Disulfide Bond Prediction with Hybrid Models*

Chong-Jie Wang, Chang-Biau Yang† and Chiou-Yi Hor
Department of Computer Science and Engineering
National Sun Yat-sen University, Kaohsiung 80424, Taiwan
†Email: cbyang@cse.nsysu.edu.tw

Kuo-Tsung Tseng‡
Department of Information Management
Fooyin University, Kaohsiung 83102, Taiwan
‡Email: ft051@mail.fy.edu.tw

***Abstract*—Disulfide bonds are special covalent cross links between two cysteines in a protein. This kind of bonding state plays an important role in protein folding and stabilization. For connectivity pattern prediction, it is a very difficult problem because of the fast growth of possible patterns with respect to the number of cysteines. For the prediction, in this paper, we propose a hybrid approach based on support vector machine (SVM). With this approach, we can improve the prediction accuracies by selecting appropriate models. In order to evaluate the performance of our method, we apply the method with 4-fold cross-validation on SP39 dataset, which contains 446 proteins. We achieve accuracies with 70.8% and 65.9% for pair-wise and pattern-wise prediction respectively, which is better than the previous works.**

Keywords— disulfide bond, cysteines, hybrid model, SVM, prediction

I. Introduction

A *disulfide bond*, also called *SS-bond* or *SS-bridge* in chemistry, is a covalent cross-link formed by coupling two thiol groups. It plays an important role in protein folding process. In addition, it has a great help to stabilize the 3D structure of a protein because it imposes geometrical constraints on the protein backbones. Although there are two kinds of amino acids belonging to thiol groups, however, only *cysteines* can make up disulfide bonds. According to this property, it becomes easier to do the task of the predicting disulfide bonds.

Among disulfide bonding researches, the *connectivity prediction problem*[15] has been an intensively studied. Its main purpose is to find out the disulfide bonded pairs among all possible candidates in a protein. This work is often divided into two parts: pair-wise and pattern-wise. The former one aims to find the bonded probabilities among all possible pairs while the latter one manages to determine the unique connectivity pattern. For the pattern-wise version, it is difficult to get an exactly correct solution because the connectivity patterns grow rapidly with respect to the number of cysteines in a protein. Suppose that a protein contains M disulfide bridges, it means that at least $2M$ cysteines are contained in the protein. Since two cysteines form a pair, the number N of all possible pairs can be determined by the following formula:

$$N = C_2^{2M} = M(2M-1). \quad (1)$$

*This research work was partially supported by the National Science Council of Taiwan under contract NSC 100-2221-E-242-003.

Moreover, the number P of all possible connectivity patterns is given as follows:

$$P = (2M-1)(2M-3)(2M-5)\cdots 1 = (2M-1)!! \quad (2)$$

For example, we get P = 945 for M = 5, and we get P = 10395 when M = 6. Clearly, P grows up rapidly with the increase of M. Therefore, the prediction work of connectivity patterns is not so easy.

In most of recent researches, SVM (*support vector machine*) models have become popular to be built for classification or prediction [4, 5, 8, 9, 12–14, 17, 18]. In 2005, Tsai *et al.*[17] achieved an accuracy of 63% with SVM. In 2009, Chung [6] combined the down-sampling scheme with SVM to get the accuracy of 70%. And in the same year, Zhu *et al.*[19] presented an algorithm with the help of feature selection and they achieved the accuracy of 80%.

In this paper, we propose an approach for predicting the connectivity pattern of disulfide bonds in a target protein. In this problem, we are given a target protein whose cysteine bonded states (oxidized or reduced) are known. We build hybrid SVM models for prediction. Our feature set include PSSM, secondary structure, and some other features. To test our work, we apply SP39 dataset, which contains 446 proteins with bonded size ranging from 2 to 5. We perform the 4-fold cross-validation, which was also adopted by Chung's work [6]. We achieve accuracies with 70.8% and 65.9% for pair-wise and pattern-wise, and improve 0.7% and 2.4% compared with Chung's work, respectively.

The rest of this paper is organized as follows. In Section II, we will introduce some preliminary knowledge about this study. In Section III, we will present our method to the problem. Section IV shows the experimental results and compares them with some previous results. Finally, Section V gives a conclusion and some possible future works.

II. Preliminary

In this section, we give an introduction to support vector machine, position-specific score matrix and secondary structure, which serves as background knowledge of our prediction method. We also review some methods for disulfide bond prediction in previous studies.

A. Position-Specific Score Matrix

Position-specific scoring matrix (PSSM) is a special scoring matrix in bioinformatics. It is calculated by the sequence

alignment method in order to get the score of each kind of amino acid in each position within a protein. The sequence alignment manages to determine the similarities between the target sequence and the sequences in the queried database. Consequently, it can find the homogeneous sequences with highest similarity, and summarize the information from homogeneity among the queried sequences.

