TABLE OF CONTENTS

SPECIAL SECTION ON THE SEVENTH BRAZILIAN SYMPOSIUM ON BIOINFORMATICS
817 Guest Editorial for Special Section on BSB 2012
M.C. P. de Souto and M. Kann

819 Extending the Algebraic Formalism for Genome Rearrangements to Include Linear Chromosomes
P. Feijão and J. Meidanis

832 2D Meets 4G: G-Quadruplexes in RNA Secondary Structure Prediction
R. Lorenz, S.H. Bernhart, J. Qin, C. Hörner zu Siederdissen, A. Tanzer, F. Amman, I.L. Hofacker, and P.F. Stadler

845 Proximity Measures for Clustering Gene Expression Microarray Data: A Validation Methodology and a Comparative Analysis
P.A. Jaskowiak, R.J.G.B. Campello, and I.G. Costa

REGULAR PAPERS

858 A Closed-Loop Control Scheme for Steering Steady States of Glycolysis and Glycogenolysis Pathway
S. Parjá, S. Patra, A. Mukherjee, M. Basu, S. Sengupta, and P.K.Dutta

869 A Divide and Conquer Approach for Construction of Large-Scale Signaling Networks from PPI and RNA Data Using Linear Programming
O.E. Ozsoy and T. Can

884 A Knowledge-Based Multiple-Sequence Alignment Algorithm
K.D. Nguyen and Y. Pan

897 A Two-Phase Bio-NER System Based on Integrated Classifiers and Multiagent Strategy
L. Li, W. Fan, and Q. Huang

905 An Improved Approximation Algorithm for Scaffold Filling to Maximize the Common Adjacencies
N. Liu, H. Jiang, D. Zhu, and B. Zhu

914 An Optimization Rule for In Silico Identification of Targeted Overproduction in Metabolic Pathways
M. Das, C.A. Murthy, and R.K. De

927 Algebraic Representation of Asynchronous Multiple-Valued Networks and Its Dynamics
C. Luo and X. Wang

939 Algorithms for Genome-Scale Phylogenetics Using Gene Tree Parsimony
M.S. Bansal and S. Eustein

957 Analytical Solution of Steady-State Chemical Reaction Networks with Bilinear Rate Laws
A.M. Haláš, H.-J. Lai, M. McCabe Pryor, K. Radhakrishnan, and J.S. Edwards

(Contents continued on inside back cover)
1009 GeneOnEarth: Fitting Genetic PC Plots on the Globe
S. Torres-Sánchez, N. Medina-Medina, C. Gignoux, M.M. Abad-Grau, and E. González-Burchard

1017 Identification of DNA-Binding and Protein-Binding Proteins Using Enhanced Graph Wavelet Features
Y. Zhu, Z. Zhou, D.-Q. Dai, and H. Yan

1032 Pareto Optimality in Organellar Energy Metabolism Analysis
A. Giongo, G. Carapazza, J. Costanta, P. Liu, and G. Nicosia

1045 Protein Function Prediction Using Multilabel Ensemble Classification
G. Yu, H. Hang, C. Domenico, G. Zhang, and Z. Yu

1058 Temporal Logics for Phylogenetic Analysis via Model Checking
J.I. Roquero, D. de Miguel Casado, R. Blanco, and J.M. Colom

SHORT PAPERS

1071 Application of Dempster-Schafer Method in Family-Based Association Studies
F. Rajabi, U. Oktay, and G. Inan

1076 Hamiltonian Walks of Phylogenetic Trees
K. Gordon, E. Ford, and K. John

1086 Call for Papers for Special Issue on Software and Databases

(Contents continued from back cover)
Designing Template-Free Predictor for Targeting Protein-Ligand Binding Sites with Classifier Ensemble and Spatial Clustering

Dong-Jun Yu, Jun Hu, Jing Yang, Hong-Bin Shen, Jinhui Tang, and Jing-Yu Yang

Abstract—Accurately identifying the protein-ligand binding sites or pockets is of significant importance for both protein function analysis and drug design. Although much progress has been made, challenges remain, especially when the 3D structures of target proteins are not available or no homology templates can be found in the library, where the template-based methods are hard to be applied. In this paper, we report a new ligand-specific template-free predictor called TargetS for targeting protein-ligand binding sites from primary sequences. TargetS first predicts the binding residues along the sequence with ligand-specific strategy and then further identifies the binding sites from the predicted binding residues through a recursive spatial clustering algorithm. Protein evolutionary information, predicted protein secondary structure, and ligand-specific binding propensities of residues are combined to construct discriminative features; an improved AdaBoost classifier ensemble scheme based on random undersampling is proposed to deal with the serious imbalance problem between positive (binding) and negative (nonbinding) samples. Experimental results demonstrate that TargetS achieves high performances and outperforms many existing predictors. TargetS web server and data sets are freely available at: http://www.csbio.sjtu.edu.cn/bioinf/TargetS/ for academic use.

Index Terms—Protein-ligand binding sites, ligand-specific prediction model, template-free, classifier ensemble, spatial clustering

1 INTRODUCTION

Protein-ligand interactions are indispensable for biological activities and play important roles in virtually all biological processes [1], [2], [3]. Hence, accurately identifying the protein-ligand binding sites or pockets is of significant importance for both protein function and drug design [4]. Much effort has been made to reveal the intrinsic mechanism of protein-ligand interactions and thousands of protein-ligand complexes have been deposited into protein data bank (PDB) [5]. Due to the importance of protein-ligand interactions and the difficulty of experimentally identifying of protein-ligand binding sites, developing computational methods for the prediction of protein-ligand binding sites using sequence and/or structural information has become a hot spot in recent bioinformatics research [6], [7], [8].

There have emerged many computational methods for predicting protein-ligand binding sites during the past decades [8], [9], [10], [11] and various algorithms for identifying protein-ligand binding sites in proteins have been comprehensively reviewed and explained in detail by Leis et al. [8] and Laurie and Jackson [11]. Roughly speaking, these existing methods can be grouped into three categories according to the features they used [12]: structure-based methods, sequence-based methods, and hybrid methods that utilize both the structural and sequence information.

In the early stage, structure-based methods dominate in the fields of protein-ligand binding sites prediction. To name a few: LIGSITE [13], CASTp [14], SURFNET [15], POCKET [16], f pocket [17], Q-SiteFinder [18], SITEMHOUND [19], and so on. Often, these structure-based predictors try to utilize the protein 3D information and appropriate geometry measurements to locate the potential binding sites (pockets). For example, in LIGSITE [13], a regular 3D grid is placed around the protein; then, lines are drawn from each grid point along the $x$, $y$, $z$-axis as well as the cubic diagonals of the grid; segments of lines that are enclosed by protein from both sides are considered as pockets [8]. In the CASTp [14], alpha shape theory and triangulation methods are used to predict pockets: a Delaunay triangulation of the protein is first performed; then, based on the direction of norm vectors associated with triangles for a set of neighboring triangles, a potential pocket can be detected. Most structure-based methods relay on the assumption that proteins binding similar ligands have similar overall structural or biochemical properties. However, researchers have found proteins that do not display any overall sequence or structure similarity may also present similar binding sites [20]. Therefore, local comparison of binding pockets is a more appropriate approach to predict if two proteins bind similar ligands [21]. For example, Hoffmann et al. [20] proposed a method to quantify the similarity between binding pockets, based on which they can predict binding ligands for a given target pocket by comparing it to an ensemble of pockets with known ligands.
Whereafter, researchers found that features derived from protein sequences can also be effectively used for protein-protein interfaces and protein-ligand binding sites prediction. For example, ConSurf [22] and Rate4Site [23] use the evolutionary data derived from multiple-sequence alignment technique to identify hot spots and surface patches that are likely to be in contact with other proteins, domains, peptides, DNA, RNA, or other ligands; L1pred [24] predicts catalytic residues in enzymes by using the L1-logreg classifier to integrate eight sequence-based scoring functions.

Recently, considerable attention has been paid to the methods that combine both the structural and sequential information to further improve prediction accuracy of protein-ligand binding sites. For example, LIGSITEccc [25] extends the LIGSITE [13] by incorporating the degree of conservation of the involved surface residues, ConCavity [26] integrates evolutionary sequence conservation estimates with structure-based methods for identifying protein surface cavities, and SURFNET-ConSurf [27] also incorporates residue evolutionary conservation into pocket detection.

