structure of each protein complex, the second column denoted as New shows the predicted results of the proposed method, and the third and fourth column shows the predicted interface residues from Wan's and Yan's method denoted as Wang and Yan respectively. Spheres with blue, red, and green are false negatives (FN), false positives (FP), and true positives (TP) respectively.

3.2.4 Blind test

To evaluate the capability of generalization in the proposed method, three blind tests are conducted. Without knowing the true binding sites, the blind tests evaluate the capability of predicting interface residues with peptide complexes. The structures of all protein complexes in this test are constructed with VMD software [114].

First, three structural component of the DnaK molecular chaperone system which is a member of eukaryotic 70-kDa heat-shock protein (Hsp70) family used in another study [59] is evaluated on the proposed method by predicting its potential interface residues which are not yet revealed but can be referred from other literatures.

Before describing the results, it will be worth to explain the chaperone system to help understanding the experiment and results in this section. Chaperons are proteins that help with 3-dimensional folding of proteins. DnaK is a chaperone of Hsp 70 family and has relatively strong binding affinity for adenosine triphosphate (ATP) which is like a battery storing energy in cell. Binding of ATP to DnaK leads to the release of substrates and possible rebinding of others. DnaK is divided into two separable functional units: i) ATPase activity on N-terminal and ii) binding polypeptide substrates on C-terminal.

Although DnaK itself has binding high affinity for ATP, it is slow adenosine triphosphatase (ATPase), so DnaK itself is not enough to bind and release substrates, so it needs other chaperones like a nucleotide exchange factor, Hsp24 GrpE and Hsp40 DnaJ to complete the binding and releasing process of substrates. Although the joint presence of GrpE and DnaJ stimulates ATPase activity of DnaK and hydrolyzes ATP, the action of GrpE and DnaJ are considered as being sequential since the presence of DnaJ alone leads to an acceleration in the rate of hydrolysis of DnaK-bound ATP, and the presence of GrpE alone increase the rate of

releasing the DnaK-bound ATP/ADP [118]. With the hydrolysis of ATP, DnaK is switched back into the ADP-bound form which exchanges substrates slowly. GrpE protein forms a stable ATP-sensitivity complex by binding to the ATPase unit of DnaK which is N-terminal of the DnaK [119], and the complex of GrpE and DnaK is dissociated by introducing of ATP since the role of GrpE is releasing or dissociating ADP [120]. DnaJ can bind both ATPase unit (N-terminal of DnaK) and substrate-binding unit (C-terminal of DnaK), and DnaK and DnaJ form a stable complex in the presence of ATP [119].

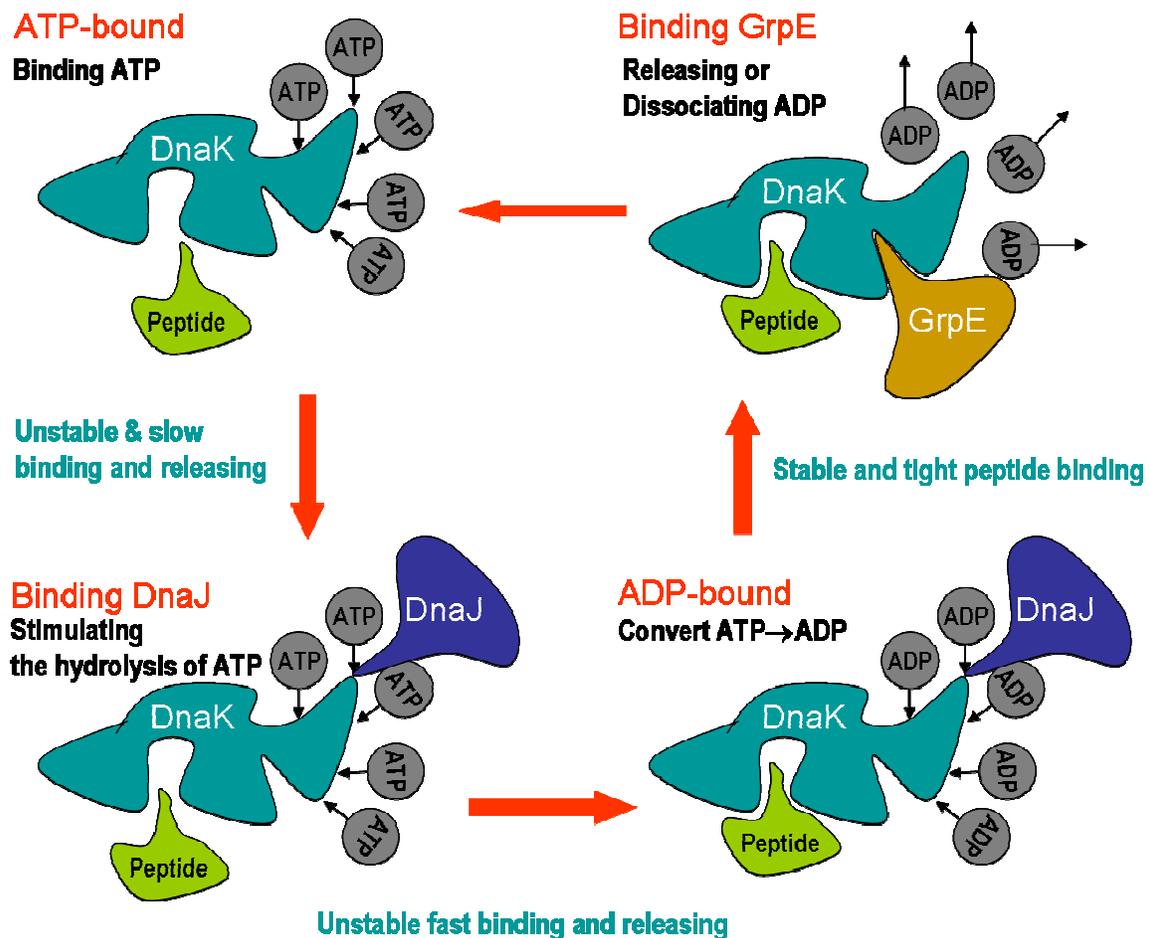


Figure 3.19 DnaK chaperone system

In summary, with the presence of ATP, the DnaK chaperone undergoes dramatic conformational changes as following: at the first stage, ATP is hydrolyzed by ATPase in the N-terminal of DnaK and then DnaK is converted to weak ADP-bound DnaK conformation which has limited affinity for peptide substrates and GrpE co-chaperone but efficiently binds the DnaJ chaperone. With the presence of DnaJ, ATP hydrolysis is accelerated, so the weak ADP-bound DnaK is converted to strong ADP-bound DnaK which binds peptide substrates more tightly. At the ATP-bound DnaK, GrpE and ATP hydrolysis promotes the fast release of the peptide substrate from the chaperone complex and convert DnaK to weak ADP-bound DnaK [121]. The simplified mechanisms of DnaK chaperone system is drawn in figure 3.19.

For the experiments, three structural components of the DnaK (eukaryotic Hsp70) molecular chaperone system which is used in another study [59] is tested. The first two components are two DnaK domains: a C-terminal domain / substrate-binding unit and a N-terminal domain / ATPase unit which are corresponding to the structure of 1DKX and 1DKG in PDB respectively. The third component is a J-domain cochaperone, DnaJ which is corresponding to the structure of 1XBL in PDB.

Figure 3.20 shows the structure of DnaJ which is corresponding to 1XBL in PDB ID, C-terminal of DnaK which is corresponding to 1DKX in PDB, and N-terminal of DnaK which is corresponding to 1DKG in PDB. The interface residues identified by the proposed method are shown as purple spheres. More precisely, individual purple sphere denotes the atoms belonging to an amino acid and green spheres depict carbon α among those atoms. For the convenience, the index of amino acids in PDB file and the name of amino acid is added as digits followed by three alphabetic letters: the number denotes the index of residue in PDB file and the three alphabetic letters represent short abbreviations of 20 amino acids.

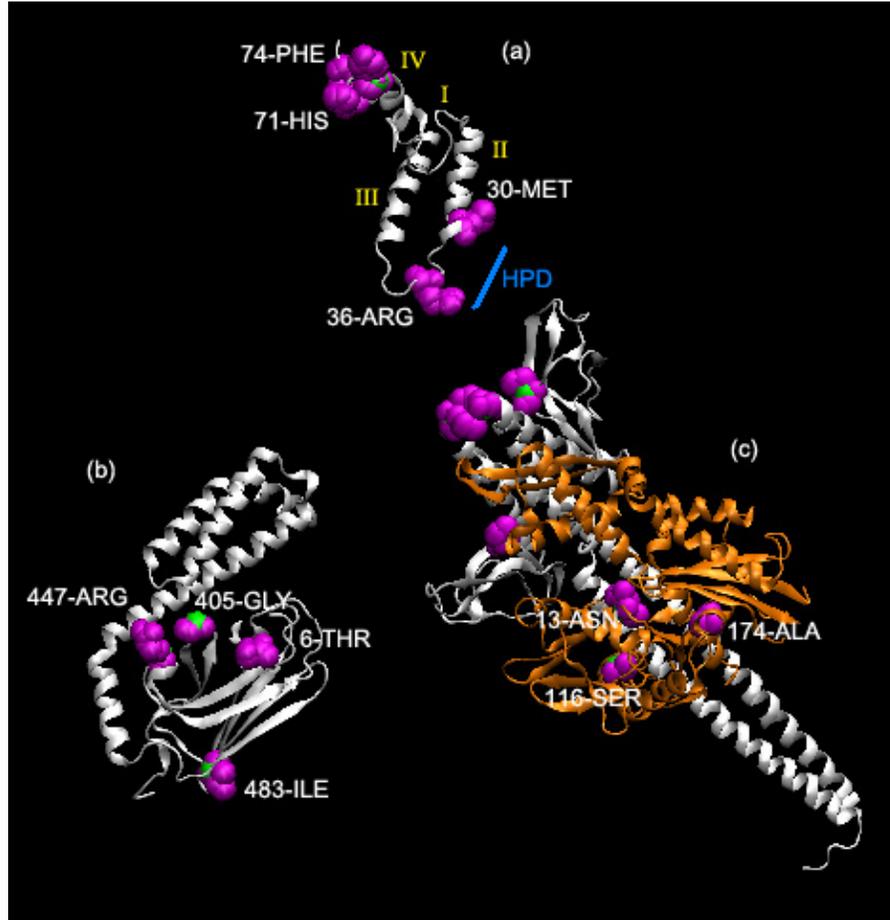


Figure 3.20 DnaK molecular chaperone system: (a) DnaJ (PDB ID 1XBL) (b) the structure of DnaK C-terminal (PDB ID 1DKX) and (c) the structure of DnaK N-terminal (PDB ID 1DKG). Orange structure denotes ATPase domain. Purple spheres denote predicted interfaces and green spheres denote carbon α at each residue.