To get PSSM, there are two widely-used tools: BLAST (*basic local alignment search tool*)[1] and PSI-BLAST (*position-specific iterated-BALST*)[2]. BLAST is an algorithm for querying a protein or DNA sequence. With the database specified, it can find the sequences similar to the target sequence by the similarity score function. PSI-BLAST uses the result of BLAST as input to run BLAST iteratively, so it is more sensitive than BLAST to locate distantly homogeneous family.

The PSSM is calculated by PSI-BLAST. The score of each entry in it represents the occurrence frequency of one residue. If one score is higher than other scores in a row, it means that it is more common for this residue in this position in the homogeneous homology family.

B. Secondary Structure

In biochemistry and structural biology, the secondary structure is defined as the interaction of *hydrogen bonds* between residues. Various hydrogen bonding states exhibit distinct structural attributes, and these bonding states may further influence the three-dimensional structure of a protein. Therefore, it is usually believed that the secondary structure provides important information for protein structure prediction.

The secondary structure, as defined in DSSP (*Dictionary of Protein Secondary Structure*)[11], includes several types of hydrogen bonding mode, such as α-helix, β-sheet, 3_{10}-helix, π-helix, etc. Among these types, α-helix and β-sheet are commonly found, and the others are relatively rare in nature proteins. In protein sequences, the secondary structures are roughly categorized into three states: helix, sheet and coil, each of which is found to have its dominant kinds of amino acids. Even though, it is still believed to be intractable to predict secondary structures by means of only the information of an amino acid sequence.

Many algorithms have been proposed for solving the secondary structure prediction, such as neural networks, hidden Markov models, and support vector machines. Among these methods, one of the most accurate is PSIPRED[10], which is a tool based on neural networks. It first invokes PSI-BLAST to locate homogeneous proteins to obtain the evolutionary information, such as replacements, insertions, and deletions of an amino acid. After that, the result of PSI-BLAST is served as input of the neural network to generate prediction results. The reason of its high accuracy is that it uses PSI-BLAST to get homology information. Even though, it still works well without PSI-BLAST.

PSIPRED roughly classifies each residue into three types of secondary structure ($helix$, $strand$ and $coil$), and outputs the results into three kinds of file (.horiz, .ss and .ss2). The .horiz file stores the predicted secondary structure type and the confidence value of the prediction of each amino acid in a protein. Both .ss and .ss2 files contain probabilities for the three kinds of secondary structure associated with each residue. These files contain the predicted secondary structure and probabilities, belonging to coil, helix and strand states, of each residue. The total probability of each residue in .ss file is equal to 1, while that in .ss2 file is not equal to 1. In this paper, we adopt three probability information from the .ss2 file.

C. Support Vector Machine

Support Vector Machine (SVM), is a machine learning method. It is widely used in classification and regression problems. The basic idea is to map the input data into a higher dimensional space, and create a hyperplane to divide different groups of samples. Furthermore, after an SVM model is trained, the class corresponding to a data element can be predicted by this model.

Assume that $\mathbf{x_i}$ is a d-dimensional vector (data), and y_i is the label of $\mathbf{x_i}$, where $y_i \in \{1, -1\}$. The main purpose of SVM is to find an optimal decision hyperplane $\omega^{\mathbf{T}}\mathbf{x} + b = 0$, where $\omega^{\mathbf{T}}, \mathbf{x} \in R^d$ and $b \in R$. The hyperplane is desired to separate as many data elements as possible, by maintaining a maximal margin between the two groups of data. The answers, which are parameters of the hyperplane, are obtained by solving the following optimization problem:

$$\begin{cases} \text{minimize} & \frac{1}{2}\mathbf{w}^T\mathbf{w} \\ \text{subject to} & y_i(\mathbf{w}^T\mathbf{x_i} + b) \geq 1 \ , i = 1 \leq i \leq m. \end{cases} \tag{3}$$

In the ideal situation, the hyperplane can separate all data elements perfectly. However, it rarely happens in real worlds. Consider the case that all kinds of data are mixed and scattered within a 2-D space, and they could not be separated exactly by any linear function. It turns out that we may separate them by curves, not straight lines. To overcome the difficulty, SVM provides nonlinear kernel functions to map the data into a higher dimensional space. Then the problem can be solved with an ordinal linearly separable scheme. In either linear or nonlinear cases, SVM performs very well in most classification problems.

In this paper, LIBSVM[3] is employed to perform the SVC (*Support Vector Classification*) task. In addition, it also provides SVR (*Support Vector Regression*) for regression. The SVC function can classify a set of data elements with their probabilities, while the SVR can generate a regression value for an input data element. These values are helpful to determine which class is more likely for each data element. Furthermore, it provides several kernel functions for various problems. Because of its outstanding performance and extensive functionalities, LIBSVM is widely adopted in the researches on data analysis.