Although much progress has been made in computational methods for protein-ligand binding sites prediction and many applications based on these methods have emerged, there still exist several issues deserved to be further investigated:

First, many structure-based methods are template-based that require tertiary protein structures as inputs to search for existing well-characterized proteins as templates for homology comparative prediction [9], [10], [28], [29]. However, on the one hand, it is very common in many realistic scenarios (e.g., drug design project) where a given protein target only has sequence information and no corresponding 3D structure is available [8], thus the applicability of these structure-based methods will be limited. On the other hand, these template-based methods will also fail to perform novel predictions for features absent in the template library [2]. These are two of the major reasons that motivate researchers in this field to develop useful methods for predicting protein-ligand binding sites from the protein sequence information alone.

Second, a common drawback of most existing sequence-based protein-ligand binding prediction methods (e.g., ATPsite [7] and NsitePred [30] for ATP binding residue prediction; DiANNA [31] for ternary cysteine prediction: discriminating between free, metal binding, disulphide binding for cysteines [32], etc.) is that they are binding state prediction [33], i.e., they can only predict the protein-ligand binding residues from sequences and cannot tell which residues may potentially form the binding sites (pockets). It is believed that a deep understanding of the mechanism of protein-ligand interaction requires more than this simple two-state or three-state prediction and thus developing effective methods for further identifying the binding sites (pockets) from the predicted binding residues is necessary.

Third, most of the aforementioned methods focused on predicting general-purpose ligand binding sites without carefully considering the differences between various ligands. In fact, protein binding sites vary significantly in their roles, sizes, and distributions for different types of protein-ligand interactions and different ligands tend to binding to different residues with specificities [34], [35]. Considering the significant difference between different types of ligands, developing ligand-specific binding sites predictor has attracted considerable attentions to obtain much more accurate predictions, and many ligand-specific binding sites predictors have emerged recently [36]. For example, Liu et al. developed HemeNet [37] and HemeBIND [12] for specifically predicting HEME binding residues based on structural and sequential information, Sodhi et al. [38] exploited neural network methods to predict metal ions binding sites, Brylinski and Skolnick [39] extended the FINDSITE software to FINDSITE-metal for specifically predicting metal ions binding sites, and Kumar et al. [40] developed Pprint, an RNA binding site predictor using SVM and PSSM profile; several predictors specifically designed for Adenosine-5’-triphosphate (ATP) binding residues prediction have also been released [6], [7], [30] recently. In this study, we will experimentally demonstrate that ligand-specific binding sites predictor is better than general-purpose ligand binding sites predictor and really helps to further improve prediction accuracy.

In view of the above-mentioned three important issues, we thus developed a ligand-specific template-free protein-ligand binding sites predictor, called TargetS, which currently can predict binding sites for 12 types of ligands and is flexible to incorporate prediction modules for other new types of ligands. Protein conservation matrix, predicted protein secondary structure matrix, and ligand-specific binding propensities of residues are combined to extract discriminative features; classifier ensemble with SVM [41], [42] as base classifier and random undersampling technique are integrated to tackle the serious imbalance phenomenon between negative and positive samples (i.e., nonbinding and binding residues). TargetS performs prediction task with a two-stage scheme: In the first stage, it predicts which residues are protein-ligand binding residues with ligand-specific prediction modules, while in the second stage, the predicted binding residues are spatially clustered into the binding sites (pockets) according to the protein 3D structures either provided by the user or modeled by the MODELLER software [43]. The TargetS is freely accessible at http://www.csbio.sjtu.edu.cn/bioinf/TargetS/ for academic use.

2 MATERIALS AND METHODS

2.1 Benchmark Data Sets

Most ligand-binding sites prediction methods use the protein structures from the PDB [44] as templates [45]. However, not all ligands present in the PDB are biologically relevant, as small molecules are often used as additives for solving the protein structures [46], [47]. Much effort has been made to filter out the biologically ligand-protein interaction from the PDB and several purified ligand-protein interaction data sets have appeared, such as FireDB [47], LigASite [46], PDBbind [48], and BioLip [49], among which BioLip is the most recently released semimanually curated database for biologically relevant ligand-protein interactions. BioLip was constructed by a four-step biological feature filtering procedure followed by careful manual verifications [45]: First, an automated four-step hierarchical procedure is used to verify the biological relevance of a ligand. After the
automated procedure is completed, a careful manual check is performed to eliminate possible false positives, which can occur for entries with the commonly used crystallization additives. By doing so, it is believed with high confidence that the ligand-protein interactions collected from PDB are real biologically relevant. Details for constructing BioLip can be found in [45].

To evaluate the effectiveness of the proposed TargetS, we thus constructed training data sets and independent validation data sets based on the BioLip [45] rather than on PDB. Twelve different types of ligands, i.e., five types of metal ions, five types of nucleotides, DNA, and HEME, were considered in this study. For each of the 12 types of the considered ligands, we constructed its training data set and independent validation data set as follows:

**Training data sets.** We extracted all the protein sequences, which interact with the given ligand and were released into PDB before 10 March 2010, from BioLip, and then the maximal pairwise sequence identity of the extracted protein sequences was culled to 40 percent with PISCES software [49] and the resulting sequences constitute the training data set for that ligand.

**Independent validation data sets.** We extracted all the protein sequences that interact with the ligand and were deposited into PDB after 10 March 2010, from BioLip, and then the maximal pairwise sequence identity of the extracted protein sequences was reduced to 40 percent and the resulting sequences constitute the validation data set. Moreover, if a given sequence in the validation data set shares \( >40\% \) identity to a sequence in the training data set, then we remove the sequence from the validation data set. This assures that the sequences in validation data set are independent of those in training data set. Table 1 summarizes the detailed compositions of the training data sets and the independent validation data sets for the 12 types of ligands.

To further demonstrate the effectiveness of the proposed TargetS, CASP9 data set was used for blind test. The ninth community-wide critical assessment of techniques for protein structure prediction (CASP9) released 129 target protein sequences for blind test of protein structure and function prediction methods. Among the 129 sequences, 31 were used for evaluating the ligand binding-site predictions, where the predictors were asked to identify ligand binding residues in the sequences. As one sequence (Target ID: T0533) was canceled on 26 May 2010, the remaining 30 sequences were, thus, taken as targets for our consideration.

It has not escaped from our notice that the percentages of binding residues in training and validation data sets for a given ligand are different. However, this difference will not affect the objective evaluation procedure of the proposed method as we performed both the cross-validation evaluation on training data set and the independent test on the testing data set. The purpose of the cross-validation is to evaluate the overall performance of the proposed method on a given data set. While independent test is often used to evaluate the generalization capability of the proposed method, which has been widely accepted in this field.

### Feature Extraction

#### 2.2.1 Position Specific Scoring Matrix Feature

Position specific scoring matrix (PSSM) well encodes the evolutionary information of a protein sequence. Tremendous previous studies have shown its prominent discriminative capability for many prediction problems in bioinformatics, such as protein function prediction [50], protein-ATP binding sites prediction [51], transmembrane helices prediction [52], protein secondary structure prediction [53], subcellular localization prediction [54], [55], [56], and so on.

The position specific scoring matrix for protein sequence is built by using the PSI-BLAST [57] to search the Swiss-Prot database through three iterations with 0.001 as the e-value cutoff for multiple sequence alignment against the query sequence.
sequence. The obtained PSSM is further normalized with the logistic function:

\[ f(x) = \frac{1}{1 + \exp(-x)} \]  

where \( x \) is the original score in PSSM matrix. Then, a sliding window of size \( W \) is applied to each residue to extract its PSSM feature vector. In this study, we have tested different values of \( W \) and found that \( W = 17 \) is a better choice. Thus, the dimensionality of PSSM feature vector is \( 17 \times 20 = 340 \).

2.2.2 Predicted Protein Secondary Structure Feature

Previous studies have shown that there exists a close relationship between protein structures and functionalities. Many structural characteristics related to identification of critical residues (e.g., ligand-binding residues) have been intensively investigated, such as the secondary structure information and so on. [51], [58]. The predicted secondary structure information of a protein sequence is obtained by applying PSIPRED [59] software, which predicts the probabilities of belonging to three secondary structure classes (coil (C), helix (H), and strand (E)) for each residue in a protein sequence. More specifically, for a protein sequence with \( L \) residues, the PSIPRED outputs an \( L \times 3 \) probability matrix, which represents the predicted secondary structure information of the protein. Again, a sliding window of size 17 was used to extract protein secondary structure feature, denoted as PSS, for each residue and the dimensionality of the extracted PSS feature is \( 17 \times 3 = 51 \).