The Figure 3.20-(a) shows that DnaJ structure (1XBL, PDB ID) consists of four α - helices and a loop region containing HPD motif (tripeptide of histidine, proline, and aspartic acid residues) between the second and third alpha helix. The HPD motif is highly conserved and presented in almost all known J-domain protein families and this is known as a critical site to stimulate Hsp70 ATPase activity, and mutations on the conserved tripeptide HPD of the J-domain abolish the ability of proteins to function with Hsp70 proteins; therefore, the HPD tripeptide could mediate specific interactions between Hsp40 and Hsp70 proteins [59, 97, 122-

125]. The proposed method predicted two amino acid residues: 30-MET and 36-ARG which are near the HPD motif (33-HIS, 34-PRO, and 35-ASP). This prediction results are similar to the results of Greene *et al.* (1998) that the potential binding sites are residues between 1 and 35 and binding sites are concentrated on the outer surface of helix II which is a right-side α -helix where the prediction made by the proposed method, 30-MET, is located. 71-HIS and 74-PHE are also shown as conserved residues based on the consensus of amino acid position [124]; therefore, these evidences show that the predictions of interface residues on DnaJ are reasonable.

For Hsp70 C-terminal of DnaK also known as substrate-binding unit (1DKX, PDB ID) in Figure 3.20-(b), the proposed method predicted four residues (6-THR, 405-GLY, and 447-ARG) as interface residues. Previous studies [126-127] showed that mutants observed in the loops on sandwich sub-domain are closely related to the peptide-binding site; therefore, the predictions made by the proposed method are in agreement with the results from Davis *et al* (1999) and Montgomery *et al.* (1999).

For Hsp70 N-terminal of DnaK also known as ATPase unit (1DKG, PDB ID) in Figure 3.20-(c), the proposed method predicted three residues (13-ASN, 116-SER, and 174-ALA) as interface residues in ATPase domain (i.e. orange color structure in Figure 3.20-(c)). Other studies [122, 126] showed that most of mutants, which affect interaction with C-terminal domain, are located in the bottom of ATPase domain. The predictions made by the proposed method are spatially very close to those mutants observed in Davis and Gassler's studies.

For the further evaluation on the proposed method, a structure of DnaK protein, 1YUW [128] is downloaded from PDB and interface residues are predicted by the proposed method. 1YUW shows the direct binding between DnaK N-terminal and C-terminal and is shown Figure 3.21.

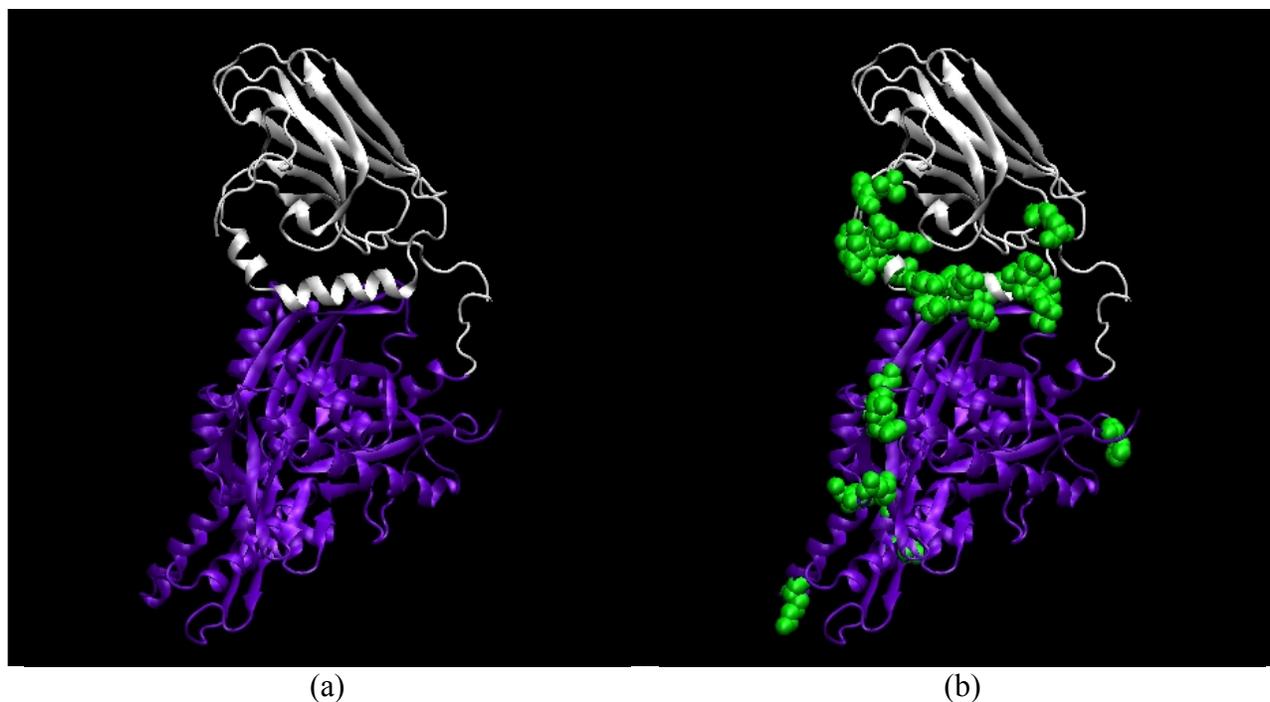


Figure 3.21 Predicted results of 1YUW using the proposed method. (a) displaying two sections: white color denotes C-terminal (amino acid 395-554) and purple color denotes N-terminal (amino acid 1-383) of DnaK, (b) atoms in predicted sites are shown as green spheres.

The results show that the proposed method predicted interface residues condensed mostly surrounding the alpha helix of C-terminal as shown in Figure 3.20-(b). The results are very similar to the results of Jiang *et al* [128]. Interestingly, more predicted interface residues are also observed in the N-terminal of DnaK, and some of these may be the binding sites between N-terminal and other chains like GrpE shown in Figure 3.20-(c).

By comparing the results between Figure 3.20 and Figure 3.21, it is interesting that in spite of same DnaK proteins, the predicted interface residues on C-terminal and N-terminal are different.

However, when you look closely into the differences between two structures depicted in Figure 3.120-(b,c) and Figure 3.21, there are differences between two structures in terms of sequences and binding sites. In fact, 1DKX and 1DKG which is DnaK C-terminal and N-

terminal respectively are from *Escherichia coli* and 1YUW is from *Bus Taurus*. Thus the sequence identity of C-terminals between 1DKX and the white chain of 1YUW is 37% and N-terminals between 1DKG and the purple chain of 1YUW is 43%. Consequently, as proposed method's point of view, it is not surprising that the predicted interface residues are in difference under the same DnaK proteins since the features of the proposed method are directly derived from protein sequences.

Finally, the proposed method is applied to predicting interface residues of 1DKG which is co-crystallized with N-terminal of DnaK and a nucleotide exchange factor GrpE. Figure 3.22 shows that the N-terminal of DnaK, ATPase domain, Figure 3.22(a) and a nucleotide exchange factor GrpE Figure 3.22(b) which is a tightly associated homodimer and stoichiometrically binds ATPase domain of DnaK. The interface residues predicted by the proposed method are depicted as green and red spheres. Although each monomer in GrpE, Figure 3.22(b) is colored as white (chain A) and purple (chain B), and corresponding interface residues are drawn as green and red spheres respectively, for the input as a test sequence is not distinguished or separated into two separated sequences but two chains are treated as a protein sequence.

The six binding sites reported by Harrison *et al.* [120] are marked as I, II, III, IV, V, and VI in each figure, and same numerals between two figures correspond to the binding site of a partner region.

The results show that the predicted interface residues in the N-terminal (ATPase domain) of DnaK are very close to those of the reported regions III, V, and VI. The predicted interface residues on chain A of GrpE which is colored as white are also located in the reported binding regions II, III, IV, V and VI. The other side of predicted residues corresponding to the binding

regions II and III in chain B of GrpE which is colored as purple are the results of the fact that GrpE exists as a dimer.

