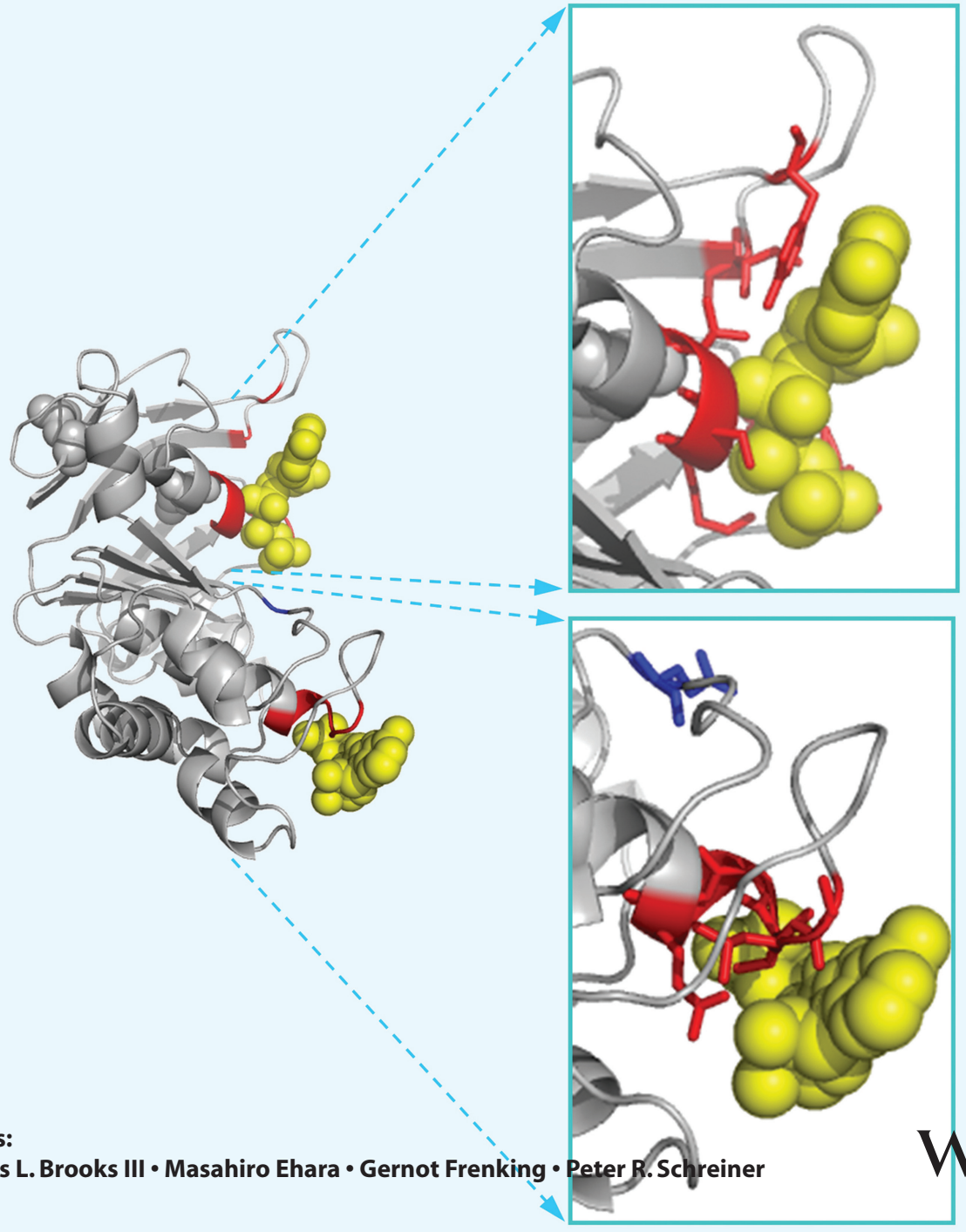
Volume 34 | Issues 11–12 | 2013
Included in this print edition:
**Issue 11 (April 30, 2013)**
Issue 12 (May 5, 2013)

# Journal of COMPUTATIONAL CHEMISTRY

Organic • Inorganic • Physical
Biological • Materials

www.c-chem.org



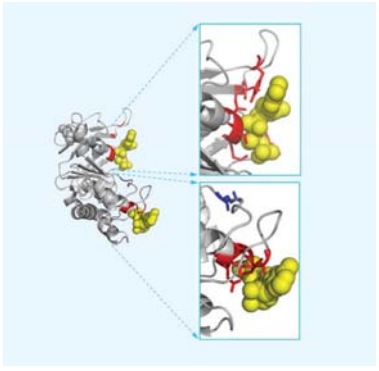


WILEY

Journal of COMPUTATIONAL CHEMISTRY
Organic • Inorganic • Physical
Biological • Materials

## Parallel Molecular Docking

The front cover illustrates the parallel molecular docking of large databases on the Sequoia, a petascale IBM Blue Gene/Q supercomputer at Lawrence Livermore National Laboratory. A mixed parallel scheme that combines MPI and multithreading is implemented by Xiaohua Zhang, Sergio E. Wong, and Felice C. Lightstone on page 915 in the Vina molecular docking program named VinaLC, where LC stands for Livermore Computing. Parallel performance analysis shows the code scales up to more than 15K CPUs with a very low overhead cost of 3.94%. One million flexible compound docking calculations take only 1.4 hours on about 15K CPUs. The picture shows ligands that have been docked into various receptors to form ligand–receptor complexes via calculations on the Sequoia.



## ATP Binding Site Prediction

TargetATPsite, a new method based on residue evolution image sparse representation and classifier ensemble, is developed for predicting ATP-binding sites from primary sequences, as presented by Dong-Jun Yu, Jun Hu, Yan Huang, Hong-Bin Shen, Yong Qi, Zhen-Min Tang, and Jing-Yu Yang on page 974. The high performance of TargetATPsite originates from the good discriminative capability of the new image sparse representation feature and the power of the modified AdaBoost classifier ensemble. TargetATPsite also features the capability of further identifying the binding pockets from the predicted binding residues through a spatial clustering algorithm.

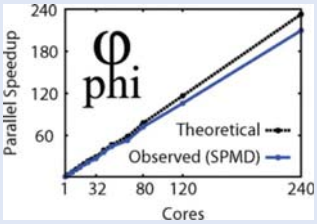
# Coming Soon

**Look for these important papers in upcoming issues**

**PHI: A powerful new program for the analysis of anisotropic monomeric and exchange-coupled polynuclear *d*- and *f*-block complexes**

Keith S. Murray et al.

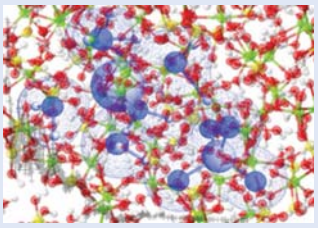
A new and extensively parallelized code for the calculation of the magnetic properties of large spin systems or complex orbitally degenerate compounds is presented. The program can simulate theoretical systems or fit experimental data with a specific Hamiltonian.
DOI: 10.1002/jcc.23234




**Parameterization of a reactive force field using a Monte Carlo algorithm**

C. C. M. Rindt et al.

Parameterization of a reactive force field (ReaxFF) is performed using a robust Metropolis Monte Carlo algorithm for a system of magnesium sulfate hydrates. This new method for optimizing the force field is efficient especially without good initial conditions. The stochastic nature enables one to arrive at the global minimum in the parameter space and thereby the best obtainable force field.
DOI: 10.1002/jcc.23246



# TargetATPsite: A Template-free Method for ATP-Binding Sites Prediction with Residue Evolution Image Sparse Representation and Classifier Ensemble

Dong-Jun Yu,[a,b] Jun Hu,[a] Yan Huang,[c] Hong-Bin Shen,*[d,e] Yong Qi,[a,b] Zhen-Min Tang,[a] and Jing-Yu Yang[a]

Understanding the interactions between proteins and ligands is critical for protein function annotations and drug discovery. We report a new sequence-based template-free predictor (TargetATPsite) to identify the Adenosine-5′-triphosphate (ATP) binding sites with machine-learning approaches. Two steps are implemented in TargetATPsite: binding residues and pockets predictions, respectively. To predict the binding residues, a novel image sparse representation technique is proposed to encode residue evolution information treated as the input features. An ensemble classifier constructed based on support vector machines (SVM) from multiple random under-samplings is used as the prediction model, which is effective for dealing with imbalance phenomenon between the positive and negative training samples. Compared with the existing ATP-specific sequence-based predictors, TargetATPsite is featured by the second step of possessing the capability of further identifying the binding pockets from the predicted binding residues through a spatial clustering algorithm. Experimental results on three benchmark datasets demonstrate the efficacy of TargetATPsite.