2.2.3 Ligand-Specific Binding Propensity Feature

Previous studies have shown that different ligands tend to bind different types of residues and this binding propensity may potentially help to improve the binding sites prediction accuracy [34], [35]. For example, Gromiha and Fukui [60] analyzed the binding propensity among protein-DNA complexes and found that positively charged, polar, and aromatic residues are important for binding. In light of this, we will also incorporate binding propensity feature in our method. We calculated the frequencies of the 20 native amino acids among binding residues for each type of ligands and found that the ligands clearly has binding propensity. Taking metal ion ligands as example, for each of the five considered metal ion ligands, we calculated and plotted the frequencies of the 20 native amino acids among binding and nonbinding residues, as shown in Fig. 1. By observing Fig. 1, several observations can be drawn:

First, each type of metal ion ligands favors to bind some specific types of residues. For example, ASP (D), GLU (E), ASN (N), and GLY (G) are the top four types of residues to which \( \text{Ca}^{2+} \) tends to bind, while HIS (H), CYS (C), ASP (D), and GLU (E) are the top four types of residues which will more often coordinate the \( \text{Zn}^{2+} \). This observation agrees with the arguments obtained by other researchers such as Lu et al. [61]. Second, some types of residues will always appear with high frequencies for different types of metal ion-binding residues. It is easy to find that the residues ASP (D), and GLU (E) consistently appear with high frequencies among all the five types of metal ion-binding residues. These statistical results inspired us that the binding propensities of residues can be utilized as effective indexes to guide the identification process of protein-ligand binding. Together with the fact that the binding propensities of a same residue differ significantly for different ligands, we thus extract ligand-specific 17D binding propensity feature vector for each residue in a protein sequence by concatenating the binding propensities of its neighboring residues within the window of size 17 centered at the residue.

2.3 Application of Machine Learning Methods

Protein-ligand binding prediction is a typical imbalanced learning problem, i.e., the numbers of samples in different classes (binding or nonbinding) differ significantly. Directly applying the traditional statistical machine learning algorithms, which assume that samples in different classes are balanced, to imbalanced problems often leads to a poor performance [62]. To circumvent this problem, random undersampling technique is taken [63] to alter the size of the majority class by randomly removing samples from the
Majority class. Random undersampling can provide a parsimonious training data set because it removes samples from the original data set. However, part of the important information buried in the removed samples may also be lost simultaneously. We, thus, exploited the method of combing multiple undersamplings with classifier ensemble to perform protein-ligand binding prediction: first, we sample $L$ different majority training subsets by random undersampling the majority class $L$ times; then, we train a base classifier on each of the majority training subsets plus the minority training set. By doing so, on the one hand, merits derived from undersampling such as sample balance can be retained. On the other hand, multiple samplings can reduce the loss of information caused by random undersampling to some extent, thus may potentially provide better prediction performance. Finally, the trained base classifiers are ensembled by an appropriate ensemble scheme to perform the final decision. Fig. 2 illustrates the architecture of the classifier ensemble procedure based on random undersampling technique. In this study, support vector machine (SVM) [41], [42] was used as base classifier.

The next key problem is how to achieve best final decision from the outputs of the $L$ base classifiers with an effective ensemble scheme. Classifier ensemble has been widely applied in bioinformatics, such as prediction of all-alpha membrane proteins [64], protein fold prediction [65], [66], protein subcellular localization prediction [54], and protein structural class prediction [67] and so on. However, ensemble scheme are problem-dependent and the theoretical justification for choosing is still unavailable. In view of this, in this study, we have tested several popular ensemble schemes including Maximum ensemble, Minimum ensemble, Mean ensemble, Dempster-Shafer ensemble [68], AdaBoost ensemble [69], and Decision Template ensemble [70]. The best one, i.e., AdaBoost, was finally chosen.

The idea of AdaBoost ensemble is to develop the classifier team $S^E = \{S_1^E, S_2^E, \ldots, S_{K}^E\}$ by incrementally selecting one base classifier each time from the base classifier pool $S = \{S_1, S_2, \ldots, S_L\}$, $K \leq L$ and $S_k^E \in S$. The base classifier $S$, that joins the ensemble at step $i$ is trained on a training subset selectively sampled from the training data set $X^{Tr} = \{x_i^{Tr}\}_{i=1}^N$ by applying a sample-distribution-based sampling technique. Details for AdaBoost could be found in [69] or [70].

However, two reasons motivate us to develop a modified AdaBoost ensemble scheme, denoted as MAdaBoost, rather than utilizing the traditional AdaBoost ensemble directly:

1. In the traditional AdaBoost, samples (binding residues and nonbinding residues in this study) in the whole training data set are used as evaluation samples to calculate the ensemble error of each base classifier. In other words, evaluation samples and training samples originate from the same protein sequence thus having high homology, which will lead to an overoptimistic performance on training data while having poor generalization performance on the testing data. In view of this, we will utilize and independent evaluation data set, among which samples have low homology with those in training data set, to facilitate the ensemble procedure. How to choose the independent evaluation data set will be described subsequently.

2. To reduce the impact of the serious imbalance between positive (binding) and negative (non-binding) samples, we will still use random undersampling technique to balance samples for training the base classifiers during the ensemble procedure.

Let $\Omega = \{\omega_1, \omega_2, \ldots, \omega_C\}$ be the set of classes, $S = \{S_1, S_2, \ldots, S_L\}$ be the set of $L$ base classifiers which are trained on a corresponding data set randomly undersampled from training data set $X^{Tr} = \{x_i^{Tr}\}_{i=1}^N$. For a given input sample $x$, the $i$th base classifier $S_i$ outputs a $C$-dimensional vector $S_i(x) = (s_{i,1}(x), s_{i,2}(x), \ldots, s_{i,C}(x))$, where $s_{i,j}(x)$ measures the probability of $x$ being classified into class $j$, $1 \leq j \leq C$. Let $X^{Eval} = \{x_i^{Eval}\}_{i=1}^M$ be the independent evaluation data set, the two-tuple $(S^E, \{\varepsilon_k^E\}_{k=1}^K)$ be the ensembled results, where $S^E = \{S_1^E, S_2^E, \ldots, S_{K}^E\}$ consists of the base classifiers selected from the base classifier pool and $\varepsilon_k^E$ is the corresponding weighted ensemble error of the $k$th selected base classifier. Then, the flowchart of the proposed modified AdaBoost (MAdaBoost) procedure can be illustrated in Fig. 3.

We use $V$-fold cross-validation to evaluate the performance of the proposed MAdaBoost ensemble as follows:

First, we randomly partition the sequences rather than the residues of sequences in the original training data set into $V$ disjoint subsets. By partitioning the original training data set on sequence rather than residue level, the low homology between evaluation samples and training samples during ensemble procedure can be guaranteed.

Second, residues of sequences in one subset constitute the testing data set $X^T$ in the current round of cross validation, residues of sequences in one subset are used to construct independent evaluation data set $X^{Eval}$, and the residues of sequences in the remaining $V-2$ subsets are used as training data set $X^{Tr}$. $X^T$ and $X^{Eval}$ are then used for classifier ensemble with the MAdaBoost ensemble scheme (see Fig. 3). After classifier ensemble, samples in $X^T$ are fed to the ensembled classifier $(S^E, \{\varepsilon_k^E\}_{k=1}^K)$ to obtain their probability outputs: for a sample $x$ in $X^T$, its support for class $\omega_j$ ($1 \leq j \leq C$) obtained from the ensembled classifier $(E, \{\varepsilon_k^E\}_{k=1}^K)$ is formulated as
YU ET AL.: DESIGNING TEMPLATE-FREE PREDICTOR FOR TARGETING PROTEIN-LIGAND BINDING SITES WITH CLASSIFIER ENSEMBLE...