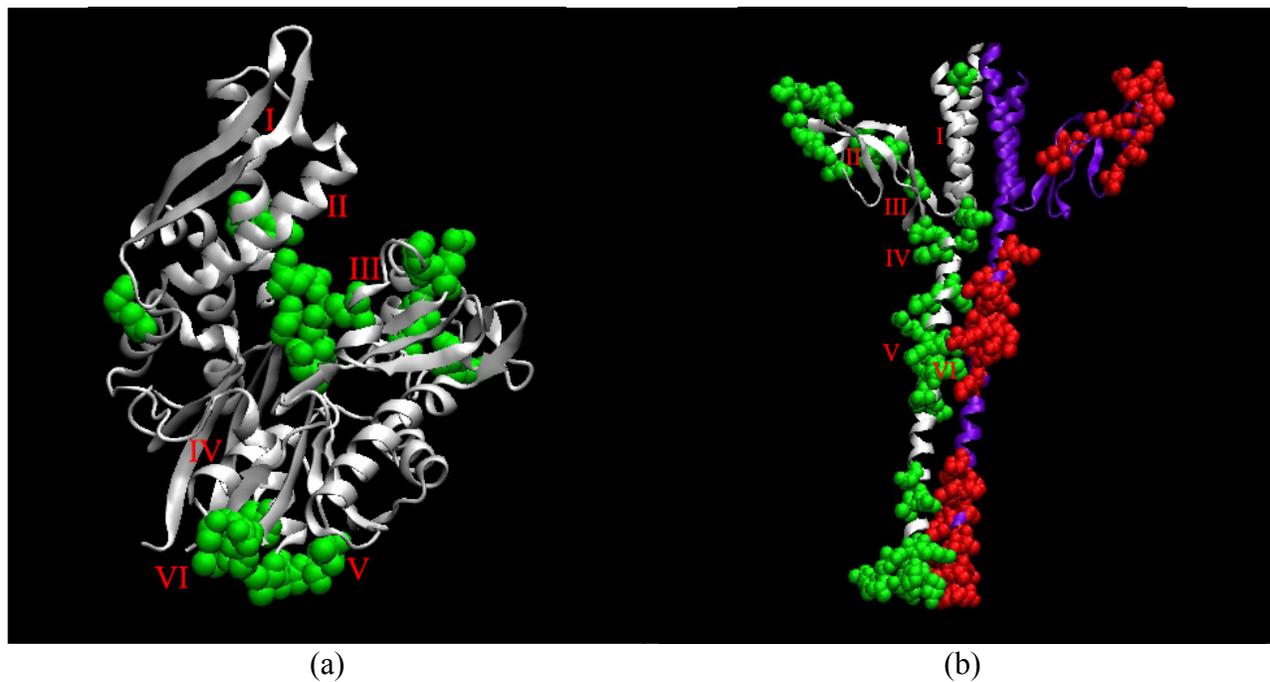


Figure 3.22 Predicted results of three chains in 1DKG using proposed method. (a) DnaK N-terminal chain D of 1DKG, green spheres are predicted interface residues and (b) GrpE, chain A and B of 1DKG. All spheres show predicted atoms.

3.2.5 Further Evaluation on Random Forest Framework

Although leave-one-out cross-validation and blind tests show that the proposed method has ability to make reliable predictions toward interface residues on protein-protein interactions, it is not clear enough what makes the proposed method works better than other two methods.

To figure out the reasons of the improvement, AUCs are compared upon same PSSM feature with three different methods, and the results are shown in Figure 3.23

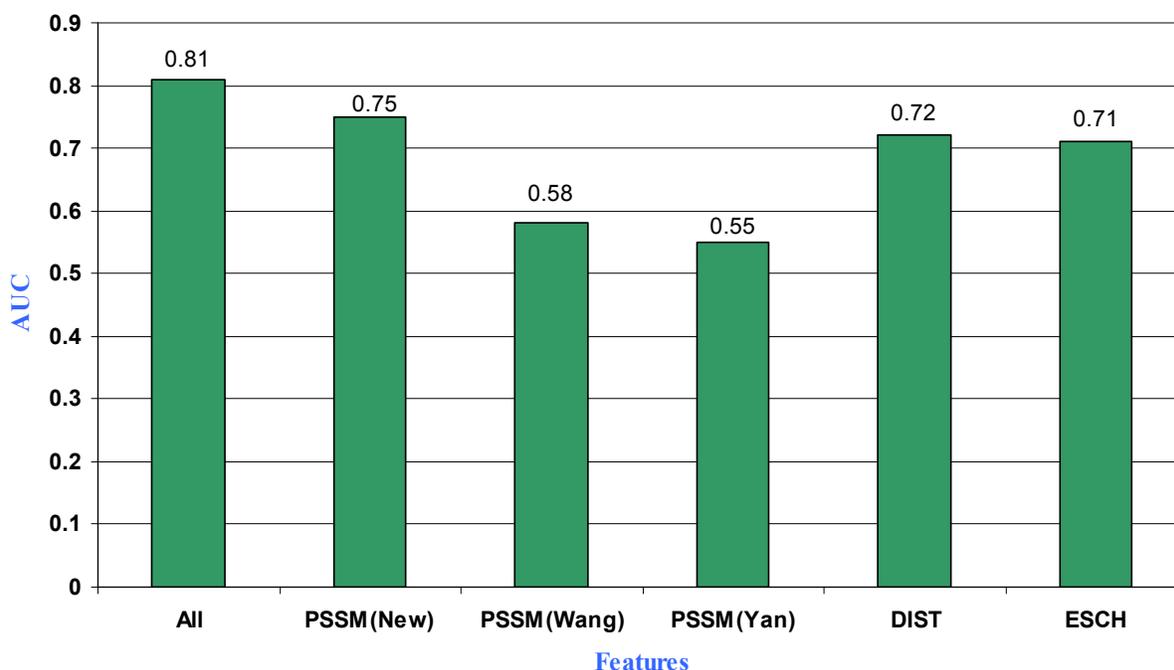


Figure 3.23 Comparing AUC on random forest: PSSM(New) - using PSSM feature only with the proposed method, PSSM(Wang) – using PSSM feature only with Wang’s method, PSSM(Yan) – using PSSM feature only with the Yan’s method, DIST - using distance feature only, ESCH – using the feature of evolutionary scores and physicochemical properties only, and ALL – using all features listed above which is the feature of the proposed method

There are two main reasons choosing PSSM feature as a base feature for comparing three different methods. First, Wang and Yan’s method originally used PSSM as their primary feature and second, the comparisons on individual feature with the proposed method show that PSSM feature outperforms other two feature groups.

By comparing three methods, the differences between two classifiers, SVM and random forests are revealed. Indeed, as mentioned previously, Wang and Yan's method used SVM as their classifier but they introduced different approaches to make final predictions such that Wang's method used five different models trained by five-fold cross-validations, and the final predictions are made by the majority vote among five models; therefore, instead of comparing two basic classifiers, more sophisticatedly manipulated SVMs are compared to the random forest frame work.

Figure 3.23 shows that the AUC of the proposed method outperforms other two methods on PSSM feature. In other words random forests improved performance 17% and 20% of SVMs which are corresponding to Wang's method denoted as 'PSSM(Wang)' and Yan's method denoted as 'PSSM(Yan)' respectively. These results imply that the improvement on the proposed method is considerably affected by choosing a classifier.

However, Figure 3.23 also shows that combining two feature groups, amino acid distance based features (DIST), and evolutionary scores and physicochemical properties (ESCH) also help to improves the performance of the proposed method by enforcing the performance 6% more on PSSM features as a result of comparing two bars 'All' and 'PSSM(New)' in Figure 3.23; therefore, the effects of other two feature groups on improving performance of the proposed method are needed to be further explored.

To see the role of the two feature groups, the margin of decision boundaries between using all features and PSSM feature only is compared. In other words, the differences on the number of trees between positive and negative class on random forests are compared such that the graph in Figure 3.24 shows that the portion of samples corresponding to the differences in the number of predictions between positive and negative votes in random forests.

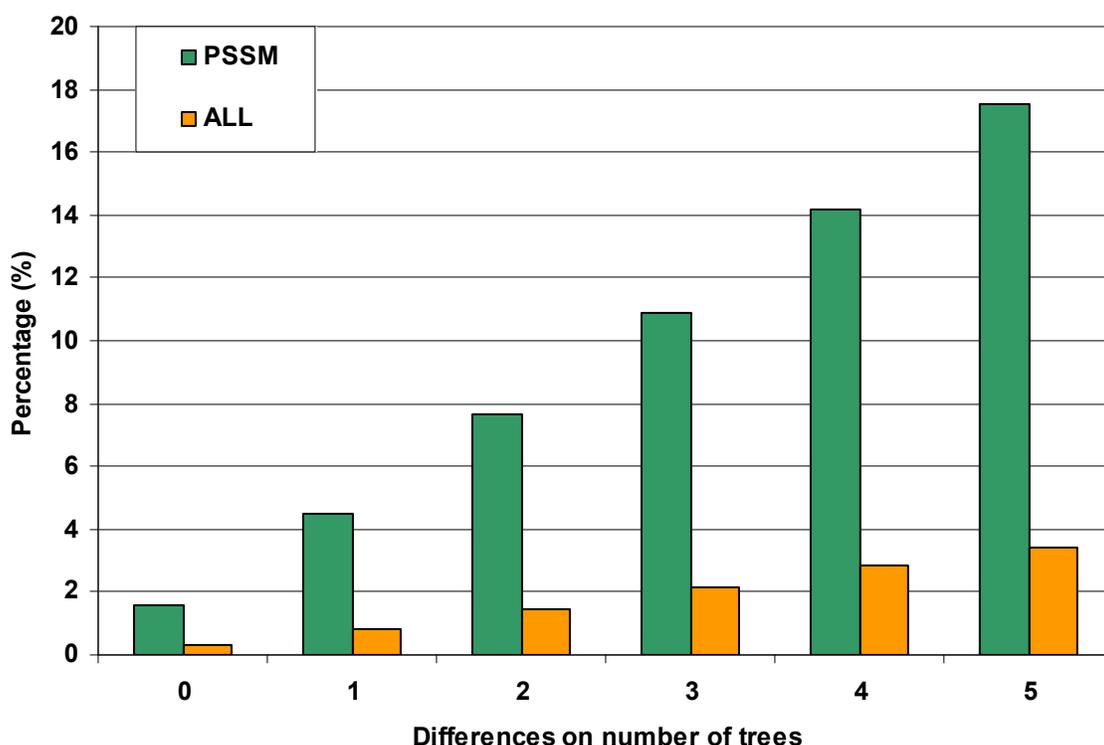


Figure 3.24 Comparing the offset of decision boundary between using all feature and PSSM feature only: the horizontal axis shows the differences on number of trees predicting between interface and non-interface residues, and the vertical axis shows the portion of corresponding residues among entire residues