DOI: 10.1002/jcc.23219

## Introduction

Protein–ligand interactions are ubiquitous and play important roles in a wide variety of biological processes.[1–3] Hence, accurately identifying the protein–ligand binding sites or pockets is of significant importance for both protein function analysis and drug design.[4] Tremendous experimental efforts have been made to understand protein–ligand interactions and thousands of protein–ligand interaction structure complexes have been deposited into PDB.[5] However, experimentally identifying the protein–ligand interaction sites is still labor-intensive and time-consuming. Hence, it is highly desired to develop intelligent automatic computational methods for protein–ligand binding sites prediction to speed up the annotation process especially when facing with the large-scale protein sequences in the post-genomic era.[6–8]

There have emerged many computational methods for predicting protein–ligand binding sites during the past decade.[9,10] Roughly speaking, these existing methods can be grouped into three categories[11]: sequence-based methods, structure-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: POCKET,[12] LIGSITE,[13] SURFNET,[14] and fpocket,[15] etc. Later on, researchers found that sequence-induced conservation information can also be effectively used for protein–ligand binding sites prediction. For example, ConSurf[16] and Rate4Site[17] use the evolutionary data in the form of multiple-sequence alignment for a protein family to identify hot spots and surface patches that are likely to be in contact with other proteins, domains, peptides, DNA, RNA, or ligands; L1pred[18] predicts catalytic residues in enzymes by using the L1-logreg classifier to integrate eight sequence-based scoring functions. Recently, much attention has been paid to the methods that combine both the structure and the sequence information. For example, LIGSITE$^{csc}$ [19] extends the LIGSITE[13] by further

[a] D.-J. Yu, J. Hu, Y. Qi, Z.-M. Tang, J.-Y. Yang
School of Computer Science and Engineering, Nanjing University of Science and Technology, Xiaolingwei 200, Nanjing 210094, China
[b] D.-J. Yu, Y. Qi
Research and Development Center, Changshu Institute, Nanjing University of Science and Technology, Changshu 215513, China
[c] Y. Huang
National Laboratory for Infrared Physics, Shanghai Institute of Technical Physics, Chinese Academy of Science, Yutian Road 500, Shanghai 200083, China
[d] H.-B. Shen
Department of Automation, Shanghai Jiao Tong University, and Key Laboratory of System Control and Information Processing, Ministry of Education of China, Dongchuan Road 800, Shanghai 200240, China
[e] H.-B. Shen
Department of Computational Medicine and Bioinformatics, University of Michigan, 100 Washtenaw Avenue, Ann Arbor, Michigan 48109
Fax: (+86) 21 34204022
E-mail: hbshen@sjtu.edu.cn

Contract/grant sponsor: National Natural Science Foundation of China; Contract/grant numbers: 91130033, 61175024, 61233011, 61222306, 61006091; Contract/grant sponsor: Natural Science Foundation of Jiangsu; Contract/grant number: BK2011371; Contract/grant sponsor: Jiangsu Postdoctoral Science Foundation; Contract/grant number: 1201027C; Contract/grant sponsor: Industry-Academia Cooperation Innovation Fund Projects of Jiangsu Province; Contract/grant number: BY2012022; Contract/grant sponsor: Foundation for the Author of National Excellent Doctoral Dissertation of PR China; Contract/grant number: 201048; Contract/grant sponsor: Shanghai Science and Technology Commission; Contract/grant number: 11JC1404800.

incorporating the degree of conservation of the involved surface residues; ConCavity[20] integrates evolutionary sequence conservation estimates with structure-based methods for identifying protein surface cavities; SURFNET-ConSurf[21] also incorporates residue evolutionary conservation into pocket detection. All in all, much progress has been made in computational methods for protein–ligand binding site prediction and many applications based on these methods have emerged.

However, there are still several issues deserved to be further discussed. For example, many structure-based methods are template-based that require the tertiary protein structures as inputs to search for homology protein–ligand complex structures for comparison.[9,10] Although the success of these structure-based methods have been demonstrated with many good software packages, e.g., Q-SiteFinder[22] and SITEHOUND,[23] their applicabilities will be limited especially in post-genomic era where there exists large number of protein sequences without known structures and at the same time for some query targets there are no homology templates in current PDB database. This is one of the major reasons that motivate researchers in this field to develop useful tools for predicting protein–ligand binding sites from only the protein sequence information. On the other hand, previous reports have shown that protein binding sites vary significantly in their roles in different types of protein–ligand interactions.[24] Thus, developing ligand-specific binding site predictor has attracted considerable attentions to expect much more accurate predictions. As a result, many ligand-specific binding site predictors have emerged recently. For example, Sodhi et al.[25] exploited neural network methods to predict metal ions binding sites; Brylinski et al.[26] extended the FINDSITE software to FINDSITE-metal specifically for predicting metal ions binding sites; Kumar et al.[27] developed Pprint, a RNA binding site predictor using support vector machines (SVM) and position-specific scoring matrix (PSSM) profiles; Liu et al. developed HemeNet[28] and HemeBIND [11] for specifically predicting heme binding residues based on structural and sequential information. Recently, several predictors for Adenosine-5′-triphosphate (ATP) binding residues prediction have also been developed.[6,7,29]

Adenosine-5′-triphosphate is an important molecule in cell that plays important roles in membrane transport, muscle contraction, cellular motility, signaling, replication, and transcription of DNA, and various metabolic processes.[30,31] Adenosine-5′-triphosphate interacts with proteins through protein–ATP binding sites and provides chemical energy to proteins through the hydrolysis of ATP.[32] Powered by the chemical energy, a protein can then perform various biological functions. In addition, the ATP binding sites are also valuable drug targets for antibacterial and anti-cancer chemotherapy. Hence, accurately localizing the protein–ATP binding sites is of significant importance for both protein function analysis and drug design. Developing accurate intelligent automatic computational methods for protein–ATP binding prediction is in urgent need.[6–8]

Unfortunately, because of the limited experimentally verified ATP binding proteins, there are no such predictors until ATPint[6] was reported. ATPint was built on a benchmark dataset consisting of 168 non-redundant ATP-binding proteins. Following ATPint, another two protein–ATP binding predictors were constructed very recently based on a larger benchmark dataset of 227 protein sequences, i.e., ATPsite[7] and NsitePred,[29] where only NsitePred provides online services.

A drawback of existing sequence-based protein–ATP binding prediction methods, including ATPint,[6] ATPsite,[7] and NsitePred[29], is that they only predict the protein–ATP binding residues and do not tell which residues may potentially form the binding sites (pockets). Besides knowing individual binding residues alone, it would also be of great use if the real binding pockets can be identified through the set of predicted individual residues.

This article will follow the abovementioned pioneering work on predicting ATP-binding residues from sequences, and aims to further improve the binding residues prediction performance. In addition, we also make efforts to further identify the potential binding sites from the predicted binding residues. A new template-free predictor, called TargetATPsite, is proposed with machine-learning techniques. The first task of TargetATPsite is to predict the binding residues from the primary sequence, which is achieved by an ensembled classifier. The input features to the predictor are encoded by a novel image sparse representation of the residue conservation matrix. In order to release the serious imbalance between negative and positive samples (i.e., non-binding and binding residues), random under-sampling technique is applied in TargetATPsite, followed by which the AdaBoost classifier ensemble scheme[33] with SVM[34,35] as base classifier is used to relieve the impact of information loss caused by random under-sampling. According to the predicted binding residues, a spatial clustering algorithm is developed to find the binding sites (pockets) from the protein 3D structures either provided by the user or modeled by the MODELLER software.[36]

## Materials and Methods

### Benchmark datasets

Two benchmark datasets were used to demonstrate the effectiveness of the proposed TargetATPsite. The first dataset was selected from SuperSite encyclopedia[37] by Chauhan et al.[6] and consists of 168 non-redundant protein sequences, denoted as ATP168. The sequence identity between any two sequences in ATP168 is below 40%. The second dataset was constructed by Chen et al.[7]: first, they extracted all complexes in PDB (as of February 2010) that include ATP; then, the maximal pairwise sequence identity of the resulting protein sequences was reduced to 40% with CD-hit[38]; the remaining 227 chains constitute the final dataset, denoted as ATP227. To further demonstrate the generalization capability of the TargetATPsite, an independent testing dataset which contains 17 ATP-binding protein sequences was also taken as done in Ref. [7].

### Sparse feature extraction

**Position-specific scoring matrix feature.** Protein evolutionary information encoded in position-specific scoring matrix (PSSM) has been demonstrated to be an effective feature source for

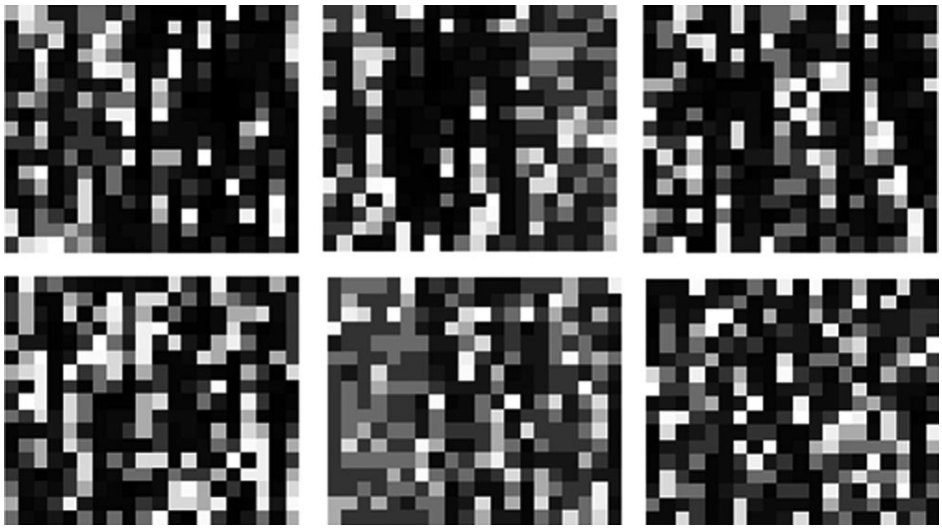


**Figure 1.** Evolution images of six residues randomly chosen from the dataset ATP227. Images in top row are of the three binding residues and images in bottom row are of the three non-binding residues.

reflecting the residue conservations. For a protein sequence P with $N$ amino acid residues, we obtain its PSSM ($N$ rows and 20 columns) by using the PSI-BLAST[39] 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. Elements in the $i$ th row of PSSM measure the probabilities of the $i$ th residue in the protein sequence being mutated to 20 native residues during the evolution process. Then, we normalize the obtained PSSM by the logistic function defined as follows:

$$f(x) = \frac{1}{1 + \exp(-x)} \quad (1)$$

where $x$ is the original score in PSSM matrix.