**Input:**

\[ X^T = \{x^T_i\}_{i=1}^n \text{: Training dataset} \]

\[ X^{\text{indep}} = \{x^{\text{indep}}_i\}_{i=1}^m \text{: Independent evaluation dataset} \]

\[ L \text{: Number of base classifiers} \]

**Ensemble:**

\[ w_i \leftarrow \frac{1}{n}, \quad i \leftarrow 1, \quad k \leftarrow 1 \]

\[ w_i \leftarrow \frac{1}{n}, \quad i \leftarrow 1, \quad k \leftarrow 1 \]

1. Random under-sampling
2. Train classifier \( S_k \) on \( X^T \)
3. Calculate the weighted ensemble error \( \varepsilon \)
   \[ \varepsilon = \frac{1}{K} \sum_{k=1}^{K} \sum_{i=1}^{n} w_i \left| \left( \hat{y}_i - y_i \right) \right| \]
   where \( \hat{y}_i \) is the predicted label and \( y_i \) is the true label.
4. For \( k = 1 \) to \( K \):
   - \( S_k = S_k(x) \)
   - \( S_k = S_k(x) \)
   - \( S_k = S_k(x) \)
   - \( S_k = S_k(x) \)
5. Return the ensembled classifier \( S^E \)

**Output:**

**Fig. 3.** Flowchart of the proposed modified AdaBoost (MAdaBoost).

\[ \mu_j(x) = \sum_{1 \leq k \leq K} \text{label}(S_k^E(x)) \omega_k \left( 1 - \varepsilon_k^E(x) \right) \times s_k^E(x) \]
\[ + \sum_{1 \leq k \leq K} \text{label}(S_k^E(x)) \omega_k \varepsilon_k^E \times s_k^E(x), \]

where \( \varepsilon_k^E \) is the weighted ensemble error of the base classifier \( k \) in the ensembled classifier team, \( S_k^E(x) = (s_{k1}^E(x), s_{k2}^E(x), \ldots, s_{kC}^E(x))^T \), \( \text{label}(S_k^E(x)) = \omega_j \)

**Third, this practice continued until all the \( V \) subsets of the original training data set are traversed over for testing.**

After \( V \)-fold cross-validation, the probabilities of being binding residues of all the residues in the original training data set can be obtained. Further, the prediction performance of proposed method on training data set over \( V \)-fold cross-validation can be calculated by setting a threshold, i.e., residues with probabilities above the threshold are marked as ligand binding residues, while those ones with probabilities below the threshold are marked as non-binding residues. How to choose the threshold \( T \) for reporting the performance will be described in the experimental results section.

**2.4 Clustering Binding Residues to Binding Site(s)**

As stated in Section 1, most of the existing sequence-based protein-ligand binding predictors can only perform binding state prediction, i.e., they can only predict protein-ligand binding residues rather than binding sites from sequences. In fact, it will be more useful for biologists and users if the predictor can tell which binding residues actually form a binding site (pocket), especially in the situation where there exist more than one binding sites (pockets) in one protein sequence.

Previous studies have shown that residues located in protein-ligand binding interfaces tend to form spatial clusters [71]. Taking chain A of protein 1L2T (interacts with ligand ATP) and chain A of protein 1A29 (interacts with ligand Ca\(^{2+}\)) as examples, we drew their 3D structures with cartoon representation as shown in Fig. 4, where the blue and red residues are observed ligand-binding residues, and ligands (ATP and Ca\(^{2+}\)) are highlighted in yellow color. From Fig. 4, it is easy to find that the residues colored in blue and red are spatially clustered and form binding sites 01 and 02, respectively, for both proteins.

Based on this observation, we thus have developed a recursive spatial clustering procedure, denoted as RSCP, which has been demonstrated effective in our recent work for protein-ATP binding site prediction [72], to further identify which of the predicted binding residues may potentially form binding site(s).

Let \( PBR \) be the set of the predicted binding residues for a given protein, \( T_{\text{ligand}} \) be the ligand-specific threshold for spatial clustering, and \( P_{\text{3D}} \) be the 3D structure (predicted or observed) of the protein. Then, the predicted binding residues in \( PBR \) can be clustered into predicted binding sites, denoted as \( PBS \), using the following recursive spatial clustering procedure, as shown in Fig. 5.

Note that in the spatial clustering procedure, the clustering threshold \( T_{\text{ligand}} \) is ligand-dependent and dominates how many clusters (binding sites) will be obtained after spatial clustering. Obviously, a large \( T_{\text{ligand}} \) will produce small number of clusters, while a small \( T_{\text{ligand}} \) will lead to a large number of clusters. Thus, how to set an appropriate \( T_{\text{ligand}} \) is crucial for the spatial clustering procedure. In this study, the ligand-dependent clustering threshold \( T_{\text{ligand}} \) was obtained as follows: For each type of ligand, we performed spatial clustering procedures on the observed binging residues in the corresponding training data set for that ligand with different clustering thresholds (\( T_{\text{ligand}} \)) by varying the values of \( T_{\text{ligand}} \) from 10 to 100 with a step size of 0.5, and the one which maximizes the clustering accuracy was chosen as the final threshold for that ligand.

Two measures were used to evaluate the performance of clustering algorithm and optimize the clustering parameters.
on the training data sets. The first measure is $V_{site}$, which measures the percentage of the observed binding sites that have been correctly clustered. In this study, an observed binding site is considered to be correctly clustered if its 90 percent binding residues are included in the clustered binding site. The second measure is $V_p$, which measures the percentage of proteins in the training data set that have been correctly clustered. A protein is considered being correctly clustered if all the binding sites in this protein are correctly clustered and the number of the clustered binding sites is equal to the number of the observed binding sites in the protein. Table 2 lists the optimal clustering thresholds for the 12 types of ligands after performing the above-mentioned empirical evaluations on their corresponding training data sets. As we can see that generally the larger of the ligand, the bigger of the threshold will be.

2.5 Designing Ligand-Specific Prediction Model

Many existing protein-ligand prediction methods focused on predicting general ligand binding sites without carefully considering the differences in various ligands [12]. In fact, protein-ligand binding sites vary significantly in their roles, sizes, and distributions for different types of protein-ligand interactions [34]. For example, we calculated that the averaged diameter of nucleotide ligands is about 22 Å, while the averaged diameter of metal ion ligands is only about 2 Å. We also calculated the averaged size of binding site (averaged number of residues in each binding site/pocket) for the 12 types of considered ligands as listed in Table 3. It is easy to find that the averaged size of binding site differs significantly between different ligand categories. For example, the averaged size of binding site for nucleotide category (ATP, AMP, ADP, GDP, and GTP) is about 12, while the averaged size of binding site for metal ion category (Ca$^{2+}$, Mg$^{2+}$, Mn$^{2+}$, Fe$^{3+}$, and Zn$^{2+}$) is only about 3, the difference between them is approximately 9; on the other hand, the averaged sizes are similar among the same ligand category. Taking metal ion category as an example, the maximal and minimal averaged sizes of the five considered ligands are 2.95 and 3.80, respectively, and the difference is only about 0.85. Similar phenomenon can also be found in nucleotide category. In addition, different ligands (e.g., Ca$^{2+}$, Mg$^{2+}$, Mn$^{2+}$, Fe$^{3+}$, and Zn$^{2+}$) are also found to tend to bind different residues even if they belong to the same ligand category (e.g., metal ion) which we have discussed in Section 2.2.

In view of the above observations, we believe and will demonstrate that developing ligand-specific protein-ligand binding sites predictor, i.e., designing a specific model for each type of ligands, may help to further improve the prediction accuracy.

2.6 Workflow of the Proposed Predictor

We designed and implemented a ligand-specific template-free predictor for targeting protein-ligand binding sites, called TargetS, with modular design strategy to facilitate it with good flexibility and scalability. Fig. 6 illustrates the workflow of the proposed TargetS.

Table 3. It is easy to find that the averaged size of binding site differs significantly between different ligand categories. For example, the averaged size of binding site for nucleotide category (ATP, AMP, ADP, GDP, and GTP) is about 12, while the averaged size of binding site for metal ion category (Ca$^{2+}$, Mg$^{2+}$, Mn$^{2+}$, Fe$^{3+}$, and Zn$^{2+}$) is only about 3, the difference between them is approximately 9; on the other hand, the averaged sizes are similar among the same ligand category. Taking metal ion category as an example, the maximal and minimal averaged sizes of the five considered ligands are 2.95 and 3.80, respectively, and the difference is only about 0.85. Similar phenomenon can also be found in nucleotide category. In addition, different ligands (e.g., Ca$^{2+}$, Mg$^{2+}$, Mn$^{2+}$, Fe$^{3+}$, and Zn$^{2+}$) are also found to tend to bind different residues even if they belong to the same ligand category (e.g., metal ion) which we have discussed in Section 2.2.