In the graph, ‘0’ in the axis of ‘Differences on number of trees’ denotes that the difference between positive and negative votes is zero, so the random forest with majority vote cannot decide final predictions for the corresponding samples. In other words, with zero difference, the possibility of a sample to be predicted as positive or negative class is 50 percent so that the final prediction depends on the default rule such that in this experiment, the default rule of proposed method defines a sample as a positive class if the number of vote on positive class is equal or greater than the number of vote on negative class. In the other hand, the vertical axis shows the percentage of samples corresponding to the offset of the number of trees between two classes such that for even votes or zeros tree difference, the performance of the proposed method is affected by about two percent of samples of which their final predictions can be easily

changed by defining default rules. Like the case of zero difference, when five tree differences are considered, the graph shows that 18% samples are corresponding to this category. In contrast to using PSSM feature only, when all features are used, the portion of samples corresponding to the five tree differences is about four percent; therefore, the results show that adding two feature groups helps the proposed method to help reducing the range of ambiguous decision boundary by enforcing predictions on one of the classes.

The results so far show that two important factors that make the proposed methods outperform other two methods are both choosing a classifier (i.e. random forest) and adding new feature groups (i.e. distance and physicochemical properties). However, by considering the framework of random forest as an ensemble method, applying the alternatives of an ensemble method for deciding final predictions into the proposed random forest framework is very interesting since this simple approach can answer two questions whether the performance of the proposed method could be improved, and whether the proposed method is stable on applying different ensemble methods.

For evaluating the potentials of the proposed method, the results of applying four different ensemble methods (i.e. majority vote, weight vote, k-mean ensemble, and SVM stacking ensemble) are compared. The applied four different methods are described below.

Majority vote: it is the simplest ensemble method and used in the proposed method as a default ensemble method such that the final predictions are made by majority of class labels predicted by multiple trees in random forest. In other words, the final class labels are decided by choosing a class that most commonly appeared in the voting pool. The equation of a making final prediction \hat{y} for i^{th} sample is shown below. Here, k is an element of given classes (C) (i.e. interface and non-interface residues), m is the index of trees in vote pool, M is the number of all

trees in the vote pool, and $I(\cdot)_i$ is a step function that if a prediction of m^{th} tree corresponds to class k then returns a value '1' and others '0'.

$$\hat{y}_i = \max_{\arg(k \in C)} \left(\sum_{m=1}^M I(t_m = k)_i \right), \quad i = 1, \dots, n$$

Here,

$$I(t_m = k)_i \begin{cases} 1 : \text{if } t_m = k \text{ for } i^{\text{th}} \text{ sample} \\ 0 : \text{otherwise} \end{cases}$$

(3.3)

Weight vote: majority vote is simply using the mode of predicted class labels in random forest to choose the final prediction. In contrast to the majority vote, weight vote reflects the trust rate of individual vote to make final predictions. The hypotheses on weight vote scheme is that the higher confidence of a tree tends make better decision than trees with lower confidence; therefore, intuitively reflecting the confidence of individual predictors into integrating individual predictors could make better decision. In order to define the confidence rate of individual tree, correlation coefficient is used. To calculate the confidence score, first the predictions on validation data for individual tree are made by using out-of-bag (OOB) algorithm such that the rest of training samples after learning a tree are used as validation dataset. Due to the highly imbalanced data, the structure of validation dataset is also imbalanced. In order to correct an imbalanced problem into a normal problem, the same number of samples for both majority class (non-interface residues) and minority class (interface residues) is randomly extracted. Once the predictions of validation data are made, the class labels between predicted and true class label are used for calculating confidence scores. For the convenience, the predicted labels and true labels of validation data are denoted as \hat{y} and y respectively, and the scoring function of correlation coefficient is shown below.

Correlation coefficient is measuring how two sets of labels between the predicted label set and true label set are correlated. The score is ranged from -1 to 1, so the higher score means that two label vectors between predicted and true class label are highly correlated. In other words, a tree having the higher correlation coefficient has the higher confidence rate, so high weight is assigned to that tree. More precisely, the equation of calculating correlation coefficient is defined in Equation (3.4)

$$f(\hat{\mathbf{y}}, \mathbf{y}) = \frac{\sum_{i=1}^n (\hat{y}_i - \bar{\hat{y}})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (\hat{y}_i - \bar{\hat{y}})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}} \quad (3.4)$$

In Equation (3.4), $\hat{\mathbf{y}}$ and \mathbf{y} denotes a vector of predicted and true class label respectively. n is the total number of residues in validation set, and $\bar{\hat{y}}$ and \bar{y} denotes the mean of predicted and true labels. Once the confidence score of each tree is made, the final prediction is made by choosing the class having maximum sum of weights. The specific equation is shown below. Here, M is total number of trees in a vote pool, k is one of possible classes (i.e. an interface residue and non-interface residue), w_m is a weight for m^{th} tree assigned by Equation (3.4), $t_{m,i}$ is the predicted label made by m^{th} tree for i^{th} residue, and the function $I(t_{m,i} = k)$ is a step function such that if the predicted label is equal to the given label k then return the value 1 otherwise 0.

$$\hat{y}_i = \max_{\arg(k \in C)} \left(\sum_{t=1}^M I(t_{m,i} = k)_i \cdot w_m \right), \quad i = 1, \dots, n$$

Here,

$$w_m = f(\hat{\mathbf{y}}, \mathbf{y})_m \quad (3.5)$$

$$I(t_{m,i} = k)_i \begin{cases} 1 : \text{if } t_{m,i} = k \text{ for } i^{\text{th}} \text{ sample} \\ 0 : \text{otherwise} \end{cases}$$

In short, majority vote is a special case of weight vote such that the majority vote uses a weight '1' equally for all trees, but weight vote assigns a confidence rate as a weight value for individual tree by calculating correlation coefficient between predicted labels and true labels of validation dataset.

k-mean ensemble : The idea behind this method is that increasing the diversity of classifiers can reduce the risk of choosing a wrong classifier such that if classifiers have maximum diversity then the risk of making biased prediction can be minimized since the final prediction is made by the average of maximum diversity classifiers [100-101]. In other words, the average on maximum diversity classifiers can be reached to the global mean of decisions; therefore, the global mean can reduce making biased final decision caused by emphasizing the subset of classifiers, instead of treating all classifiers equally.

To make a diverse distribution for classifiers, k-mean clustering is used since k-mean clustering is capable of redistributing a set of data into expected number (k) of clusters, and by choosing the centroid of each cluster as the representative of each cluster, this can lead the diversity of classifiers as ignoring the original distribution of classifiers and treating each cluster evenly.

For convenience, we now call the predicted labels for validation set used in above *weight vote* ensemble method as classification pattern matrix (CPM) such that CPM is a matrix which is composed of predicted label matrix filled out corresponding to trees and samples. An example of CPM is shown Figure 3.25 in which random forest predicts n samples with M trees (votes).

One of the interesting points on CPM is that the distinctions among different feature groups are omitted such that the validation sets produced from three feature groups is merged

into a CPM. More clearly, for the experiment, the 300 trees from three feature groups are merged into a matrix form, so the total number of trees in CPM is fixed as $M = 300$.

Trees Samples	1 st	2 nd	3 rd	...	M-1 th	M th
Residue 1	1	1	-1	...	-1	-1
Residue 2	-1	1	-1	...	-1	-1
Residue 3	1	-1	1	...	1	1
...
Residue $n-1$	-1	-1	-1	...	1	1
Residue n	1	-1	1	...	-1	1

Figure 3.25 An example of Classification Pattern Matrix (CPM)

For the convenience of explaining k-mean ensemble algorithm, from now on, the trees (row vector) in CPM are considered as samples and the residues (column vector) are considered as features. In other words, the fundamental of this method is that trees are clustered based on their prediction patterns against to residues to construct maximum diversity, and the algorithm is shown below. Although it should be interest whether a specific clustering algorithm can improve the results or not, it is noted that k -mean clustering algorithm alone is evaluated without prior knowledge or validation toward the given dataset. There are two reasons why this is enough in this stage. First this experiment is targeted to see if there is dramatic improvement by changing ensemble method. k -mean is simple and most well known clustering methods and can represent principles of most clustering methods, dense samples within a cluster and sparse samples between clusters. Second, the number of samples in a cluster which consists of number of trees produced by random-forests is very small, and the expected number of samples in a cluster is

very small which should be less than 3; therefore, there will be not much difference on identifying a representative tree in a cluster even if applying different clustering algorithms. The details about applied k -mean algorithm are introduced below.

k-mean ensemble algorithm

CPM : classification pattern matrix

T_j : trees in j^{th} cluster

t_j : a tree representing i^{th} cluster

L : the number of expecting clusters

Step1: run k -mean clustering with CPM
redistributing trees into L clusters

Step2: choose trees representing each cluster

$$t_j = f(\text{centroid}(j), T_j) \quad j=1 \dots L$$

$f(\text{centroid}(j), T_j)$: a function returning a tree which is closest to the centroid of j^{th} cluster

Step3: majority vote

$$\hat{y}_i = \max_{\arg(k \in C)} \left(\sum_{j=1}^L I(t_j = k)_i \right)$$

Here,

$$I(t_m = k)_i \begin{cases} 1 : \text{if } t_m = k \text{ for } i^{\text{th}} \text{ sample} \\ 0 : \text{otherwise} \end{cases}$$

In details of the experiments, to get the maximum diversity among all trees in random forest, 300 trees in which each of three different feature groups produces 100 trees are initially used to choose 100 most diversity trees. Once 100 clusters are defined by k -mean clustering algorithm then 100 trees representing each of 100 clusters are chosen by selecting trees which

are closest to the mean of each cluster. For making the final prediction, the votes from these 100 trees are considered only and all other trees are ignored.