Based on the normalized PSSM, a sliding window with size $W$ is taken to extract feature vector for each residue. More specifically, the feature vector of a residue is obtained by concatenating the scaled PSSM scores of its neighboring residues within the window centered at the residue. In this study, we have tested different $W$ and found that $W = 17$ is a better choice. Thus, the PSSM feature for each residue is a matrix of size $17 \times 20$.

**Sparse representation of evolution image.** As stated in above section, the PSSM feature of a residue is represented by a matrix of size $17 \times 20$, among which the value of each element is within the range of 0–1. Interestingly, from the perspective of digital image processing, this matrix can be considered as the evolution image of the residue.

Figure 1 intuitively illustrates evolution images of six residues randomly selected from the dataset ATP227, where three residues are binding residues and the other three are non-binding residues. By carefully observing images illustrated in Figure 1, we can find that there does exist differences between binding and non-binding residue images, although it is hard for humankind to directly cognize the actual meaning of these images. It is found there tends to appear more dark areas in binding residue images.

Considering these evolution images are generated by PSI-BLAST[39] software, it is inevitable that noises will be introduced into the generated images since redundant sequences may be contained in the multiple sequence alignments. Thus, reducing noises contained in images will help to improve the image qualities and thus enhance the subsequent prediction performance. On the other hand, dimensionality reduction is demonstrated as an effective procedure to remove redundancy in image processing.[40] Considering the above two reasons, we apply a new sparse representation way to represent the residue evolution images in this study.

Recently, much attention has been paid to image sparse representation and it has been found that sparse representation is an effective tool for image denoising and dimensionality reduction.[41,42] Sparse representation provides a class of algorithms for finding succinct representations of image data. Sparse representation learns a small set of basis functions (basis vectors) that capture higher-level features buried in the training dataset.[43] Based on the learned basis functions, any new sample can be approximately represented as a weighted linear combination of the learned basis functions. Here, we only briefly introduce the main principles, and the details can be found in Refs. [41–43].

Let $X = \{\boldsymbol{x}_i\}_{i=1}^N$ be the set of training vectors, where $N$ is the number of training samples and $\boldsymbol{x}_i \in \mathbb{R}^{n \times 1}$ ($n = 340$ in this study). The sparse representation aims to find a set of $m$ basis vectors $\{\boldsymbol{b}_j\}_{j=1}^m$, where $\boldsymbol{b}_j \in \mathbb{R}^{n \times 1}$, so as to any $\boldsymbol{x}_i \in X$ can be succinctly represented using basis vectors and a sparse vector weights or coefficients $\mathbf{s} = (s_1, s_2, \cdots, s_m) \in \mathbb{R}^{m \times 1}$ such that $\boldsymbol{x}_i \approx \sum_{j=1}^m \boldsymbol{b}_j \cdot s_j$. The basis vectors could be found by solving the following optimization problem[43]:

$$\begin{aligned} \text{minimize}_{\{\mathbf{b}_j\},\{\mathbf{s}_i\}} \quad & \sum_{i=1}^N \frac{1}{2\sigma^2} \left\| \mathbf{x}_i - \sum_{j=1}^m \mathbf{b}_j s_{ij} \right\|^2 \\ & + \beta \cdot \sum_{i=1}^N \sum_{j=1}^m \phi(s_{ij}) \\ & \text{subject to} \left\| \mathbf{b}_j \right\|^2 \le c, \forall j = 1, \cdots, m. \end{aligned} \quad (2)$$

where $\boldsymbol{s}_i = (s_{i1}, s_{i2}, \cdots, s_{im})$, $\sigma$ is the standard deviation of the reconstruction error, $\beta$ is the sparsity coefficient, and $\phi(\cdot)$ is

the penalty function. In this study, we used $L_1$ penalty function as follows:

$$\phi(s) = \|s\|_1 \tag{3}$$

Once the basis functions are computed, a sparse dictionary **D** can be constructed as follows:

$$\mathbf{D} = [\mathbf{b}_1|\mathbf{b}_2|\cdots|\mathbf{b}_m]_{n\times m} \tag{4}$$

For any new sample **x**, its sparse coefficient vector **y** can be computed as follows:

$$\mathbf{y} = \mathbf{D}^{\mathrm{T}} \cdot \mathbf{x} \tag{5}$$

And the sparse coefficient vector **y** is then used as the sparse representation feature of sample **x** for subsequent classification or prediction. In this study, the number of basis functions is set to be 128 according to our preliminary testing. More specifically, the dimensionality of the sparse representation feature of a residue evolution image is 128-D.

### Dealing with imbalance between binding and non-binding residues.

Clearly, the protein–ATP binding prediction is a typical imbalanced learning problem, i.e., the number of samples in different class differs significantly. For example, in ATP227 dataset, the number of the majority samples (non-binding residues) is more than 20 times of that of the minority samples (binding residues). Previous studies have shown that 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.[44] To circumvent this problem, random under-sampling technique is taken[45] to alter the size of the majority class by randomly removing samples from the majority class. Random under-sampling can provide a parsimonious training dataset since it removes samples from the original dataset. However, part of the important information buried in the removed samples may also be lost simultaneously.

Previous studies[46] have shown that classifier ensemble is a promising route to relieve the impact of information loss caused by random under-sampling. In this study, we exploited the method of combing multiple under-samplings with classifier ensemble and try to further improve the prediction performance of protein–ATP binding sites prediction: first, we sample $L$ different majority training subsets by random under-sampling the majority class $L$ times; then, we train a base classifier on each of the majority training subsets plus the minority training set; finally, the trained base classifiers are ensembled to perform the final decision.

In this study, SVM[34] was used as base classifier and LIBSVM[35] was applied. Here, radial basis function (RBF) was chosen as the kernel function. The two parameters contained in the RBF, i.e., the regularization parameter $\gamma$ and the kernel width parameter $\sigma$ were optimized based on ten-fold cross-validation using a grid search strategy in the LIBSVM software.

As to classifier ensemble, Kuncheva[47] well surveyed many widely used ensemble schemes and pointed out that different ensemble schemes have their own merits and shortcomings, and there does not exist a general "best" ensemble scheme for all kinds of applications. In light of this, we have tested several popular ensemble schemes in this study, i.e., *Maximum* ensemble, *Minimum* ensemble, *Mean* ensemble, *Decision Template* ensemble,[47] *Dempster-Shafer* ensemble,[48] and *AdaBoost* ensemble.[33] The best one, i.e., *AdaBoost* ensemble, was finally chosen.

After classifier ensemble, for each residue to be predicted, the ensembled classifier outputs its possibility for being a protein–ATP binding residue. If the possibility is higher than a predefined threshold $T$, the residue is labeled as binding residue; otherwise, it is labeled as non-binding residue.

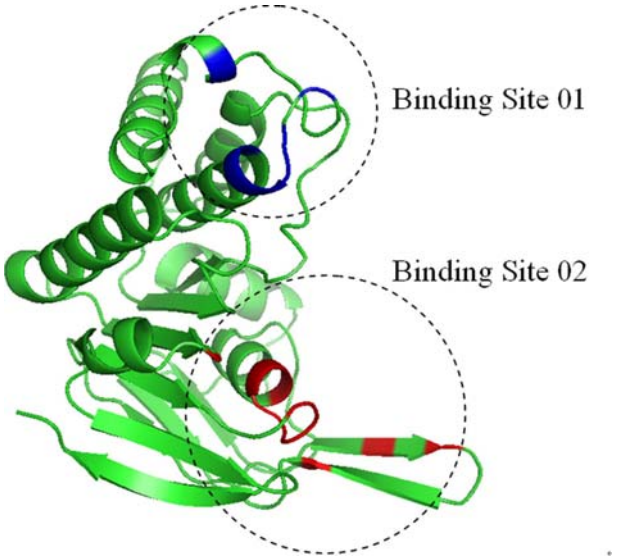



**Figure 2.** Visualization of two binding sites for chain A of protein 1L2T. The pictures were made with PyMOL.[50] [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

### Spatial clustering: identification of pockets from predicted binding residues

To the best of our knowledge, all existing sequence-based ATP-binding predictors, including ATPint,[6] ATPsite,[7] and NsitePred,[29] can only predict the potential binding residues from a given protein sequence. In fact, it will be more useful for biologists and users if the predictor can tell which residues actually form binding site (pocket) for ATP ligand, especially in the situation where there exists more than one binding sites (pockets) in one protein sequence.