In view of the above observations, we believe and will demonstrate that developing ligand-specific protein-ligand binding sites predictor, i.e., designing a specific model for each type of ligands, may help to further improve the prediction accuracy.

2.6 Workflow of the Proposed Predictor

We designed and implemented a ligand-specific template-free predictor for targeting protein-ligand binding sites, called TargetS, with modular design strategy to facilitate it with good flexibility and scalability. Fig. 6 illustrates the workflow of the proposed TargetS.

The TargetS server accepts two different types of query protein information for protein-ligand binding sites prediction: one is protein sequence in FASTA format; the other is standard PDB file format, which contains 3D structure information of a protein. Note that if the user inputs a PDB file, corresponding sequence will be picked out from the PDB file for subsequent sequence-based feature extraction (e.g., PSSM and PSS). For each protein (sequence or PDB file) submitted from the client, the server performs binding sites predictions for different types of ligands the user designated by using corresponding ligand-specific prediction models. Note that if the user designate “I don’t know the ligand types,” the TargetS will perform predictions for all the 12 types of ligands, respectively. For each type of ligand, TargetS accomplishes the prediction task with a two-stage scheme: In the first stage, the server predicts which residues are binding residues, while in the second stage, the server further identifies binding sites from the predicted binding residues with spatial clustering algorithm.

Note that if the user submits a PDB file, then the residues’ 3D coordinates contained in the PDB file can be directly utilized for spatial clustering. If the user only submits a protein sequence, the 3D structure of the query

---

**Fig. 5. Spatial clustering procedure for clustering binding residues to binding sites.**

**Table 2.** Thresholds for Spatial Clustering Procedure of the 12 Types of Ligands

<table>
<thead>
<tr>
<th>Ligand Type</th>
<th>ATP</th>
<th>AMP</th>
<th>ADP</th>
<th>GDP</th>
<th>Ca$^{2+}$</th>
<th>Zn$^{2+}$</th>
<th>Mg$^{2+}$</th>
<th>Mn$^{2+}$</th>
<th>Fe$^{3+}$</th>
<th>DNA</th>
<th>HEME</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T_{ligand}$ (Å)</td>
<td>27.5</td>
<td>29.0</td>
<td>27.5</td>
<td>23.0</td>
<td>26.0</td>
<td>19.0</td>
<td>19.0</td>
<td>20.0</td>
<td>19.0</td>
<td>19.0</td>
<td>61.0</td>
</tr>
</tbody>
</table>

---

**Table 3.** Averaged Size of Binding Site for the 12 Types of Ligands

<table>
<thead>
<tr>
<th>Ligand Type</th>
<th>ATP</th>
<th>AMP</th>
<th>ADP</th>
<th>GDP</th>
<th>Ca$^{2+}$</th>
<th>Zn$^{2+}$</th>
<th>Mg$^{2+}$</th>
<th>Mn$^{2+}$</th>
<th>Fe$^{3+}$</th>
<th>DNA</th>
<th>HEME</th>
</tr>
</thead>
<tbody>
<tr>
<td>Averaged Size</td>
<td>12.69</td>
<td>10.21</td>
<td>12.34</td>
<td>12.82</td>
<td>12.17</td>
<td>3.73</td>
<td>3.07</td>
<td>2.95</td>
<td>3.41</td>
<td>3.80</td>
<td>11.73</td>
</tr>
</tbody>
</table>
sequence will be first modeled by applying MODELLER [41] software package, and then the predicted 3D structure is used for spatial clustering.

After the two-stage prediction, TargetS returns the prediction results back to the client in two different ways: online real-time feedback with 3D illustrations and text descriptions, and an independent e-mail notification to the e-mail address (optional) provided by the user.

3 EXPERIMENTAL RESULTS AND DISCUSSIONS

Specificity (Spe), Sensitivity (Sen), Accuracy (Acc), and the Matthews correlation coefficient (MCC) were used to evaluate the performance of the proposed methods as follows:

\[
\text{Specificity} = \frac{TN}{TN + FP},
\]

\[
\text{Sensitivity} = \frac{TP}{TP + FN},
\]

\[
\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN},
\]

\[
\text{MCC} = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP + FP) \cdot (TP + FN) \cdot (TN + FP) \cdot (TN + FN)}},
\]

where \(TP, FP, TN,\) and \(FN\) denote true positive, false positive, true negative, and false negative, respectively. However, the above four evaluation indexes are threshold dependent and it is well known that Accuracy is not an appropriate evaluation criterion under the imbalanced learning scenario. In view of this, we also exploited AUC, which is the area under the receiver operating characteristic (ROC) curve and has been proved to be a reliable performance measure for imbalanced problems [62], as evaluation index. The AUC is threshold independent and is often used to evaluate the overall prediction quality of a prediction model.

3.1 Five-Fold Cross-Validation Results on Training Data Sets

In this section, we evaluate the performance of the proposed method by performing fivefold cross-validation on training data sets of the 12 considered ligands separately.

As stated in Section 2.3, protein-ligand binding sites prediction is a typical imbalanced learning problem, where the number of samples in minority class (binding residues) is significantly less than that of samples in majority class (nonbinding residues). Under the imbalanced learning scenario, over pursuing the overall accuracy is not appropriate and can be deceiving for evaluating the performance of a predictor. In general, people would expect that a predictor can provide high accuracy of the minority class (e.g., binding residues in this study) without severely jeopardizing the accuracy of the majority class (nonbinding residues) [62]. In view of this, together with the fact that the MCC provides the overall measurement of the quality of the binary predictions, we thus reported the threshold-dependent evaluation indexes (e.g., Sen, Spe, Acc, and MCC) by choosing the threshold, denoted as \(T\), which maximizes the value of MCC of predictions. The performances of the proposed method together with the identified thresholds on the training data sets of the 12 considered ligands over fivefold cross-validation were also listed in Table 4.

Note that since the TargetS is a ligand-specific predictor where the training data sets differ for 12 ligand types, thus the thresholds identified by maximizing MCC criteria will also be different accordingly. However, it is interesting for us to find that the thresholds are similar for those ligands belonging to the same category (e.g., ATP, ADP, AMP, GDP, and GTP belonging to Nucleotide category; Ca\(^{2+}\), Mg\(^{2+}\), Mn\(^{2+}\), Fe\(^{3+}\), and Zn\(^{2+}\) belonging to Metal Ion category) as shown in Table 4.

By observing Table 4, we can find that the proposed TargetS can generate accurate predictions for all the 12 considered ligands. The Specificity (Spe) varies from 94.5 to 99.8 percent, the Accuracy (Acc) from 89.9 to 99.0 percent, the Matthews correlation coefficient (MCC) from 0.320 to 0.644, and the area under the curve (AUC) from 0.784 to 0.938.

We found that for the five types of nucleotide ligands, i.e., ATP, ADP, AMP, GDP, and GTP, the proposed method achieves very similar performances, where AUCs vary
from 0.856 to 0.908. It is easy to understand this phenomenon as these five types of ligands possess similar sizes, roles, and distributions. According to this justification, we speculate that the proposed method will also achieve similar performances for the five metal ion ligands. However, among the five metal ion ligands, Ca^{2+} and Mg^{2+} perform much worse than other three metal ion ligands. The AUCs for Ca^{2+} and Mg^{2+} are less than 0.80, while the AUCs for other three metal ions are all greater than 0.90. There exists almost 10 percent gap between them. The potential reasons for this gap will be investigated in the subsequent section.

### 3.2 Why Ca^{2+} and Mg^{2+} Perform Worse Than Zn^{2+}, Mn^{2+}, and Fe^{3+}?

As stated in Section 2.2, each type of metal ion favors to bind some specific types of residues and the metal ion binding propensity was utilized as indexes to guide the identification process of prediction. Now, we seek to explain why Ca^{2+} and Mg^{2+} perform much worse than Zn^{2+}, Mn^{2+}, and Fe^{3+} from the view of metal ion binding propensity. By revisiting Fig. 1, two observations can be drawn as follows:

1. For all the five types of metal ion ligands, the frequencies of the 20 native amino acids, which are almost the same among nonbinding residues, differ significantly among binding residues.
2. The difference between frequencies of the 20 native amino acids among binding residues and that among nonbinding residues differs significantly for different types of metal ions. We believe that this difference will influence the prediction performance.