SVM-staking ensemble: staking method integrates multiple models by defining general rules which are learned from the predicted patterns of the classifiers and is well known in machine learning area [102-103]. To make final predictions, the staking method requires two learning steps: *i*) learning from training set and *ii*) learning from predictions from validation set. First step is not much different from the way learning other classifiers except it requires creating multiple models from a training set. There are couple things to create multiple models such that the models are built from a same classifier like random forest, or they can be created from different classifiers. The second step is learning prediction rules to make final predictions. The general rules are built from the prediction tendency of classifiers, and the prediction tendencies are learned from the prediction results of validation set. In other words, the final model learns general rules for model integration by building a new model from the decision patterns of validation sets predicted from individual model trained by training sets. In short, stacking ensemble is building a model which can make final decision by learning the patterns of multiple models, so it is a model's model and they are stacked.

For the experiment, CPM is used as the dataset learning the prediction patterns of multiple models, and SVMs are used as the final model for learning rules of final predictions at the second step due to the good generalization properties of SVM. It is noticed that to avoid complications caused by imbalanced datasets, CPM is defined with balanced validation set, so the distribution of samples in CPM is balanced and considered as normal problem. For the convenience, from now on, this method is called SVM-staking ensemble method, and the algorithm is summarized below together with details of creating CPM.

SVM-staking ensemble algorithm

U : total training samples
 Tr_i : training samples for i^{th} tree in random forest
 Va_i : validation samples for i^{th} tree in random forest (i.e. $Va_i = U - Tr_i$)
 Ts_i : i^{th} testing samples
 CPM_{tr} : classification pattern matrix for training SVM
 CPM_{ts} : classification pattern matrix for final prediction
 F_i : i^{th} model (tree) in random forest
 M_{svm} : the final SVM model
 c_i : final class label for i^{th} testing sample

$CPM_{tr} = \{ \}$;

Step1: training random forests and constructing CPM

DO from $i = 1$ to *the number of trees*

$F_i = \text{Randomforest}(Tr_i)$;

$Va_i = U - Tr_i$

$CPM_{tr} = CPM_{tr} \cup F_i(Va_i)$

END

Step2: Training SVM from CPM_{tr}

$M_{svm} = \text{SVM}(CPM_{tr})$;

Step3: Final prediction

DO from $i = 1$ to *the number testing samples*

$CPM_{ts} = \{ \}$;

DO from $j = 1$ to *the number of trees*

$CPM_{ts} = CPM_{ts} \cup F_j(Ts_i)$

END

$c_i = M_{svm}(CPM_{ts})$

END

Inside algorithm, $\text{Randomforest}(\cdot)$ and $\text{SVM}(\cdot)$ denotes that random forest and SVM are trained with the dataset located inside the parenthesis respectively. Similarly $F_i(\cdot)$ and $M_{svm}(\cdot)$ indicates making predictions with a model from random forest and SVM respectively.

In short, at the first step, this algorithm is learning random forests, and generating CPM by using the samples which are not used for a training model. At the second step, SVM is

learning with CPM. At the final step, CPM for each test sample is generated, and the final prediction is made by feeding the CPM into the model of SVM.

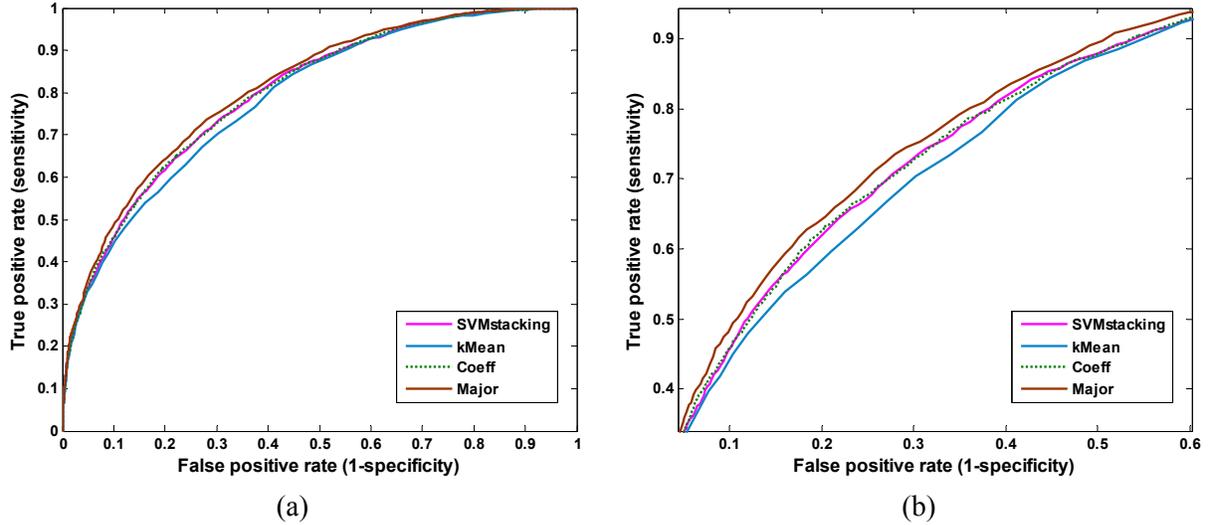


Figure 3.26 Comparing ROC among different ensemble methods: (a) ROCs on final predictions: SVMstacking denotes using stacking method with SVM, kMean denotes using k -mean clustering, Coeff denotes using correlation coefficient for defining weights, and major denotes majority vote for final predictions on random forests, (b) Enlarging the ROC curve (a) between the range of the true positive rate between 0.35 to 0.95

The results of comparing four different assemble methods are shown below. Figure 3.26 (a) shows the ROC curves of four methods: majority vote is the best ensemble method, k -mean clustering ensemble is the worst and weight vote (coefficient) and staking (SVM-staking) are somewhat between them. To see the different more clearly, Figure 3.26 (b) is drawn by enlarging the graph Figure 3.26 (a), and shows that the differences appear between 0.35 and 0.95 of true positive rate. To evaluate the actual differences among four methods, AUC of each method is compared and shown in Figure 3.25. The results show that majority vote (Major) performs best, weight vote (Coeff) performs second best followed by staking method (SVMstacking), and k -mean clustering (kMean) performs worst. The orders made by AUC is shown in the Figure 3.27

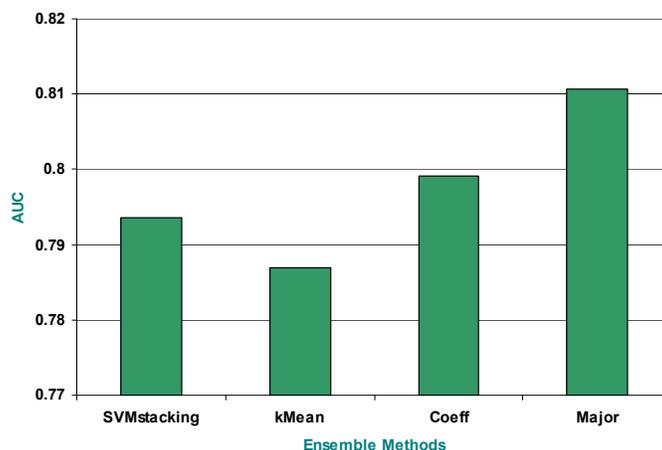


Figure 3.27 Comparing AUC among four different ensemble methods

Although there is an order based on the performances, the difference among them is subtle such that as shown in Figure 3.27, the actual difference between the best (majority vote: Major) and worst ensemble method (*k*-mean clustering: kMean) is less than 3%. By considering the scale of problems and imbalanced data status, it is not fair and cannot be the strong evidence to say that the majority vote outperforms all others including *k*-mean clustering ensemble method.

However, instead of assessing which method performs best, it is more interesting and safe to say that the proposed method gives very reliable and stable performance changes by observing the results of four different ensemble methods. The results can be used as more evidences that the proposed method and features are robust toward predicting protein interface residues, and this study can move one step closer to the practical usage of interface residue identification by solely using amino acid residue sequences.

Chapter4. Conclusion

4.1 Summary of Research

Although current technologies and genome-sequencing projects have rapidly increased the amount of protein sequences, the technology revealing specific characteristics of a protein such as the locations of potential binding sites, function and more precisely binding interface residues are still infancy and require high cost and time consuming processes; therefore, with current wet laboratory techniques, it is almost impossible to identify interface sites from all known proteins. Indeed, current studies show that many diseases are highly related to protein bindings, and by revealing the prosperities of protein bindings and/or identifying specific locations of protein binding can be directly used for developing new treatment. Due to its importance and growing demands, developing a predictive model which can efficiently identify binding residues with high speed and low cost is urgent and of highly interest.

This study investigated various characteristics of amino acid sequences and proposed a new method by integrating multiple features derived directly from amino acid sequences and manipulating machine learning algorithms; therefore, the proposed method are satisfied by both demands, high speed and low cost and are capable of identifying binding sites from current huge amount of protein sequences.