Previous studies have shown that residues located in ligand binding interfaces tend to form spatial clusters.[49] Taking chain A of protein 1L2T as an illustration, we drew its 3D structure with cartoon representation as shown in Figure 2, where the blue and red residues are observed ATP-binding residues. From Figure 2, it is clear that the blue ones and red ones are spatial clustered and form binding sites 01 and 02, respectively.

Based on this observation, we thus developed a post-processing procedure to further identify which of the predicted ATP-binding residues may potentially form binding site(s).

Let $C$ be the set of the predicted ATP-binding residues for a given protein sequence, then the residues in $C$ can be

clustered into binding site(s) using the following spatial clustering algorithm based on their spatial coordinates, as shown in Figure 3.

| | |
|---|---|
| **Algorithm** | *BindingSiteClusters* = SpatialClustering ($C$, $T_{Cluster}$) |
| **Input** | $C$: the set of predicted ATP-binding residues; $T_{Cluster}$ – threshold for spatial clustering. |
| **Output** | *BindingSiteClusters* – a set of clusters, residues in each cluster constitute a candidate binding site. |
| 1 | Calculate *max_distance*: the maximal distance between any two residues in $C$. |
| 2 | **IF** the *max_distance* is greater than the pre-defined threshold $T_{Cluster}$ |
| | 2.1 Clustering the residues in $C$ into two smaller clusters according to their spatial positions: *C_First* and *C_Second* |
| | 2.2 *BindingSiteClusters_First* = SpatialClustering(*C_First*, $T_{Cluster}$) |
| | 2.3 *BindingSiteClusters_Second* = SpatialClustering(*C_Second*, $T_{Cluster}$) |
| | 2.4 *BindingSiteClusters* = *BindingSiteClusters_First* ∪ *BindingSiteClusters_Second* |
| | **ELSE** |
| | 2.5 *BindingSiteClusters* = $C$ |
| | **END IF** |
| 3 | **RETURN** *BindingSiteClusters* |

**Figure 3.** Spatial clustering algorithm for clustering predicted binding residues into binding sites. The residue coordinates are from the protein 3D structures either provided by the user or modeled by the MODELLER software.[36]

Note that in the spatial clustering algorithm, the only parameter $T_{Cluster}$ is a threshold determining how many clusters will be obtained. Clearly, a large $T_{Cluster}$ will produce small number of clusters (binding sites); while a small $T_{Cluster}$ will lead to a large number of clusters (binding sites). Thus, how to set an appropriate $T_{Cluster}$ is crucial for enabling the spatial clustering algorithm to work well. Here, we present a possible solution: Let $R_{avg}$ be the averaged distance between the ATP-binding residues and the centers of their corresponding ATPs. We calculated that $R_{avg}$ is about 11 Å in both ATP168 and ATP227; Thus, the $T_{Cluster}$ can be initialized as $T_{Cluster} = \alpha \cdot (2 \cdot R_{avg})$, where $\alpha$ is a coefficient which controls the maximal width of cluster. We empirically tested different $\alpha$ and found that the best clustering performance was achieved when $\alpha = 1.25$, i.e., $T_{cluster} = 27.5$ Å.

## Evaluation procedure

**Cross-validation.** In this study, five-fold cross-validation was performed on both the two benchmark datasets for evaluating the performance of the proposed method. In the present study, our purpose is to predict whether a residue is a binding residue or not. However, if applying the residue-based cross-validation procedure, residues in all the training protein sequences will be randomly partitioned into five disjoint subsets; then, one subset was used for testing and the remaining four subsets were used for training; this practice continued until all the four subsets of the dataset were traversed over. The above procedure can yield the following phenomenon that testing and training residues may originate from the same protein sequence, which could make the predictor over-fitted. In light of this, we perform a sequence-based cross-validation in current study, i.e., training protein sequences are firstly randomly partitioned into five disjoint subsets; then, one subset was used for testing and the remaining four subsets were used for training; this practice continued until all the four subsets of the dataset were traversed over.

**Evaluation indexes.** Four routinely used evaluation indexes in this filed, i.e., Specificity (*Spe*), Sensitivity (*Sen*), Accuracy (*Acc*), and the Matthews correlation coefficients (MCC) were taken to evaluate the performance of the TargetATPsite as defined:

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} \tag{6}$$

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{7}$$

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{8}$$

$$\text{MCC} = \frac{\text{TP} \cdot \text{TN} - \text{FP} \cdot \text{FN}}{\sqrt{(\text{TP} + \text{FP}) \cdot (\text{TP} + \text{FN}) \cdot (\text{TN} + \text{FP}) \cdot (\text{TN} + \text{FN})}} \tag{9}$$

where TP, FP, TN, and FN are the abbreviations of true positive, false positive, true negative, and false negative, respectively. In addition to these four threshold-dependent evaluation indexes, we also exploited another evaluation index AUC, which is the area under the Receiver Operating Characteristic (ROC) curve and is threshold-independent and increases in direct proportion to the prediction performance.

The prediction performance of a predictor can be represented by a confusion matrix (contingency table), as illustrated in Figure 4. Gradually, adjusting prediction threshold will produce a series of confusion matrices. From each confusion matrix, a ROC point, whose coordinate is (FP/(FP+TN), TP/(FN+TP)), can be calculated. A series of ROC points constitute the ROC curve. Figure 5 illustrates an exemplary ROC curve.

| | | Predicted class: binding | Predicted class: Non-binding |
|---|---|---|---|
| True class | binding | TP (True Positives) | FN (False Negatives) |
| | non-binding | FP (False Positives) | TN (True Negatives) |

**Figure 4.** Confusion matrix for performance evaluation.

Next, let's consider how to choose an appropriate threshold $T$ for reporting threshold-dependent evaluation indexes. As stated in the section "Dealing with imbalance between binding and non-binding residues", for each residue to be predicted, the ensembled classifier will output its possibility (a real number between 0 and 1) for belonging to the protein–ATP binding residue. If the possibility is higher than a predefined threshold $T$, the residue is labeled as binding residue; otherwise, it is labeled as non-binding residue. Clearly, a smaller $T$ will cause higher false positive rate (FPR = FP/(FP+TN)); while a bigger $T$ will lead to higher false negative rate (FNR = FN/(FN+TP)).

Under the imbalanced learning scenario as in this study, over pursuing the overall accuracy is not reasonable and can be deceiving for evaluating the performance of a predictor/classifier. Taking ATP227 (3,393 binding residues and 80,409

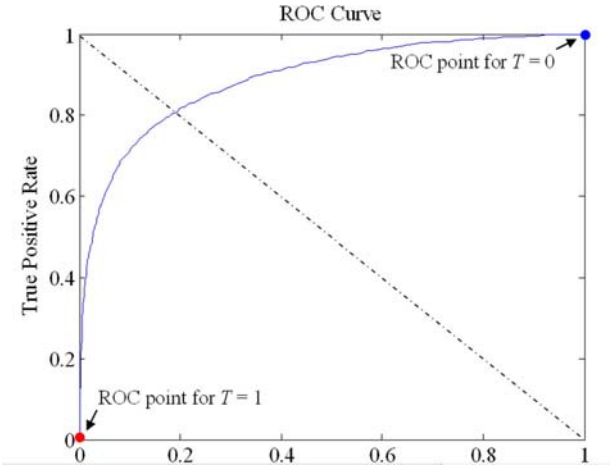



**Figure 5.** An exemplary ROC curve for performance evaluation. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

non-binding residues) as an example, supposing we choose a big $T$ (e.g. $T = 1$) by which the predictor predicts all the residues to be non-binding residues. The resulting confusion matrix is

$$\begin{bmatrix} 0 & 3,393 \\ 0 & 80,490 \end{bmatrix} \tag{10}$$

and the corresponding ROC point is the red circle as shown in Figure 5. Obviously, the predictor is meaningless as none of the binding residues can be correctly identified. However, the overall accuracy is still very high (80,409 / (80,409 + 3,393) = 96.8%).

Another extreme situation appears when choosing a small enough $T$ (e.g. $T = 0$), by which the predictor predicts all the residues to be binding ones. In this case, although all the binding residues can be correctly identified, the non-binding residues are also mistakenly identified as binding ones simultaneously. The resulting confusion matrix is

$$\begin{bmatrix} 3,393 & 0 \\ 80,490 & 0 \end{bmatrix} \tag{11}$$

and the corresponding ROC point is the blue circle in Figure 5. The overall accuracy is only 3,393 / (80,409 + 3,393) = 4.1%.

In view of this, the threshold which maximizes the MCC value of the predictions on the training folds is used to report the results, as done in ATPsite[7] and NsitePred.[29] In the presented study, the thresholds were identified to be 0.57 and 0.70 over five-fold cross-validation on ATP168 and ATP227, respectively.

## Results and Discussions.

### Sparse representation can extract more discriminative features

Table 1 compares the discriminative performance between the PSSM feature and sparse representation feature extracted from residue evolution image on ATP168 and ATP227. It was found that the sparse representation feature consistently outperforms the PSSM feature on both datasets concerning the five evaluation indexes. Taking results on ATP227 as an example, the *Sen*, MCC, and AUC of the sparse representation feature are 43.3%, 0.50, and 0.872, which are about 5%, 6%, and 1% better than that of PSSM feature, respectively. As to other two evaluation indexes, the sparse representation feature also slightly outperforms the PSSM feature. From Table 1, we can find that sparse representation can help to reduce the noises in the residue evolution images derived from the multiple sequence alignments and extract more discriminative features on the tested benchmark datasets.