For the purpose of quantitative analysis, here we define the frequency difference index, denoted as \( h_{\text{dif}} \), for metal ion ligand as follows:

\[
h_{\text{dif}} = \sum_{i=1}^{20} |f_{\text{b}}^{i} - f_{\text{nb}}^{i}|, \quad (8)
\]

where \( f_{\text{b}}^{i} \) and \( f_{\text{nb}}^{i} \) are the frequencies of the \( i \)th type of the 20 native amino acids among binding and nonbinding residues, respectively. We calculated that the frequency difference indexes for Ca^{2+}, Mg^{2+}, Mn^{2+}, Fe^{3+}, and Zn^{2+} are 73.06, 77.76, 118.09, 129.76, and 154.31, respectively.

We argue that the larger the frequency difference index, the better the prediction performance. As the frequency difference indexes for Ca^{2+} and Mg^{2+} are 73.06 and 77.76, respectively, which are the lowest two among the five frequency difference indexes, thus it is foreseeable that the prediction performances for Ca^{2+} and Mg^{2+} will also be the worst as demonstrated in Table 4. To further validate our argument, let’s take Mn^{2+}, Fe^{3+}, and Zn^{2+} for consideration: the frequency difference index for Mn^{2+} is 118.09, which are less than that for Fe^{3+} (129.76) and Zn^{2+} (154.31), we thus expect that the prediction performance for Mn^{2+} will be worse than that for Fe^{3+} and Zn^{2+}, and the Zn^{2+} will perform best. From Table 4, we found that the AUCs, which are often used to evaluate the overall prediction quality of models, for Mn^{2+}, Fe^{3+}, and Zn^{2+} are 0.901, 0.922, and 0.938, respectively. The results obviously support our speculation.

### 3.3 Ligand-Specific Model Helps to Improve Prediction Performance

In this section, we will empirically demonstrate that the proposed ligand-specific prediction model is superior to general-purpose prediction model which does not distinguish the types of ligands.

We carried out general-purpose prediction experiments as follows: training data sets of the 12 types of considered ligands were merged to obtain a combined training data set; then, a fivefold cross-validation procedure was performed on the combined training data set. Note that in each round of cross-validation, the positive samples were those binding residues regardless of the types of ligands they bind to, while the negative samples were those nonbinding residues accordingly. In the evaluation stage, performances were calculated for each type of ligands individually. Note that

<table>
<thead>
<tr>
<th>Category</th>
<th>Ligand Type</th>
<th>Threshold *</th>
<th>Sens (%)</th>
<th>Spec (%)</th>
<th>Acc (%)</th>
<th>MCC</th>
<th>AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nucleotide</td>
<td>ATP</td>
<td>( T_{\text{ATP}} = 0.50 )</td>
<td>48.4</td>
<td>98.2</td>
<td>96.2</td>
<td>0.492</td>
<td>0.887</td>
</tr>
<tr>
<td></td>
<td>ADP</td>
<td>( T_{\text{ADP}} = 0.50 )</td>
<td>56.1</td>
<td>98.8</td>
<td>97.2</td>
<td>0.591</td>
<td>0.907</td>
</tr>
<tr>
<td></td>
<td>AMP</td>
<td>( T_{\text{AMP}} = 0.50 )</td>
<td>38.0</td>
<td>98.2</td>
<td>96.0</td>
<td>0.386</td>
<td>0.856</td>
</tr>
<tr>
<td></td>
<td>GDP</td>
<td>( T_{\text{GDP}} = 0.43 )</td>
<td>63.9</td>
<td>98.7</td>
<td>97.2</td>
<td>0.644</td>
<td>0.908</td>
</tr>
<tr>
<td></td>
<td>GTP</td>
<td>( T_{\text{GTP}} = 0.50 )</td>
<td>48.0</td>
<td>98.7</td>
<td>96.9</td>
<td>0.506</td>
<td>0.858</td>
</tr>
<tr>
<td>Metal Ion</td>
<td>Ca^{2+}</td>
<td>( T_{\text{Ca}^{2+}} = 0.69 )</td>
<td>19.2</td>
<td>99.7</td>
<td>98.4</td>
<td>0.320</td>
<td>0.784</td>
</tr>
<tr>
<td></td>
<td>Mg^{2+}</td>
<td>( T_{\text{Mg}^{2+}} = 0.81 )</td>
<td>26.4</td>
<td>99.8</td>
<td>99.0</td>
<td>0.383</td>
<td>0.798</td>
</tr>
<tr>
<td></td>
<td>Mn^{2+}</td>
<td>( T_{\text{Mn}^{2+}} = 0.74 )</td>
<td>40.8</td>
<td>99.5</td>
<td>98.7</td>
<td>0.445</td>
<td>0.901</td>
</tr>
<tr>
<td></td>
<td>Fe^{3+}</td>
<td>( T_{\text{Fe}^{3+}} = 0.81 )</td>
<td>51.8</td>
<td>99.6</td>
<td>98.8</td>
<td>0.592</td>
<td>0.922</td>
</tr>
<tr>
<td></td>
<td>Zn^{2+}</td>
<td>( T_{\text{Zn}^{2+}} = 0.83 )</td>
<td>50.0</td>
<td>99.6</td>
<td>98.9</td>
<td>0.557</td>
<td>0.938</td>
</tr>
<tr>
<td></td>
<td>DNA</td>
<td>( T_{\text{DNA}} = 0.49 )</td>
<td>41.7</td>
<td>94.5</td>
<td>89.9</td>
<td>0.362</td>
<td>0.824</td>
</tr>
<tr>
<td></td>
<td>HEME</td>
<td>( T_{\text{HEME}} = 0.65 )</td>
<td>50.5</td>
<td>98.3</td>
<td>94.4</td>
<td>0.579</td>
<td>0.887</td>
</tr>
</tbody>
</table>

* For each type of ligands, say Mg^{2+}, the threshold \( T_{\text{Mg}^{2+}} \) was identified by maximizing the MCC value of predictions on the corresponding training dataset over five-fold cross-validation.
as we do not distinguish the ligand types in this experiment, thus the binding propensity feature were obtained based on the frequencies of the 20 native amino acids among binding residues regardless of the types of ligands they bind to.

The performance comparisons between the general-purpose and ligand-specific predictions on the training data sets of the 12 considered ligands are listed in Table S1, which can be found in the online supplemental material available on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TCBB.2013.104. Note that for the convenience of comparison, we restricted the false positive rate (FPR) of predictions to be (or most close to) 5 percent.

By observing Table S1, which is available in the online supplemental material, it is easy to find that the other four evaluation indexes, i.e., Sen, Acc, MCC, and AUC, of ligand-specific predictions are consistently superior to that of general-purpose predictions under the same Spe (95 percent or most close to 95 percent) for all the 12 considered ligands. Particularly, ligand-specific predictions significantly outperform general-purpose predictions on the Sen and MCC and averaged improvements of 14.6 and 10.0 percent were achieved, respectively. As to AUC, the index for evaluating the overall prediction quality of predictions, the ligand-specific method also outperforms general-purpose method with an averaged improvement of 4.7 percent.

Experimental results listed in Table S1, which is available in the online supplemental material, demonstrate that the ligand-specific method is superior to general-purpose method, at least on the tested benchmark data sets. These improvements may result from the following two aspects: On the one hand, ligand-specific method performs predictions for different ligands with corresponding specialized prediction models, which can accommodate the differences of roles, sizes, and distributions existed in different types of ligands much better. On the other hand, in general-purpose method, binding propensity feature is calculated based on the frequencies of the 20 native amino acids among binding residues regardless of the types of ligands they bind to, while in ligand-specific method, binding propensity feature is calculated for each type of ligands respectively thus can characterize the ligand binding propensities of residues much well. We believe this ligand-specific binding propensity feature possess much better discrimination capability, which maybe one of the most important factors that account for the performance improvements.