The proposed method requires extracting wide range of features from amino acid sequences and handling different features efficiently in order to accomplish the given tasks. To handle the wide range of features, the principles of ensemble methods were adapted by manipulating random-forests, and the features are partitioned in three categories in order to build basis models: *i*) nine physicochemical properties and evolutionary conservation score, *ii*) amino acid distance profiling, and *iii*) PSSM.

However, the given problem is not favorable for training ordinary machine learning classifiers including random-forests. There are two main problems, first the given dataset is based on protein sequences, so the length of sequences is varied. Data which can be expressed as matrix are the most favorable form of machine learning tasks, so varied length of sequences is very undesirable tasks and requires transforming the information in favorable version. In order to transform the data representation, a method of sliding window was introduced. By applying sliding window, it showed that almost all known characteristics of amino acid residues were easily transformed into matrix like form. The second problems are found in the distributions of data samples. The number of samples in the given problems is highly biased toward the majority class which is non-interface residues in this case. This problem is so called imbalanced problem and known to be very challenging. To solve this imbalanced problem, the method of guiding samples was proposed by manipulating the principles of random-forests.

In order to validate the proposed method, several types of experiments and criterion functions were designed. Interestingly, by considering balanced accuracy, two conventional methods reached close to the originally reported performance in their papers; therefore, this criterion function was not only reasonable but this could be used for representing a model's potentiality toward predictions of interface residues.

The proposed method was compared to two conventional methods. To make effective comparisons, the reinterpreted version of leave-one-out-cross validation (LOOCV) was conducted. The results from LOOCV showed that the proposed method clearly outperformed two other methods in both a direct categorical comparison which was evaluated based on different types of proteins and an overall comparison which was derived from AUC measurements. The tendency of predictions was analyzed and verified with visualized

comparisons. To compare the generalization of derived model, independent datasets of which the structure of the proteins were unknown when the models were originally trained were used for evaluation. The results once again verified that the proposed method outperformed other methods.

To estimate the reliability of the proposed method, blind tests which did not have explicitly known interface residues but they could be implied from literatures were also conducted. For the blind test DnaK chaperone system was considered and the results showed that the proposed method successfully predicted most potential interface residues found in literatures. Besides the literature information, to verify the reliability of the predictions on the blind test, the known structures of individual co-chaperone are further explored. The results showed that the interface residues identified by the proposed method were very reliable.

Once the robustness of the proposed method was verified by multiple steps of evaluations, new questions arose. What made the proposed method outperformed others? Is there any way the performance can be improved? To reveal the good performance, individual features were further explored and compared. The results implied that both the manipulated ensemble method and introduced new features synergically affected to improve the performance of the proposed method.

To improve the performance of the proposed method, various ensemble methods were introduced and tested. Although applying various ensemble methods could not lead significant improvement of the proposed method, the invariant performance against the applied ensemble method implied that the proposed method was very stable and had characteristics independent to designing ensembles.

4.2 Future Work

Although significant investigations are made on proposed methods, some desirable tasks and unanswered questions are still remained. First, both proposed method and conventional method have relatively high false positives and this seems to be from the partner independent training rules such that the predictors are modeled without considering the partner proteins, so that the predicted results are often including other possible binding sites as well. However, it is not easy to derive the features considering binding partners since the sequence information does not have any information about binding directions or angles. For example, let's concatenate any features of two protein pairs. How to define the order of features? This is not a trivial question, since by exchanging the order of protein pairs can significantly affect the distribution of training data; therefore, the produced model should be different.

Second, more generalized and rigorous definitions of interface and non interface residues are required since the proposed method is trained by using the information derived from protein complexes with known structural information. This can lead strongly biased model toward the proteins of known structural information. Indeed, recent studies claim that similar tendencies in cellular component, molecular functions and biological processes can be resulted in a biased prediction such that co-localization of cellular components in proteins of a dataset can result the biased distribution of negative samples and this can lead over-optimistic estimates of classifier accuracy since interacting proteins often participate in similar process and co-localized proteins also likely to participate in similar biological process; therefore, these biased distributions in a dataset make the problem easier and less general [20, 29, 129-131].

Due to both limitations of system resources and time lead somewhat restricted experiments; therefore it is of interest conducting further experiments with more samples and verifying the assumption that increasing samples can lead better performance in specific proteins.

Final and the most interesting task would be constructing strong connection between computational biology and disease treatments. In order to make the connection clearer and stronger, more investigations are required including identification of drug targets and ligand and protein bindings.

References

- [1] A. Vazquez, *et al.*, "Global protein function prediction from protein-protein interaction networks," *Nat Biotechnol*, vol. 21, pp. 697-700, Jun 2003.
- [2] K. A. Pattin and J. H. Moore, "Role for protein-protein interaction databases in human genetics," *Expert Rev Proteomics*, vol. 6, pp. 647-59, Dec 2009.
- [3] A. C. Lewis, *et al.*, "The function of communities in protein interaction networks at multiple scales," *BMC Syst Biol*, vol. 4, p. 100, 2010.
- [4] P. Imming, *et al.*, "Drugs, their targets and the nature and number of drug targets," *Nat Rev Drug Discov*, vol. 5, pp. 821-34, Oct 2006.
- [5] S. K. Kushwaha and M. Shakya, "Protein interaction network analysis--approach for potential drug target identification in Mycobacterium tuberculosis," *J Theor Biol*, vol. 262, pp. 284-94, Jan 21 2010.
- [6] R. Zeidman, *et al.*, "Protein kinase Cepsilon actin-binding site is important for neurite outgrowth during neuronal differentiation," *Mol Biol Cell*, vol. 13, pp. 12-24, Jan 2002.
- [7] Y. Sakaue, *et al.*, "Amelioration of retarded neurite outgrowth of dorsal root ganglion neurons by overexpression of PKCdelta in diabetic rats," *Neuroreport*, vol. 14, pp. 431-6, Mar 3 2003.
- [8] C. Schmitz-Peiffer, *et al.*, "Inhibition of PKCepsilon improves glucose-stimulated insulin secretion and reduces insulin clearance," *Cell Metab*, vol. 6, pp. 320-8, Oct 2007.
- [9] R. M. Kini and H. J. Evans, "A novel approach to the design of potent bioactive peptides by incorporation of proline brackets: antiplatelet effects of Arg-Gly-Asp peptides," *FEBS Lett*, vol. 375, pp. 15-7, Nov 13 1995.

- [10] X. Gallet, *et al.*, "A fast method to predict protein interaction sites from sequences," *J Mol Biol*, vol. 302, pp. 917-26, Sep 29 2000.
- [11] A. F. Fliri, *et al.*, "Drug effects viewed from a signal transduction network perspective," *J Med Chem*, vol. 52, pp. 8038-46, Dec 24 2009.
- [12] T. Niwa, "Elucidation of characteristic structural features of ligand binding sites of protein kinases: a neural network approach," *J Chem Inf Model*, vol. 46, pp. 2158-66, Sep-Oct 2006.
- [13] E. M. Phizicky and S. Fields, "Protein-protein interactions: methods for detection and analysis," *Microbiol Rev*, vol. 59, pp. 94-123, Mar 1995.
- [14] B. W. Morrison and P. Leder, "A receptor binding domain of mouse interleukin-4 defined by a solid-phase binding assay and in vitro mutagenesis," *J Biol Chem*, vol. 267, pp. 11957-63, Jun 15 1992.
- [15] I. Xenarios, *et al.*, "DIP: the database of interacting proteins," *Nucleic Acids Res*, vol. 28, pp. 289-91, Jan 1 2000.
- [16] C. von Mering, *et al.*, "Comparative assessment of large-scale data sets of protein-protein interactions," *Nature*, vol. 417, pp. 399-403, May 23 2002.
- [17] C. M. Deane, *et al.*, "Protein interactions: two methods for assessment of the reliability of high throughput observations," *Mol Cell Proteomics*, vol. 1, pp. 349-56, May 2002.
- [18] P. Legrain and L. Selig, "Genome-wide protein interaction maps using two-hybrid systems," *FEBS Lett*, vol. 480, pp. 32-6, Aug 25 2000.
- [19] R. Mrowka, *et al.*, "Is there a bias in proteome research?," *Genome Res*, vol. 11, pp. 1971-3, Dec 2001.

- [20] A. Ben-Hur and W. S. Noble, "Kernel methods for predicting protein-protein interactions," *Bioinformatics*, vol. 21 Suppl 1, pp. i38-46, Jun 2005.
- [21] R. A. Craig and L. Liao, "Phylogenetic tree information aids supervised learning for predicting protein-protein interaction based on distance matrices," *BMC Bioinformatics*, vol. 8, p. 6, 2007.
- [22] M. Pellegrini, *et al.*, "Assigning protein functions by comparative genome analysis: protein phylogenetic profiles," *Proc Natl Acad Sci U S A*, vol. 96, pp. 4285-8, Apr 13 1999.
- [23] E. M. Marcotte, *et al.*, "Detecting protein function and protein-protein interactions from genome sequences," *Science*, vol. 285, pp. 751-3, Jul 30 1999.
- [24] A. J. Enright, *et al.*, "Protein interaction maps for complete genomes based on gene fusion events," *Nature*, vol. 402, pp. 86-90, Nov 4 1999.
- [25] D. Eisenberg, *et al.*, "Protein function in the post-genomic era," *Nature*, vol. 405, pp. 823-6, Jun 15 2000.
- [26] T. Dandekar, *et al.*, "Conservation of gene order: a fingerprint of proteins that physically interact," *Trends Biochem Sci*, vol. 23, pp. 324-8, Sep 1998.
- [27] J. Tamames, *et al.*, "Conserved clusters of functionally related genes in two bacterial genomes," *J Mol Evol*, vol. 44, pp. 66-73, Jan 1997.
- [28] H. S. Najafabadi and R. Salavati, "Sequence-based prediction of protein-protein interactions by means of codon usage," *Genome Biol*, vol. 9, p. R87, 2008.
- [29] R. Jansen, *et al.*, "A Bayesian networks approach for predicting protein-protein interactions from genomic data," *Science*, vol. 302, pp. 449-53, Oct 17 2003.
- [30] G. Ast, "The alternative genome," *Sci Am*, vol. 292, pp. 40-7, Apr 2005.