**Table 1.** Performance comparison between the PSSM feature and sparse representation features on ATP168 and ATP227 with a single SVM classifier (no ensemble) over five-fold cross-validation.

| Dataset | Feature | *Sen* (%) | *Spe* (%) | *Acc* (%) | MCC | AUC |
|---|---|---|---|---|---|---|
| ATP168 | PSSM | 32.1 | 98.8 | 95.4 | 0.42 | 0.847 |
| | Sparse representation[a] | 37.9 | 98.9 | 95.7 | 0.48 | 0.851 |
| ATP227 | PSSM | 38.2 | 98.6 | 96.1 | 0.44 | 0.861 |
| | Sparse representation[a] | 43.3 | 98.8 | 96.5 | 0.50 | 0.872 |

[a] Refer to eq. (5).

### Classifier ensemble helps to further improve prediction performance

As severe imbalance exists between the majority class and minority class, random under-sampling technique is taken to balance the number of samples in majority class and minority class so as to enable the traditional statistical machine-learning algorithms, which assume that samples in different classes are balanced, to be appropriately applied. However, the information contained in the discarded majority samples will be lost. To remedy the disadvantage, classifier ensemble is utilized as described in the section "Dealing with imbalance between binding and non-binding residues". In this study, we trained $L$ independent base SVMs on the $L$ randomly under-sampled datasets and then ensembled them with AdaBoost ensemble scheme. Note that in the presented results, the number of base SVM classifiers ($L$) was set to be 5.

Table 2 lists the prediction results with and without AdaBoost ensemble on datasets ATP168 and ATP227 over five-fold

**Table 2.** Prediction results with and without AdaBoost ensemble on ATP168 and ATP227 over five-fold cross-validation.

| Dataset | Ensemble type | *Sen* (%) | *Spe* (%) | *Acc* (%) | MCC | AUC |
|---|---|---|---|---|---|---|
| ATP168 | No ensemble[a] | 37.9 | 98.9 | 95.7 | 0.48 | 0.851 |
| | AdaBoost ensemble[a] | 39.2 | 98.9 | 95.8 | 0.49 | 0.860 |
| ATP227 | No ensemble[a] | 43.3 | 98.8 | 96.5 | 0.50 | 0.872 |
| | AdaBoost ensemble[a] | 44.5 | 98.9 | 96.6 | 0.52 | 0.881 |

[a] Inputs are the residue images encoded by sparse representation of eq. (5).

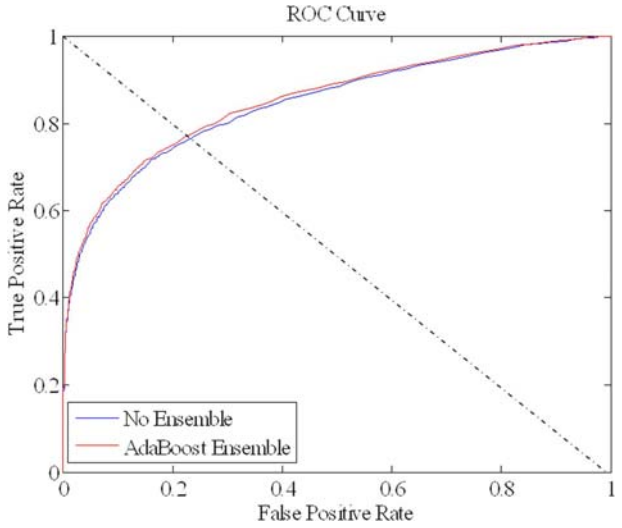



**Figure 6.** ROC curves of no ensemble and AdaBoost ensemble on ATP227. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

cross-validation. Figure 6 illustrates the ROC curves of a single SVM (no ensemble) and AdaBoost ensembled SVMs on ATP227 over five-fold cross-validation. From Table 2, we can find that the prediction performances are further improved after classifier ensemble. The considered five evaluation indexes with AdaBoost ensemble are consistently better than those without classifier ensemble.

### Comparison with existing sequence-based predictors

In this section, we compare the proposed TargetATPsite with three most recently released sequence-based protein–ATP binding residue predictors, i.e., ATPint,[6] ATPsite,[7] and NsitePred.[29]

**Comparison on dataset ATP168.** ATPint[6] is the first predictor that was specifically designed for predicting protein–ATP binding residues from protein primary sequence. The ATPint was developed based on PSSM-based feature and the SVM was used to perform classification. In Ref. [6], Chauhan et al. tried several different thresholds, and the threshold, where sensitivity and specificity are nearly equal in order to make the balance between sensitivity and specificity, was chosen for the final reporting. To fairly compare with ATPint, we also performed five-fold cross-validation on ATP168 and reported results similar to that in ATPint.

Table 3 illustrates the comparison results between TargetATPsite and ATPint on ATP168 over five-fold cross-validation.

**Table 3.** Performance comparison between TargetATPsite and ATPint on ATP168 over five-fold cross-validation.

| Predictor | *Sen* (%) | *Spe* (%) | *Acc* (%) | MCC | AUC |
|---|---|---|---|---|---|
| ATPint[a] | 74.4 | 75.8 | 75.1 | 0.25 | 0.823 |
| TargetATPsite | 78.2 | 78.4 | 78.4 | 0.29 | 0.860 |

[a] Data excerpted from Ref. [6].

**Table 4.** Performance comparison of the TargetATPsite with ATPint, ATPsite, and NsitePred over five-fold cross-validation on ATP227.

| | | | | MCC | | AUC | |
|---|---|---|---|---|---|---|---|
| Predictor | *Sen* (%) | *Spe* (%) | *Acc* (%) | Value | Sig. | Value | Sig. |
| ATPint[a] | 53.9 | 65.1 | 64.8 | 0.08 | + | 0.627 | + |
| ATPsite[a] | 36.1 | 98.8 | 96.2 | 0.43 | + | 0.854 | + |
| NsitePred[a] | 44.4 | 98.2 | 96.0 | 0.46 | + | 0.861 | + |
| TargetATPsite | 44.5 | 98.9 | 96.6 | 0.52 | | 0.881 | |

[a] Data excerpted from Ref. [29]. The significance of the differences between TargetATPsite and the other predictors are measured for the MCC and AUC and they are given in the '*Sig.*' columns. The '+' means that the TargetATPsite is statistically significantly better with $p$-value $< 0.05$.

Note that the threshold for reporting Table 3 is identified to be 0.10, by which the value of *Sen* is roughly equal to that of *Spe* as done in ATPint. From Table 3, it is easy to find that the TargetATPsite outperforms the ATPint on all the five evaluation indexes and an averaged improvement of 3–4% was obtained on each of the five considered evaluation indexes.

**Comparison on dataset ATP227.** The performances of the ATPsite and NsitePred were reported based on ATP227 over five-fold cross-validation.[7,29] To fairly compare the TargetATPsite with them, we also performed five-fold cross-validation on the same dataset and reported the results, as shown in Table 4.

By observing Table 4, we found that the TargetATPsite performs the best among all the listed predictors including the NsitePred, which is the most recently released protein–ATP binding residues predictor. We also analyzed statistical significance of the differences in the MCC and AUC values between predictions generated by TargetATPsite and the other three predictors using a paired $t$-test.[51] If the resulting $p$-value is below the desired significance level (0.05 in this study), the performance difference between two methods is considered to be statistically significant. By this test, we found that the MCC and AUC of the TargetATPsite are statistically better than that of all the listed predictors.

**Comparison on independent testing dataset.** Independent dataset test is often considered as an effective method to validate the generalization ability of a predictor. However, how to select the independent samples to test the predictor is very important. Testing a predictor with inappropriate independent dataset tends to obtain over-optimistic evaluation results. Taking this study as an example, if the proteins in the independent dataset have close homology with those proteins in the training dataset, we will definitely obtain good prediction results.

Chen et al.[29] have considered this point when constructing the independent dataset: for each protein sequence in the independent testing dataset they constructed, it shares $<40\%$ identity to any sequence in benchmark dataset ATP227. In view of this, we trained our TargetATPsite on ATP227 and then the independent dataset[29] was used to evaluate the generalization ability of the TargetATPsite. Table 5 lists the performance comparison of different predictors on the independent

**Table 5.** Performance comparison of the TargetATPsite with three most recently released protein–ATP binding residues predictors on independent testing dataset.

| Predictor | *Sen* (%) | *Spe* (%) | *Acc* (%) | MCC | AUC |
|---|---|---|---|---|---|
| ATPint[a] | 51.2 | 66.0 | 65.5 | 0.07 | 0.606 |
| ATPsite[a] | 36.7 | 99.1 | 96.9 | 0.45 | 0.868 |
| NsitePred[a] | 46.0 | 98.5 | 96.7 | 0.48 | 0.875 |
| TargetATPsite | 45.8 | 99.1 | 97.2 | 0.53 | 0.882 |

[a] Data excerpted from Ref. [29].

dataset. From Table 5, we can find that the TargetATPsite achieves satisfactory results with the best threshold independent AUC value of 0.882 and the best MCC value of 0.53. As to other three evaluation indexes, TargetATPsite also performs the best except for *Sen* which is slightly less than that of NsitePred. This experiment demonstrates that TargetATPsite has good generalization capability for ATP-binding residues prediction.