3.4 Comparison with Existing Predictors

In this section, we will experimentally demonstrate the efficacy of the proposed TargetS by comparing it with other popular predictors via independent test and blind test as follows:

3.4.1 Independent Test on Validation Data Sets

For each type of ligands, we compared TargetS with two other ligand-specific predictors and one alignment-based baseline predictor on the independent validation data set. Ligand-specific predictors for comparison differ for different types of ligands. More specifically, NsitePred [30], a most recently developed predictor specifically designed for predicting nucleotide-binding residues, and SVMPred [7] were chosen for comparison for the five types of nucleotide ligands; FunFOLD [73] and CHED [74] were taken as ligand-specific predictors for comparison for the five types of metal ion ligands; Note that FunFOLD was initially designed for general-purpose binding sites prediction. In this study, we retrained the FunFOLD on the data set of each type of the five metal ion ligands and used it as ligand-specific predictor. MetaDBSite [75], a meta approach for protein-DNA binding sites prediction, and DNABR [76] were used as ligand-specific predictors for comparison for the DNA ligand; finally, HemeNet [37] and HemeBind [12], which are the only two available predictors (to the best of our knowledge) that were specifically designed for predicting HEME binding residues. As the webserver of HemeNet does not work currently, thus only HemeBind was taken for comparison for HEME ligand. Note that when performing comparisons between TargetS and the above-mentioned ligand-specific predictors, we trained our TargetS using training data set for that ligand and tested it with the corresponding independent validation data set (see Table 1), while prediction results of other ligand-specific predictors were obtained by feeding sequences or 3D structures in the independent validation data set to their corresponding web servers. As SVMPred does not provide a web server, we thus locally implemented it based on our training data sets of the five nucleotide ligands and tested it with the corresponding independent validation data sets.

Alignment-based predictions were performed as follows: for each query sequence in a validation data set, we executed the alignment-based prediction by aligning the query sequence and all the sequences in the corresponding training data set. The residues in the query sequence that were aligned with the binding residues of the best aligned sequence are predicted as the binding residues.

Table 5 illustrates the performance comparison of TargetS with NsitePred, SVMPred, and alignment-based predictor on the independent validation data sets of the five nucleotide ligands. Considering the page limit, comparison results for other seven types of ligands, i.e., Ca$^{2+}$, Mg$^{2+}$, Mn$^{2+}$, Fe$^{3+}$, Zn$^{2+}$, DNA, and HEME are listed in the supplemental materials (see Tables S2, S3, and S4, which are available in the online supplemental material).

From Table 5, we can find that the AUCs for ATP, ADP, AMP, GDP, and GTP on the corresponding independent validation data sets are 0.898, 0.896, 0.830, 0.896, and 0.855, respectively. By revisiting Table 4, it is found that the AUCs for ATP, ADP, AMP, GDP, and GTP on the training data sets are 0.887, 0.907, 0.856, 0.908, and 0.896, respectively. In other words, for all the five nucleotide ligands, TargetS achieves similar overall prediction performances (measured by AUCs) on the training data set and independent validation data set, denoting that the generalization capability of the TargetS derived from the knowledge buried in training data sets has not been over- or underestimated.

By observing Table 5, it is found that the alignment-based predictor achieve very similar performances on Spe ($\geq$97%) to other three machine-learning-based predictors. However, its performances on Sen are poorest (i.e., the alignment-based predictor predicts too many false
negatives) resulting in low MCCs for all the five types of nucleotide ligands only except for GTP.

We also find that that the three machine-learning-based predictors almost always perform better than the alignment-based predictor. Among the three machine-learning-based predictors, the proposed TargetS consistently performs the best on Spe, Acc, and MCC for all the five nucleotide ligands and acts as the best performance. Compared with the second-best performer, i.e., NsitePred, TargetS achieves averaged improvements of 1.2, 1.1, and 7.3 percent on Spe, Acc, and MCC, respectively, while still possesses almost the equal performance on Sen as NsitePred at the same time.

### 3.4.2 Blind Test on CASP9 Data Set

As described in Section 2.1, 30 protein sequences in CASP9 data set were used for blind test. Considering that the sequences in CASP9 interact with many different types of ligands, we thus compare TargetS with several popular general-purpose rather than ligand-specific predictors. ConSeq [77], a webserver for the identification of biologically important residues in protein sequences, was chosen as the sequence-based general-purpose predictor for comparison, while SITEHOUND [19] and MetaPocket [78] were chosen as the structure-based general-purpose predictors for comparison.

The prediction results of ConSeq [77], SITEHOUND [19], and MetaPocket [78] were obtained as follows: For each blind test protein sequence, we fed it to the 12 ligand-specific prediction models, respectively, and the predicted binding residues of the 12 models were then merged as the final prediction results for that sequence. Table 6 compares the performance of TargetS with ConSeq [77], SITEHOUND [19], and MetaPocket [78] for the 30 targets in CASP9.

According to the MCC, which is the overall measurement of the quality of the binary predictions, listed in Table 6, we can find that the TargetS acts as the best performer followed by MetaPocket, SITEHOUND, and ConSeq. TargetS significantly outperforms the other sequence-based predictor, i.e., ConSeq [77], on all the four evaluation indexes and an improvement of 20 percent on MCC was achieved. In addition, TargetS also overwhelms the structure-based predictor SITEHOUND. The Sen, Acc, and MCC of TargetS are 38.9, 94.2, and 0.317 percent, respectively, which are 16.4,

<table>
<thead>
<tr>
<th>Ligand Type</th>
<th>Predictor</th>
<th>Sen (%)</th>
<th>Spe (%)</th>
<th>Acc (%)</th>
<th>MCC</th>
<th>AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>ATP</td>
<td>TargetS</td>
<td>50.1</td>
<td>98.3</td>
<td>96.5</td>
<td>0.502</td>
<td>0.898</td>
</tr>
<tr>
<td></td>
<td>NsitePred</td>
<td>50.8</td>
<td>97.3</td>
<td>95.5</td>
<td>0.439</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>SVMPred</td>
<td>47.3</td>
<td>96.7</td>
<td>94.9</td>
<td>0.387</td>
<td>0.877</td>
</tr>
<tr>
<td></td>
<td>alignment-based</td>
<td>30.6</td>
<td>97.0</td>
<td>94.5</td>
<td>0.265</td>
<td>-</td>
</tr>
<tr>
<td>ADP</td>
<td>TargetS</td>
<td>46.9</td>
<td>98.9</td>
<td>97.2</td>
<td>0.507</td>
<td>0.896</td>
</tr>
<tr>
<td></td>
<td>NsitePred</td>
<td>46.2</td>
<td>97.6</td>
<td>96.0</td>
<td>0.419</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>SVMPred</td>
<td>46.1</td>
<td>97.2</td>
<td>95.5</td>
<td>0.382</td>
<td>0.875</td>
</tr>
<tr>
<td></td>
<td>alignment-based</td>
<td>31.8</td>
<td>97.4</td>
<td>95.1</td>
<td>0.284</td>
<td>-</td>
</tr>
<tr>
<td>AMP</td>
<td>TargetS</td>
<td>34.2</td>
<td>98.2</td>
<td>95.9</td>
<td>0.359</td>
<td>0.830</td>
</tr>
<tr>
<td></td>
<td>NsitePred</td>
<td>33.9</td>
<td>97.6</td>
<td>95.3</td>
<td>0.321</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>SVMPred</td>
<td>32.1</td>
<td>96.4</td>
<td>94.1</td>
<td>0.255</td>
<td>0.798</td>
</tr>
<tr>
<td></td>
<td>alignment-based</td>
<td>19.6</td>
<td>97.3</td>
<td>94.5</td>
<td>0.178</td>
<td>-</td>
</tr>
<tr>
<td>GDP</td>
<td>TargetS</td>
<td>56.2</td>
<td>98.1</td>
<td>96.2</td>
<td>0.550</td>
<td>0.896</td>
</tr>
<tr>
<td></td>
<td>NsitePred</td>
<td>55.7</td>
<td>97.9</td>
<td>96.1</td>
<td>0.536</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>SVMPred</td>
<td>49.5</td>
<td>97.6</td>
<td>95.4</td>
<td>0.466</td>
<td>0.870</td>
</tr>
<tr>
<td></td>
<td>alignment-based</td>
<td>41.2</td>
<td>97.8</td>
<td>95.3</td>
<td>0.415</td>
<td>-</td>
</tr>
<tr>
<td>GTP</td>
<td>TargetS</td>
<td>57.3</td>
<td>98.8</td>
<td>96.9</td>
<td>0.617</td>
<td>0.855</td>
</tr>
<tr>
<td></td>
<td>NsitePred</td>
<td>58.4</td>
<td>95.7</td>
<td>94.0</td>
<td>0.448</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>SVMPred</td>
<td>48.3</td>
<td>91.7</td>
<td>89.7</td>
<td>0.276</td>
<td>0.821</td>
</tr>
<tr>
<td></td>
<td>alignment-based</td>
<td>52.8</td>
<td>97.9</td>
<td>95.9</td>
<td>0.516</td>
<td>-</td>
</tr>
</tbody>
</table>