- [31] F. Pazos, *et al.*, "Correlated mutations contain information about protein-protein interaction," *J Mol Biol*, vol. 271, pp. 511-23, Aug 29 1997.
- [32] J. R. Bock and D. A. Gough, "Predicting protein--protein interactions from primary structure," *Bioinformatics*, vol. 17, pp. 455-60, May 2001.
- [33] H. J. Hofmann and D. Hadge, "On the theoretical prediction of protein antigenic determinants from amino acid sequences," *Biomed Biochim Acta*, vol. 46, pp. 855-66, 1987.
- [34] T. P. Hopp and K. R. Woods, "Prediction of protein antigenic determinants from amino acid sequences," *Proc Natl Acad Sci U S A*, vol. 78, pp. 3824-8, Jun 1981.
- [35] E. Sprinzak and H. Margalit, "Correlated sequence-signatures as markers of protein-protein interaction," *J Mol Biol*, vol. 311, pp. 681-92, Aug 24 2001.
- [36] U. Gobel, *et al.*, "Correlated mutations and residue contacts in proteins," *Proteins*, vol. 18, pp. 309-17, Apr 1994.
- [37] C. Chothia, *et al.*, "Evolution of the protein repertoire," *Science*, vol. 300, pp. 1701-3, Jun 13 2003.
- [38] R. Jothi, *et al.*, "Co-evolutionary analysis of domains in interacting proteins reveals insights into domain-domain interactions mediating protein-protein interactions," *J Mol Biol*, vol. 362, pp. 861-75, Sep 29 2006.
- [39] R. Apweiler, *et al.*, "The InterPro database, an integrated documentation resource for protein families, domains and functional sites," *Nucleic Acids Res*, vol. 29, pp. 37-40, Jan 1 2001.

- [40] M. Iqbal, *et al.*, "Message-passing algorithms for the prediction of protein domain interactions from protein-protein interaction data," *Bioinformatics*, vol. 24, pp. 2064-70, Sep 15 2008.
- [41] W. K. Kim, *et al.*, "Large scale statistical prediction of protein-protein interaction by potentially interacting domain (PID) pair," *Genome Inform*, vol. 13, pp. 42-50, 2002.
- [42] P. C. Ng and S. Henikoff, "SIFT: Predicting amino acid changes that affect protein function," *Nucleic Acids Res*, vol. 31, pp. 3812-4, Jul 1 2003.
- [43] M. Deng, *et al.*, "Inferring domain-domain interactions from protein-protein interactions," *Genome Res*, vol. 12, pp. 1540-8, Oct 2002.
- [44] R. D. Finn, *et al.*, "The Pfam protein families database," *Nucleic Acids Res*, vol. 36, pp. D281-8, Jan 2008.
- [45] R. Riley, *et al.*, "Inferring protein domain interactions from databases of interacting proteins," *Genome Biol*, vol. 6, p. R89, 2005.
- [46] D. Sprinzak and M. B. Elowitz, "Reconstruction of genetic circuits," *Nature*, vol. 438, pp. 443-8, Nov 24 2005.
- [47] K. S. Guimaraes, *et al.*, "Predicting domain-domain interactions using a parsimony approach," *Genome Biol*, vol. 7, p. R104, 2006.
- [48] X. W. Chen, *et al.*, "Protein function assignment through mining cross-species protein-protein interactions," *PLoS One*, vol. 3, p. e1562, 2008.
- [49] H. Lee, *et al.*, "An integrated approach to the prediction of domain-domain interactions," *BMC Bioinformatics*, vol. 7, p. 269, 2006.

- [50] M. Liu, *et al.*, "Knowledge-guided inference of domain-domain interactions from incomplete protein-protein interaction networks," *Bioinformatics*, vol. 25, pp. 2492-9, Oct 1 2009.
- [51] S. J. Wodak and J. Janin, "Computer analysis of protein-protein interaction," *J Mol Biol*, vol. 124, pp. 323-42, Sep 15 1978.
- [52] J. Warwicker, "Investigating protein-protein interaction surfaces using a reduced stereochemical and electrostatic model," *J Mol Biol*, vol. 206, pp. 381-95, Mar 20 1989.
- [53] B. K. Shoichet and I. D. Kuntz, "Protein docking and complementarity," *J Mol Biol*, vol. 221, pp. 327-46, Sep 5 1991.
- [54] F. Jiang and S. H. Kim, "'Soft docking': matching of molecular surface cubes," *J Mol Biol*, vol. 219, pp. 79-102, May 5 1991.
- [55] P. H. Walls and M. J. Sternberg, "New algorithm to model protein-protein recognition based on surface complementarity. Applications to antibody-antigen docking," *J Mol Biol*, vol. 228, pp. 277-97, Nov 5 1992.
- [56] H. A. Gabb, *et al.*, "Modelling protein docking using shape complementarity, electrostatics and biochemical information," *J Mol Biol*, vol. 272, pp. 106-20, Sep 12 1997.
- [57] P. N. Palma, *et al.*, "BiGGER: a new (soft) docking algorithm for predicting protein interactions," *Proteins*, vol. 39, pp. 372-84, Jun 1 2000.
- [58] H. X. Zhou and Y. Shan, "Prediction of protein interaction sites from sequence profile and residue neighbor list," *Proteins*, vol. 44, pp. 336-43, Aug 15 2001.
- [59] P. Fariselli, *et al.*, "Prediction of protein-protein interaction sites in heterocomplexes with neural networks," *Eur J Biochem*, vol. 269, pp. 1356-61, Mar 2002.

- [60] J. R. Bradford and D. R. Westhead, "Improved prediction of protein-protein binding sites using a support vector machines approach," *Bioinformatics*, vol. 21, pp. 1487-94, Apr 15 2005.
- [61] H. Chen and H. X. Zhou, "Prediction of interface residues in protein-protein complexes by a consensus neural network method: test against NMR data," *Proteins*, vol. 61, pp. 21-35, Oct 1 2005.
- [62] J. L. Chung, *et al.*, "Exploiting sequence and structure homologs to identify protein-protein binding sites," *Proteins*, vol. 62, pp. 630-40, Mar 15 2006.
- [63] UniProt, "The universal protein resource (UniProt)," *Nucleic Acids Res*, vol. 36, pp. D190-5, Jan 2008.
- [64] H. M. Berman, *et al.*, "The Protein Data Bank," *Nucleic Acids Res*, vol. 28, pp. 235-42, Jan 1 2000.
- [65] C. Chothia and J. Janin, "Principles of protein-protein recognition," *Nature*, vol. 256, pp. 705-8, Aug 28 1975.
- [66] C. Chothia, "Hydrophobic bonding and accessible surface area in proteins," *Nature*, vol. 248, pp. 338-9, Mar 22 1974.
- [67] P. Argos, "An investigation of protein subunit and domain interfaces," *Protein Eng*, vol. 2, pp. 101-13, Jul 1988.
- [68] J. Janin, *et al.*, "Surface, subunit interfaces and interior of oligomeric proteins," *J Mol Biol*, vol. 204, pp. 155-64, Nov 5 1988.
- [69] S. Jones and J. M. Thornton, "Analysis of protein-protein interaction sites using surface patches," *J Mol Biol*, vol. 272, pp. 121-32, Sep 12 1997.

- [70] S. Jones and J. M. Thornton, "Prediction of protein-protein interaction sites using patch analysis," *J Mol Biol*, vol. 272, pp. 133-43, Sep 12 1997.
- [71] S. Jones and J. M. Thornton, "Principles of protein-protein interactions," *Proc Natl Acad Sci U S A*, vol. 93, pp. 13-20, Jan 9 1996.
- [72] R. M. Kini and H. J. Evans, "A hypothetical structural role for proline residues in the flanking segments of protein-protein interaction sites," *Biochem Biophys Res Commun*, vol. 212, pp. 1115-24, Jul 26 1995.
- [73] R. M. Kini and H. J. Evans, "Prediction of potential protein-protein interaction sites from amino acid sequence. Identification of a fibrin polymerization site," *FEBS Lett*, vol. 385, pp. 81-6, Apr 29 1996.
- [74] D. Eisenberg, *et al.*, "Analysis of membrane and surface protein sequences with the hydrophobic moment plot," *J Mol Biol*, vol. 179, pp. 125-42, Oct 15 1984.
- [75] D. Eisenberg, *et al.*, "The helical hydrophobic moment: a measure of the amphiphilicity of a helix," *Nature*, vol. 299, pp. 371-4, Sep 23 1982.
- [76] H. De Loof, *et al.*, "Use of hydrophobicity profiles to predict receptor binding domains on apolipoprotein E and the low density lipoprotein apolipoprotein B-E receptor," *Proc Natl Acad Sci U S A*, vol. 83, pp. 2295-9, Apr 1986.
- [77] C. Yan, *et al.*, "Identification of surface residues involved in protein-protein interaction-a support vector machine approach," in *Intelligent Systems Design and Applications*, Tulsa, USA, 2003, pp. 53-62.
- [78] B. Wang, *et al.*, "Predicting protein interaction sites from residue spatial sequence profile and evolution rate," *FEBS Lett*, vol. 580, pp. 380-4, Jan 23 2006.
- [79] L. Breiman, "Random forests," *Machine learning*, vol. 45, pp. 5-32, 2001.