### Performance of spatial clustering

As stated in the section "Spatial clustering: identification of pockets from predicted binding residues", existing sequence-based protein–ATP binding predictors can only predict the binding residues. In TargetATPsite of this article, we further performed spatial clustering on the predicted binding residues to identify which binding residues may form binding site(s). Two measures were used to evaluate the performance of spatial clustering on the predicted binding residues.

The first measure is $V_{site}$, which measures the percentage of the observed binding sites in the testing dataset that have been correctly predicted. In this study, an observed binding site is considered to be correctly predicted if its 30% binding residues are included in the predicted binding site. The second measure is $V_p$, which measures the percentage of proteins in the testing dataset that have been correctly predicted. A protein is considered being correctly predicted if all the binding sites in this protein are correctly predicted and the number of the predicted binding sites is also equal to the number of the observed binding sites in the protein. We calculated that the values of $V_{site}$ and $V_p$ on dataset ATP227 over five-fold cross-validation are 66% and 53%, respectively. Although the results have space to be further improved, they are encouraging especially considering they are derived from the *ab initio* predictions with the sequences alone without using any homology complex structures in the PDB. It is expected to be particularly useful in the following two conditions: (1) Protein sequences with no solved structures. In this case, we can firstly predict the potential ATP-binding residues using the above TargetATPsite protocol, and then perform the developed spatial clustering algorithm on a modeled 3D structure from the state-of-the-art algorithms like Rosetta, MODELLER, and I-TASSER.[52–54] In this article, MODELLER is used for this purpose. (2) Hard targets with no or very few homology protein–ATP complex structures in the PDB. For some hard targets, if we cannot find any homology protein–ligand complex structure in current database, the homology template-based ATP site detection approach cannot be applied. In this case, current approach is expected to play an important complementary role for correctly predicting the binding residues and the pockets. As ATPint and NsitePred do not provide the capability of detecting the binding pockets, we do not compare the results with them on finding pockets.

### Online implementation

The system architecture of the proposed TargetATPsite is illustrated in Figure 7. The final online implementation was built on dataset ATP227 and is freely available at: http://www.csbio.sjtu.edu.cn/bioinf/TargetATPsite/

The TargetATPsite server accepts two different types of query protein information for protein–ATP binding sites prediction: one is protein sequence in FASTA format; the other is

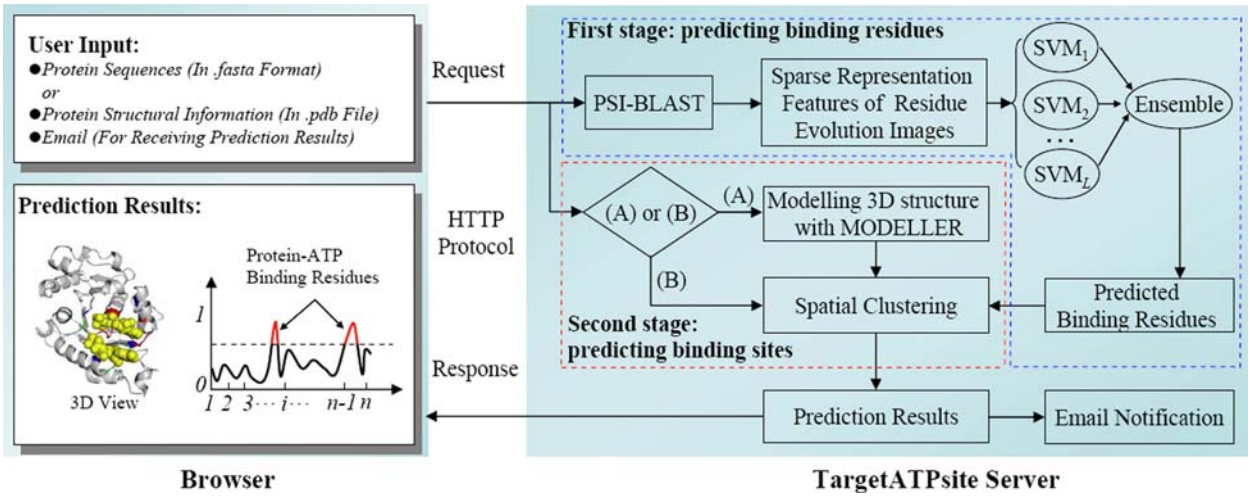



**Figure 7.** System architecture of the TargetATPsite predictor. (A) denotes that user submits protein sequence; (B) denotes that user submits a PDB file. MODELLER[36] is a software package for predicting 3D structure from sequences. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

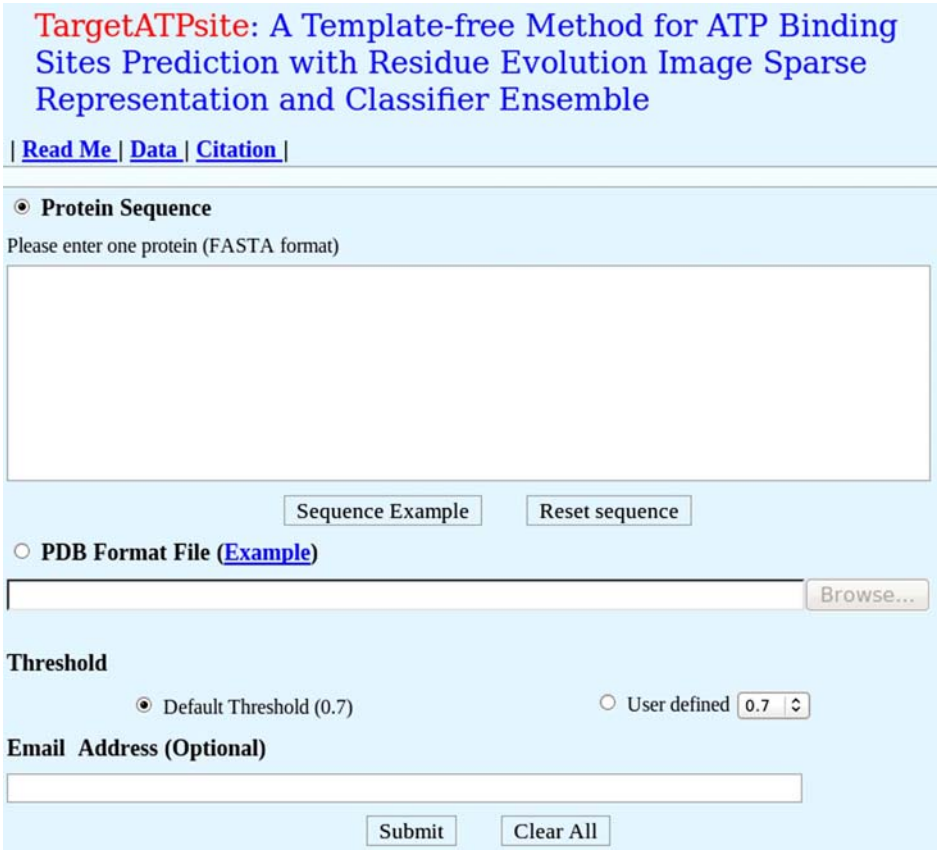



**Figure 8.** Illustration to show the TargetATPsite Web page at http://www.csbio.sjtu.edu.cn/bioinf/TargetATPsite/. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

standard PDB file format, which contains 3D structure information of a protein. For each protein (sequence or PDB file) submitted from the client, the server performs prediction with a two-stage scheme: in the first stage, the server predicts which residues are protein–ATP binding residues; while in the second stage, the server further identifies binding sites from the predicted binding residues with spatial clustering algorithm. After the two-stage prediction, the server returns the prediction

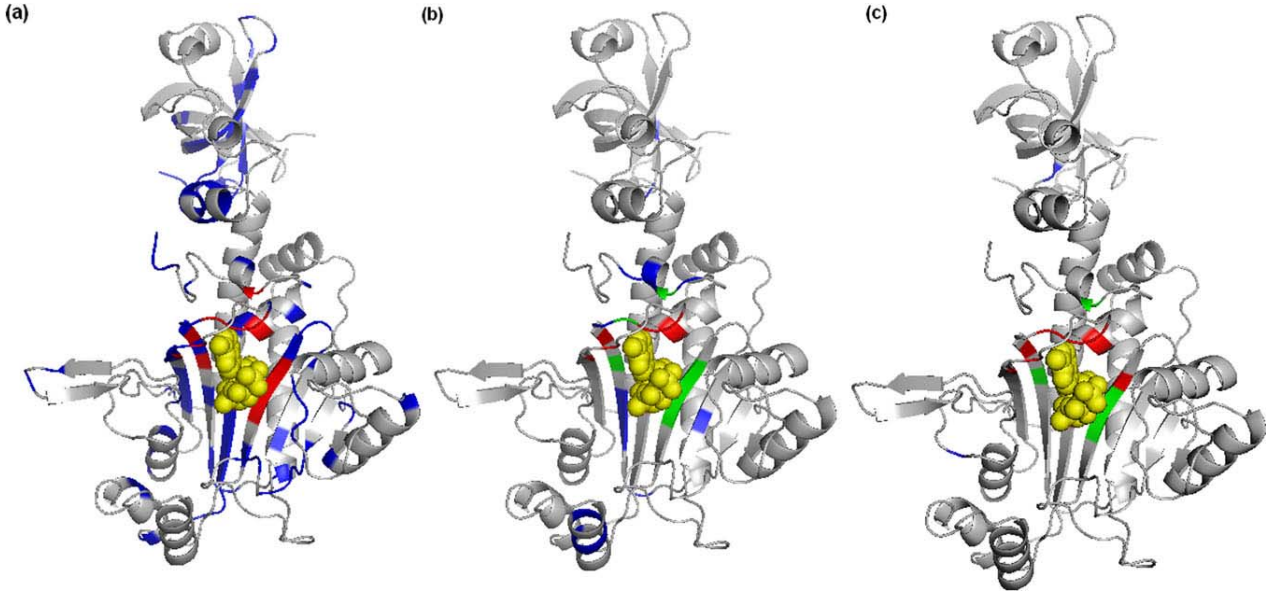



**Figure 9.** Visualization of prediction results for 2XTIB. (a) ATPint, (b) NsitePred, and (c) TargetATPsite. The following color scheme is used: ATP in yellow, true positives in red, false positives in blue, false negatives in green. The pictures were made with PyMOL.[50]