### Table 6

Performance Comparison of TargetS with ConSeq, SITEHOUND and MetaPocket for the 30 Targets in CASP9

<table>
<thead>
<tr>
<th>Predictor</th>
<th>Sen (%)</th>
<th>Spe (%)</th>
<th>Acc (%)</th>
<th>MCC</th>
</tr>
</thead>
<tbody>
<tr>
<td>TargetS</td>
<td>38.9</td>
<td>96.5</td>
<td>94.2</td>
<td>0.317</td>
</tr>
<tr>
<td>MetaPocket [78]</td>
<td>70.4</td>
<td>80.0</td>
<td>79.7</td>
<td>0.237</td>
</tr>
<tr>
<td>SITEHOUND [19]</td>
<td>22.5</td>
<td>95.1</td>
<td>92.3</td>
<td>0.149</td>
</tr>
<tr>
<td>ConSeq [77]</td>
<td>32.1</td>
<td>87.7</td>
<td>85.5</td>
<td>0.114</td>
</tr>
</tbody>
</table>
1.9, and 16.8 percent higher than that of SITHOUND, which is the third-best performer. We also found that the second-best performer, i.e., MetaPocket, achieves the highest Sen (70.4 percent), which is almost twice that, i.e., 38.9 percent, of the TargetS. However, the Spe of MetaPocket is only 80.0 percent which is lowest one among the four considered predictors. In other words, MetaPocket predicts too many false positives (nonbinding residues are predicted as binding ones), thus the reliability of the predicted binding residues of MetaPocket is the lowest among the listed predictors.

Currently, TargetS embraces 12 types of ligands and most types of ligands contained in CASP9 were in fact not covered by TargetS. However, by observing Table 6, we can find that TargetS still achieves satisfactory performances for the 30 blind test proteins and acts as the best performer among the four listed predictors. We speculate that is due to the multiple ligands coverage of TargetS and the hypothesis that binding residues for different ligands can share some common features, TargetS can thus yield good prediction results for the ligands that not belonging to the 12 types.

3.4.3 Performing Comparison on Benchmark Data Set That Has Been Used by Other Predictors

Except for the independent validation test and blind test performed above, we will try to further demonstrate the efficacy of the proposed TargetS by comparing it with other predictors based on the same benchmark data set that has been used by the compared predictors. Taking protein-nucleotide binding sites prediction as an example, the data set 1 constructed by Chen et al. [30] was taken as the benchmark data set. The data set 1 [30] consists of 227, 321, 140, 56, and 105 sequences that bind to ATP, ADP, AMP, GTP, and GDP, respectively, and the maximal pairwise sequence identity of the sequences among each type of the five nucleotides was less than 40 percent.

We compared TargetS with SVMPred [7], Rate4site [23], and NsitePred [30] on data set 1 over fivefold cross-validation as done in [30] and the comparison results were listed in Table 7. By observing Table 7, we found that the TargetS obtains AUC >0.85, MCC >0.41 for all the five types of nucleotide ligands. The AUC and MCC values of TargetS are consistently superior to that of all the other three considered predictors, i.e., SVMPred [7], Rate4site [23], and NsitePred [30] and the averaged improvements of 2.3 and 5.4 percent were achieved, respectively, if compared with the second-best performer NsitePred [30]. It has not escaped from our notice that Rate4site [23] achieves highest Sen values, i.e., 56.2 and 56.9 percent for AMP and GTP, respectively. However, the corresponding Spe values are much lower, i.e., 79.9 and 80.6 percent, denoting too many false positives were incurred during prediction. On the other hand, SVMPred [7] achieves the best performances on Spe for several nucleotide ligands (i.e., 99.3, 99.6, and 99.7 percent for ADP, AMP, and GTP, respectively) while with much lower Sen values implying too many false negatives were produced during prediction.

4 Conclusions

We have designed and implemented a ligand-specific template-free predictor, called TargetS, for predicting protein-ligand binding sites. TargetS was trained on data collected from a most recently released protein-ligand interaction database BioLip [45] with our proposed modified AdaBoost (MAdaBoost) learning algorithm [79]. Experimental results on training data sets, independent validation data sets and blind test on CASP9 data set have demonstrated the efficacy of the proposed TargetS. The success of TargetS is due to several reasons include good benchmark data sets, ligand-specific propensity discriminative feature design, careful construction of the prediction model and the spatial clustering algorithm. Currently, the TargetS prediction server has already been put online and been able to predict binding sites for 12 types of ligands. Our future work will focus on further incorporating binding sites prediction modules for other new types of ligands into TargetS to enhance its capability. In addition, we will also continue to improve the prediction accuracy of TargetS by developing new effective feature extraction methods and applying more powerful classifiers.

In the current work, a major problem of TargetS is that the computation time is relatively long (about 325 s for a sequence with 300 residues), although it is fast than many template-based methods such as MetaPocket, because the TargetS have to perform PSI-BLAST [57], PSIPRED [59], and MODELLER [43] to extract features and model the 3D structure from a protein sequence. We will also try to accelerate the computation speed by using several servers to parallelly perform these computations in the future.

Acknowledgments

The authors would like to thank the anonymous reviewers for suggestions and comments which helped improve the quality of this paper. This work was supported by the National Natural Science Foundation of China (no. 61373062, 91130033, 61175024, 61222306, and 61233011), the Jiangsu Postdoctoral Science Foundation.


Dong-Jun Yu received the BS degree in computer science and the MS degree in artificial intelligence from Jiangsu University of Science and Technology, China, in 2000 and 2002, respectively, and the PhD degree in pattern analysis and machine intelligence from Nanjing University of Science and Technology in 2003. In 2008, he acted as an academic visitor at the University of York in the United Kingdom. He is currently an associate professor at the School of Computer Science and Engineering, Nanjing University of Science and Technology. His current interests include bioinformatics, pattern recognition, and data mining. He is a member of the IEEE and the CCF.

Jun Hu received the BS degree in computer science from Anhui Normal University, China, in 2011. Currently, he is working toward the PhD degree at the School of Computer Science and Engineering, Nanjing University of Science and Technology, China. His research interests include bioinformatics, data mining, and pattern recognition.
Jing Yang received the BS degree in automation from the North University of China, in 2010. He is working toward the MS degree in pattern recognition at Shanghai Jiao Tong University, China. His research interests include pattern recognition, machine learning algorithms, and bioinformatics.

Hong-Bin Shen received the PhD degree from Shanghai Jiaotong University, China, in 2007. He was a postdoctoral research fellow of Harvard Medical School from 2007 to 2008. Currently, he is a professor at the Institute of Image Processing and Pattern Recognition, Shanghai Jiaotong University. His research interests include data mining, pattern recognition, and bioinformatics. He has published more than 60 papers and constructed 20 bioinformatics servers in these areas, and he serves as editorial board member of several international journals.

Jinhui Tang received the BE and PhD degrees both from the University of Science and Technology of China, in 2003 and 2008, respectively. He is currently a professor at the School of Computer Science and Engineering, Nanjing University of Science and Technology. His current research interests include large-scale multimedia search, social media mining, and computer vision. He has authored more than 80 journal and conference papers in these areas. He serves as an editorial board member of Pattern Analysis and Applications, Multimedia Tools and Applications, Information Sciences, Neurocomputing; a technical committee member for about 30 international conferences; and a reviewer for about 30 prestigious international journals. He is a co-recipient of the Best Paper Award from ACM Multimedia 2007, PCM 2011, and ICIMCS 2011. He is a member of the IEEE, the ACM, and the CCF.

Jing-Yu Yang received the BS degree in computer science from Nanjing University of Science and Technology (NUST), China. From 1982 to 1984, he was a visiting scientist at the Coordinated Science Laboratory, University of Illinois at Urbana-Champaign. From 1993 to 1994, he was a visiting professor in the Department of Computer Science, Missouri University, and in 1998, he acted as a visiting professor at Concordia University, Canada. He is currently a professor and chairman in the Department of Computer Science at NUST. His current research interests include the areas of pattern recognition, robot vision, image processing, data fusion, and artificial intelligence. He is the author of more than 150 scientific papers on computer vision, pattern recognition, and artificial intelligence. He has won more than 20 provincial awards and national awards.

For more information on this or any other computing topic, please visit our Digital Library at www.computer.org/publications/dlib.