- [80] L. Breiman, "Bagging predictors," *Machine learning*, vol. 24, pp. 123-140, 1996.
- [81] S. Kawashima, *et al.*, "AAindex: amino acid index database, progress report 2008," *Nucleic Acids Res*, vol. 36, pp. D202-5, Jan 2008.
- [82] W. Kabsch and C. Sander, "Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features," *Biopolymers*, vol. 22, pp. 2577-637, Dec 1983.
- [83] B. Rost and C. Sander, "Conservation and prediction of solvent accessibility in protein families," *Proteins*, vol. 20, pp. 216-26, Nov 1994.
- [84] M. Gribskov, *et al.*, "Profile analysis: detection of distantly related proteins," *Proc Natl Acad Sci U S A*, vol. 84, pp. 4355-8, Jul 1987.
- [85] S. F. Altschul, *et al.*, "Gapped BLAST and PSI-BLAST: a new generation of protein database search programs," *Nucleic Acids Res*, vol. 25, pp. 3389-3402, Sep 1 1997.
- [86] M. O. Dayhoff, *et al.*, "A model of evolutionary change in proteins," *Atlas of Protein Sequence and Structure*, vol. 5, pp. 345-352, 1978.
- [87] C. Sander and R. Schneider, "Database of homology-derived protein structures and the structural meaning of sequence alignment," *Proteins*, vol. 9, pp. 56-68, 1991.
- [88] T. F. Smith and M. S. Waterman, "Identification of common molecular subsequences," *J Mol Biol*, vol. 147, pp. 195-7, Mar 25 1981.
- [89] D. Voet and J. G. Voet, *Biochemistry*, 3rd ed. Hoboken, NJ: J. Wiley & Sons, 2004.
- [90] V. N. Vapnik, *Statistical Learning Theory*: John Wiley and Sons, Inc., 1998.
- [91] T. Joachims, "Making large-scale support vector machine learning practical, Advances in kernel methods: support vector learning," ed: MIT Press, Cambridge, MA, 1999.
- [92] C.-C. Chang and C.-J. Lin, "LIBSVM: a library for support vector machines," 2001.

- [93] J. Janin and C. Chothia, "The structure of protein-protein recognition sites," *J Biol Chem*, vol. 265, pp. 16027-30, Sep 25 1990.
- [94] T. K. Ho, "The random subspace method for constructing decision forests," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, pp. 832-844, 1998.
- [95] C. Yan, *et al.*, "Predicting protein-protein interaction sites from amino acid sequence," Department of computer science, Iowa State University, USA, Technical report ISU-CS-TR 02-11 ISU-CS-TR 02-11, 2002.
- [96] C. Chen, *et al.*, "Using random forest to learn imbalanced data," Department of Statistics, UC Berkeley, USA2004.
- [97] X. W. Chen and J. C. Jeong, "Sequence-based prediction of protein interaction sites with an integrative method," *Bioinformatics*, vol. 25, pp. 585-91, Mar 1 2009.
- [98] X. W. Chen and M. Liu, "Prediction of protein-protein interactions using random decision forest framework," *Bioinformatics*, vol. 21, pp. 4394-400, Dec 15 2005.
- [99] M. K. Taghi, *et al.*, "An Empirical Study of Learning from Imbalanced Data Using Random Forest," presented at the Proceedings of the 19th IEEE International Conference on Tools with Artificial Intelligence - Volume 02, 2007.
- [100] T. G. Dietterich, "Ensemble methods in machine learning," *Lecture Notes in Computer Science*, pp. 1-15, 2000.
- [101] L. I. Kuncheva and C. J. Whitaker, "Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy," *Machine Learning*, vol. 51, pp. 181-207, 2003.
- [102] K. M. Ting and I. H. Witten, "Stacking Bagged and Dagged Models," in *14th International Conference on Machine Learning*, 1997, pp. 367-375.

- [103] D. H. Wolpert, "Stacked generalization," *Neural Networks*, vol. 5, pp. 241-259, 1992.
- [104] P. Chakrabarti and J. Janin, "Dissecting protein-protein recognition sites," *Proteins*, vol. 47, pp. 334-43, May 15 2002.
- [105] G. W. Litman, *et al.*, "Phylogenetic diversification of immunoglobulin genes and the antibody repertoire," *Mol Biol Evol*, vol. 10, pp. 60-72, Jan 1993.
- [106] S. R. Neves, *et al.*, "G protein pathways," *Science*, vol. 296, pp. 1636-9, May 31 2002.
- [107] S. Gong, *et al.*, "A protein domain interaction interface database: InterPare," *BMC Bioinformatics*, vol. 6, p. 207, 2005.
- [108] Nguyen, *et al.*, "Protein-Protein Interface Residue Prediction with SVM Using Evolutionary Profiles and Accessible Surface Areas," *Computational Intelligence and Bioinformatics and Computational Biology*, pp. 1-5, 2006.
- [109] J. Thomas, *et al.*, "Optimisation and evaluation of random forests for imbalanced datasets," *Lecture Notes in Computer Science*, vol. 4203, p. 622, 2006.
- [110] N. Japkowicz, "Learning from imbalanced data sets: a comparison of various strategies," presented at the AAAI Workshop on Learning from Imbalanced Data Sets, 2000.
- [111] G. Wu and E. Y. Chang, " Adaptive Feature-Space Conformal Transformation for Imbalanced-Data Learning," in *Proceedings of the Twentieth International Conference on Machine Learning (ICML-2003)*, Washington DC, 2003, pp. 816-823.
- [112] M. Kubat and S. Matwin, "Addressing the Curse of Imbalanced Training Sets: One-Sided Selection," in *Proceedings of 14th International Conference on Machine Learning (ICML'97)*, 1997, pp. 179-186.

- [113] C. Li, "Classifying imbalanced data using a bagging ensemble variation (BEV)," in *Proceedings of the 45th ACM Southeast Regional Conference*, Winston-Salem, North Carolina, USA, 2007, pp. 203-208.
- [114] W. Humphrey, *et al.*, "VMD: visual molecular dynamics," *J Mol Graph*, vol. 14, pp. 33-8, 27-8, Feb 1996.
- [115] Jmol. Dec. 18). *Jmol: an open-source Java viewer for chemical structures in 3D*. Available: <http://www.jmol.org/>
- [116] N. Ban, *et al.*, "Crystal structure of an idiotype-anti-idiotype Fab complex," *Proc Natl Acad Sci U S A*, vol. 91, pp. 1604-8, Mar 1 1994.
- [117] M. A. Wall, *et al.*, "The structure of the G protein heterotrimer Gi alpha 1 beta 1 gamma 2," *Cell*, vol. 83, pp. 1047-58, Dec 15 1995.
- [118] K. Liberek, *et al.*, "Escherichia coli DnaJ and GrpE heat shock proteins jointly stimulate ATPase activity of DnaK," *Proc Natl Acad Sci U S A*, vol. 88, pp. 2874-8, Apr 1 1991.
- [119] K. Liberek and C. Georgopoulos, "Autoregulation of the Escherichia coli heat shock response by the DnaK and DnaJ heat shock proteins," *Proc Natl Acad Sci U S A*, vol. 90, pp. 11019-23, Dec 1 1993.
- [120] C. J. Harrison, *et al.*, "Crystal structure of the nucleotide exchange factor GrpE bound to the ATPase domain of the molecular chaperone DnaK," *Science*, vol. 276, pp. 431-5, Apr 18 1997.
- [121] X. Zhu, *et al.*, "Structural analysis of substrate binding by the molecular chaperone DnaK," *Science*, vol. 272, pp. 1606-14, Jun 14 1996.
- [122] C. S. Gassler, *et al.*, "Mutations in the DnaK chaperone affecting interaction with the DnaJ cochaperone," *Proc Natl Acad Sci U S A*, vol. 95, pp. 15229-34, Dec 22 1998.

- [123] M. K. Greene, *et al.*, "Role of the J-domain in the cooperation of Hsp40 with Hsp70," *Proc Natl Acad Sci U S A*, vol. 95, pp. 6108-13, May 26 1998.
- [124] F. Hennessy, *et al.*, "Analysis of the levels of conservation of the J domain among the various types of DnaJ-like proteins," *Cell Stress Chaperones*, vol. 5, pp. 347-58, Oct 2000.
- [125] W. C. Suh, *et al.*, "Interaction of the Hsp70 molecular chaperone, DnaK, with its cochaperone DnaJ," *Proc Natl Acad Sci U S A*, vol. 95, pp. 15223-8, Dec 22 1998.
- [126] J. E. Davis, *et al.*, "Intragenic suppressors of Hsp70 mutants: interplay between the ATPase- and peptide-binding domains," *Proc Natl Acad Sci U S A*, vol. 96, pp. 9269-76, Aug 3 1999.
- [127] D. L. Montgomery, *et al.*, "Mutations in the substrate binding domain of the Escherichia coli 70 kDa molecular chaperone, DnaK, which alter substrate affinity or interdomain coupling," *J Mol Biol*, vol. 286, pp. 915-32, Feb 26 1999.
- [128] J. Jiang, *et al.*, "Structural basis of interdomain communication in the Hsc70 chaperone," *Mol Cell*, vol. 20, pp. 513-24, Nov 23 2005.
- [129] A. Ben-Hur and W. S. Noble, "Choosing negative examples for the prediction of protein-protein interactions," *BMC Bioinformatics*, vol. 7 Suppl 1, p. S2, 2006.
- [130] P. W. Lord, *et al.*, "Investigating semantic similarity measures across the Gene Ontology: the relationship between sequence and annotation," *Bioinformatics*, vol. 19, pp. 1275-83, Jul 1 2003.
- [131] P. W. Lord, *et al.*, "Semantic similarity measures as tools for exploring the gene ontology," *Pac Symp Biocomput*, pp. 601-12, 2003.