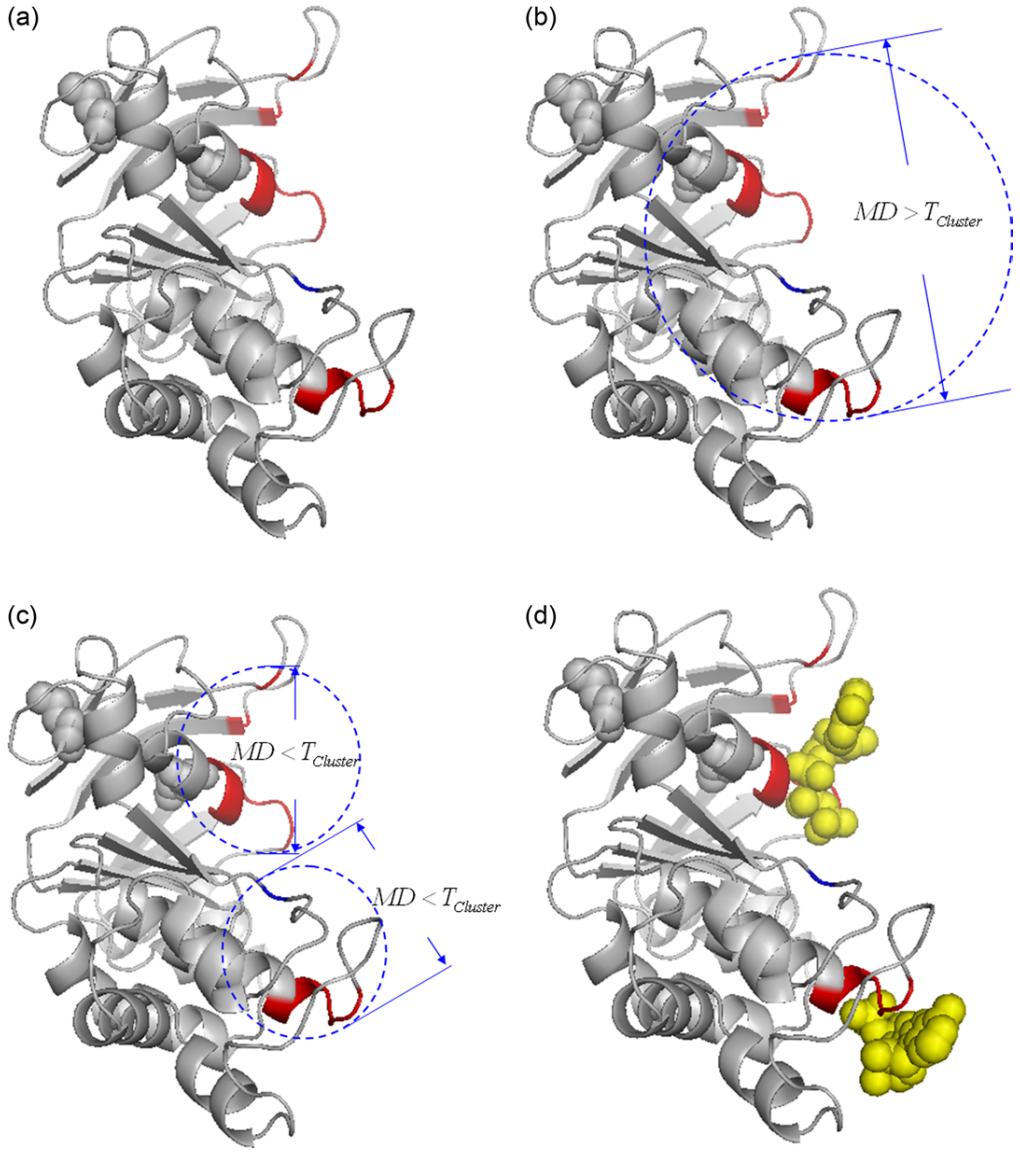



**Figure 10.** Visualization of spatial clustering procedure on the predicted binding residues for 1XEFA. (a) Initial predicted binding residues, (b) Cluster all the predicted binding residues in one cluster, (c) Split the cluster into two smaller clusters, and (d) Final predicted binding sites. The following color scheme is used: ATP in yellow, true positives in red, false positives in blue. The pictures were made with PyMOL.[50]

results back to the client in two different ways: online real-time feed back with 3D illustrations and text descriptions, and independent email notifications (optional) to the email address provided by the user.

Note that if the user submits a PDB file, then the residue 3D coordinates contained in the PDB file can be directly utilized for spatial clustering, as denoted by (B) in Figure 7. If user only submits a protein sequence, the 3D structure of the query sequence will be first modeled by applying MODELLER[36] software package, and then the predicted 3D structure is used for spatial clustering, as denoted by (A) in Figure 7.

Next, we briefly introduce how to use TargetATPsite.

Step 1. Open the Web page http://www.csbio.sjtu.edu.cn/bioinf/TargetATPsite/ and you will see the top page of the TargetATPsite on your computer screen, as shown in Figure 8.

Step 2. If you have protein sequences, either type or copy and paste the query protein sequences into the input box (depicted by the box at the top of Figure 8). The input sequences should be in FASTA format, as shown by clicking on the Sequences Example button below the input box. If you have a PDB file, first click the PDB Format File radio button, then click Browse button to locate the PDB file. After inputting protein sequences or PDB file, you can also input your email address (optional) to receive an email notification of your future prediction results.

Step 3. Click on the Submit button, the protein information you inputted will be sent to TargetATPsite server for prediction and the predicted results will be delivered back to your browser after a few minutes. For each protein to be predicted, the outputted prediction results consist of the following seven

components: sequence name, sequence, prediction threshold, predicted binding residues, predicted binding sites, 3D view, and probability of being protein–ATP binding for each residue.

### Case studies

To further demonstrate the effectiveness of the proposed predictor, we take two ATP binding proteins that are not included in the training dataset of our online prediction system for case studies. The first protein is 2XTIB which has only one binding site (pocket); while the second other one is 1XEFA which has two binding sites (pockets).

The prediction results of ATPint were obtained by feeding the two sequences to the web server that is available at: http://www.imtech.res.in/raghava/atpint/. As ATPsite does not provide a web server, thus it is not included in this section. The prediction results of NsitePred were obtained by feeding the two sequences to the web server that is available at: http://biomine.ece.ualberta.ca/nSITEpred/. Note that we fed the sequence of 2XTIB and the PDB file of 1XEFA to the TargetATPsite respectively to demonstrate that TargetATPsite can predict ATP binding sites from both the protein sequence and the protein 3D structure information.

The prediction results of 2XTIB generated by ATPint, NsitePred, and TargetATPsite are illustrated in Figure 9. From Figure 9, it is easy to find that the TargetATPsite and NsitePred significantly outperform the ATPint. The ATPint predicted too many false positives (31 for 2XTIB), thus the predicted results cannot be mapped easily to the binding pockets. TargetATPsite correctly predicted 10 out of the 15 binding residues and only two non-binding residues are mistakenly identified as binding residues (two false positives). In contrast, the NsitePred correctly identified only 8 out of the 15 binding residues while with 16 false positives.

As for 1XEFA, TargetATPsite also performs best with the prediction results of 15 true positives, 1 false positive, and 1 false negative (ATPint: 12 true positives, 40 false positives, and 4 false negatives; NsitePred: 15 true positives, 4 false positives, and 1 false negative). In addition, TargetATPsite correctly identified the two pockets from the 16 predicted binding residues (15 true positives and 1 false positive) by applying the proposed spatial clustering algorithm. Figure 10 illustrates the spatial clustering procedure. In Figure 10a, all the predicted binding residues, i.e., the 15 true positives and 1 false positive, are labeled in red and blue color, respectively. When applying spatial clustering algorithm, all the predicted binding residues are initially clustered into one cluster as shown in Figure 10b; then, the maximal distance (MD) between any two residues in the cluster is calculated; as the MD is larger than $T_{Cluster}$, the cluster is splitted into two smaller clusters as shown in Figure 10c; it is found that the MDs of the two splitted clusters are all smaller than $T_{Cluster}$, thus the spatial clustering procedure terminates and the residues in the two clusters form two ATP-binding sites as shown in Figure 10d.

## Conclusions

In this study, a sequence-based template-free protein–ATP binding site predictor, named TargetATPsite, is proposed. Evolutionary information derived from PSSM is considered as image and further processed by sparse representation to obtain more discriminative features. To effectively deal with the intrinsic imbalance between the positive and negatives samples, random under-sampling and classifier ensemble techniques are integrated. Experimental results on different datasets demonstrate that the TargetATPsite is better than the existing state-of-the-art sequence-based predictors for predicting the binding residues. In addition, compared with the existing predictors, the TargetATPsite is featured by the capability of reporting binding pockets from the predicted binding residues with a spatial clustering process. Our work enriches the contents of the protein–ATP binding sites prediction, which is anticipated to become a useful tool in the area of *in silico* identification of protein–ATP binding sites. TargetATPsite is freely available for academic use at: http://www.csbio.sjtu.edu.cn/bioinf/TargetATPsite/

## Acknowledgments

*The authors wish to thank the two anonymous reviewers for valuable suggestions and comments, which were very helpful for improvement of this article.*

**Keywords:** protein functional annotation · protein–ATP binding sites prediction · residue evolution image · sparse representation · classifier ensemble

How to cite this article: D.-J. Yu, J. Hu, Y. Huang, H.-B. Shen, Y. Qi, Z.-M. Tang, J.-Y. Yang, *J. Comput. Chem.* **2013,** *34,* 974–985. DOI: 10.1002/jcc.23219

[1] B. Alberts, Ed. Molecular Biology of the Cell, 5th ed., Garland Science: New York, **2008**.
[2] M. Gao, J. Skolnick, *Proc. Natl. Acad. Sci. USA.* **2012**, *109,* 3784.
[3] H. Kokubo, T. Tanaka, Y. Okamoto, *J. Comput. Chem.* **2011**, *32,* 2810.
[4] P. Schmidtke, X. Barril, *J. Med. Chem.* **2010**, *53,* 5858.
[5] H.M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T.N. Bhat, H. Weissig, I.N. Shindyalov, P.E. Bourne, *Nucleic. Acids. Res.* **2000**, *28,* 235.
[6] J.S. Chauhan, N.K. Mishra, G.P. Raghava, *BMC Bioinformatics.* **2009**, *10,* 434.
[7] K. Chen, M.J. Mizianty, L. Kurgan, *Proteome. Sci.* **2011**, *9 Suppl 1,* S4.
[8] Y.N. Zhang, D.J. Yu, S.S. Li, Y.X. Fan, Y. Huang, H.B. Shen, *BMC Bioinformatics.* **2012**, *13,* 118.
[9] A. Roy, Y. Zhang, *Structure.* **2012**, *20,* 987.
[10] M. Brylinski, J. Skolnick. *PLoS. Comput. Biol.* **2009**, *5,* e1000405.
[11] R. Liu, J. Hu, *BMC Bioinformatics.* **2011**, *12,* 207.
[12] D.G. Levitt, L.J. Banaszak, *J. Mol. Graph.* **1992**, *10,* 229.
[13] M. Hendlich, F. Rippmann, G. Barnickel, *J. Mol. Graph. Model.* **1997**, *15,* 359.
[14] R.A. Laskowski, *J. Mol. Graph.* **1995**, *13,* 323.
[15] V. Le Guilloux, P. Schmidtke, P. Tuffery, *BMC Bioinformatics.* **2009**, *10,* 168.
[16] A. Armon, D. Graur, N. Ben-Tal, *J. Mol. Biol.* **2001**, *307,* 447.
[17] T. Pupko, R.E. Bell, I. Mayrose, F. Glaser, N. Ben-Tal, *Bioinformatics.* **2002**, *18 Suppl 1,* S71.
[18] Y. Dou, J. Wang, J. Yang, C. Zhang, *PLoS One.* **2012**, *7,* e35666.
[19] B. Huang, M. Schroeder, *BMC. Struct. Biol.* **2006**, *6,* 19.
[20] J.A. Capra, R.A. Laskowski, J.M. Thornton, M. Singh, T.A. Funkhouser, *PLoS. Comput. Biol.* **2009**, *5,* e1000585.
[21] F. Glaser, R.J. Morris, R.J. Najmanovich, R.A. Laskowski, J.M. Thornton, *Proteins.* **2006**, *62,* 479.
[22] A.T. Laurie, R.M. Jackson, *Bioinformatics.* **2005**, *21,* 1908.

[23] M. Hernandez, D. Ghersi, R. Sanchez, *Nucleic. Acids. Res.* **2009**, *37(Web Server issue),* W413.
[24] S. Henrich, O.M. Salo-Ahen, B. Huang, F.F. Rippmann, G. Cruciani, R.C. Wade, *J. Mol. Recognit.* **2010**, *23,* 209.
[25] J.S. Sodhi, K. Bryson, L.J. McGuffin, J.J. Ward, L. Wernisch, D.T. Jones, *J. Mol. Biol.* **2004**, *342,* 307.
[26] M. Brylinski, J. Skolnick, *Proteins* **2011**, *79,* 735.
[27] M. Kumar, A.M. Gromiha, G.P.S. Raghava, *Proteins—Struct. Func. Bioinformatics.* **2008**, *71,* 189.
[28] R. Liu, J. Hu, *PLoS. One.* **2011**, *6,* e25560.
[29] K. Chen, M.J. Mizianty, L. Kurgan, *Bioinformatics.* **2012**, *28,* 331.
[30] N. Hirokawa, R. Takemura, *Trends. Biochem. Sci.* **2003**, *28,* 558.
[31] N. Gresh, G.B. Shi, *J. Comput. Chem.* **2004**, *25,* 160.
[32] J. Ito, J.L. Heazlewood, A.H. Millar, *J. Proteome. Res.* **2006**, *5,* 3459.
[33] Y. Freund, R.E. Schapire, *J. Comp. Syst. Sci.* **1997**, *55,* 119.
[34] V.N. Vapnik, Ed. Statistical Learning Theory; Wiley-Interscience: New York, **1998**.
[35] R.E. Fan, P.H. Chen, C.J. Lin, *J. Machine Learning Res.* **2005**, *6,* 1889.
[36] M.A. Marti-Renom, A.C. Stuart, A. Fiser, R. Sanchez, F. Melo, A. Sali, *Annu. Rev. Biophys. Biomol. Struct.* **2000**, *29,* 291.
[37] R.A. Bauer, S. Gunther, D. Jansen, C. Heeger, P.F. Thaben, R. Preissner, *Nucleic. Acids. Res.* **2009**, *37(Database issue),* D195.
[38] W. Li., A. Godzik, *Bioinformatics* **2006**, *22,* 1658.
[39] A.A. Schaffer, *Nucleic. Acids. Res.* **2001**, *29,* 2994.
[40] T. Acharya, A.K. Ray, Image Processing: Principles and Applications, John Wiley: Hoboken, N.J., **2005**.
[41] B.V. Gowreesunker, A.H. Tewfik, *IEEE. T. Signal. Process.* **2010**, *58,* 3055.
[42] J. Wright, Y. Ma., J. Mairal, G. Sapiro, T.S. Huang, S.C. Yan. *P IEEE* **2010**, *98,* 1031.
[43] H. Lee, A. Battle, R. Raina, A.Y. Ng, In Proceedings of the 19th Annual Conference on Neural Information Processing Systems, MIT Press, Cambridge, MA, USA, **2007**, pp. 801–808.
[44] H. Haibo, E.A. Garcia, *IEEE T. Knowl. Data. En.* **2009**, *21,* 1263.
[45] Z.Y. Lin, Z.F. Hao, X.W. Yang, X.L. Liu, In ADMA, R. Huang., Ed.; Springer-Verlag: Berlin Heidelberg, **2009**, pp. 536–554.
[46] X.Y. Liu, J.X. Wu, Z.H. Zhou, *IEEE. T. Syst. Man. Cy. B.* **2009**, *39,* 539.
[47] L.I. Kuncheva, Combining Pattern Classifiers: Methods and Algorithms; John Wiley & Sons, The United States of America, **2004**.
[48] G. Rogova, *Neural Netw.* **1994**, *7,* 777.
[49] O. Schueler-Furman, D. Baker, *Proteins.* **2003**, *52,* 225.
[50] The PyMOL Molecular Graphics System, Available at: http://www.pymol.org. Accessed on 26 November 2012.
[51] J. Yang, L. Zhang, J.Y. Yang, D. Zhang, *Pattern. Recogn.* **2011**, *44,* 1387.
[52] N. Eswar, D. Eramian, B. Webb, M.Y. Shen, A. Sali, *Methods. Mol. Biol.* **2008**, *426,* 145.
[53] R. Das, D. Baker, *Annu. Rev. Biochem.* **2008**, *77,* 363.
[54] A. Roy, A. Kucukural, Y. Zhang, *Nat. Protoc.* **2010**, *5,* 725.

Received: 24 July 2012
Revised: 9 November 2012
Accepted: 7 December 2012
Published online on 3 January 2013