III. Algorithms for Disulfide Bond Prediction

In this section, we present our algorithm for solving the connectivity prediction problem. We explain how to develop

our work based on the previous work. We also introduce the features involved in the prediction.

A. Motivation

In 2009, Chung *et al.*[6] did well in connectivity prediction with the down-sampling technique. To further improve the accuracy, we have to seek some other ways to achieve this goal. And in 2007, Song *et al.*[16] suggested that the secondary structure information also provides good discrimination capability in connectivity prediction. This motivates us to consider the possibility to incorporate these two methods.

Since the feature set used in Chung's work only contains DOC (*distance of cysteines*) and PSSM, we may adopt more features to facilitate the prediction. Furthermore, from the preliminary study, we also find some specific feature sets provide different predictive capability for different types of proteins. This also motivates us to develop our method according to their binding configurations.

Since our goal is to find the pattern combination of the oxidized cysteines, the oxidized-reduced and reduced-reduced pairs are assumed to be irrelevant for the training models. Thus, the oxidized-reduced pairs and reduced-reduced pairs are removed, then all oxidized-oxidized pairs are involved for training. Note that only a part of oxidized-oxidized pairs are included in Chung's work due to the down-sampling scheme. Under this circumstance, the number of our training samples is a little more than that in Chung's work. Thus, our trained model is reasonably more sensitive for identifying truly bonding pairs.

B. Feature Extraction

Our approach is based on Chung's work [6], thus we adopt all features used by Chung. Let L, W, L_{max}, W_{max}, F_c, P_i and AA_x be the length, molecular weight, longest protein length in the dataset, greatest molecular weight in the dataset, number of cysteines in a protein, position of the ith cysteine in a protein and number of the amino acid x in a protein, respectively. The normalized features are defined as follows:

1) $DOC(i, j)$ (*distance of cysteines*): This feature represents the distance of a pair of cysteines in the primary sequence. This value is normalized as $log_2(1.0+\frac{P_j-P_i}{L})$.
2) Cysteine order: This feature shows the order of each cysteine within all cysteines in a protein. This value is normalized as $\frac{i}{F_C}$.
3) Protein weight: The weight is the sum of all residues' molecular weights in the protein. This value is normalized as $\frac{W}{W_{max}}$.
4) Protein length: The feature represents the length of primary sequence of a protein. This value is normalized as $\frac{L}{L_{max}}$.
5) Amino acid composition: This value is to show the occurrence frequency of the amino acid x in a protein. This value is normalized as $\frac{AA_x}{L_{max}}$.

In additional to the features mentioned above, we also involve PSSM and secondary structure features. The normalized PSSM is defined as follows:

$$p_{ij} = \frac{p_{ij} - p_{min}}{p_{max}}, \tag{4}$$

where p_{ij}, p_{max} and p_{min} denote the score of column j in row i, the maximum and minimum values in the matrix, respectively. The secondary structure information is extracted from the .ss2 file, which is generated by PSIPRED v3.2 [10].

We adopt the window approach to retrieve elements around a cysteine as our features. Here, the window size $2k + 1$ is set to 13. Table I shows the names and sizes of all features. In our approach, the training models for the protein with odd number of oxidized pairs and the protein with even number of oxidized pairs are distinct. This is because each feature set has its own strength in the prediction of a unique situation. Their distinct feature sets are also shown in Table I. For example, when the window size is 13, the number of features is 521 for the even model, which contains 1 feature for DOC and $13 \times 20 \times 2$ for PSSM, and 623 for the odd model, which contains 1 for DOC, 1 for length, 1 for weight, 2 for orders, 20 for amino acid composition, $13 \times 20 \times 2$ for PSSM and $13 \times 3 \times 2$ for secondary structure.

C. Algorithm for Connectivity Prediction

Figure 1 exhibits the flow chart of our work. Our approach is also described as follows:

Algorithm: Connectivity prediction with a hybrid model

Input: The primary structure information and the bonded states of cysteines in the target protein, where the real states (oxidized or reduced) of cystienes and numbers of disulfide bonds are assumed to be known in advance.

Output: The connectivity pattern of the target protein.

Step 1: *Bonded type determination*. Determine whether the protein is odd or even disulfide-boned. This step determines which model is selected for prediction.

Step 2: *Feature encoding*. Encode each possible cysteine pair of the target protein into the feature vector according to the selected SVM model.

Step 3: *SVM prediction*. For each pair of oxidized cysteines, predict whether they are connected or not with the selected SVM model. After this step, the connecting probability of each pair is obtained..

Step 4: *Pattern construction*. Build an undirected weighted graph, where each node represents one oxidized cysteine and the connecting probability of two cysteines is the weight of the corresponding edge. Apply the algorithm for the weighted matching to derive the connectivity pattern [7].

As we get a target protein (input), we first feed it to PREPRED and PSI-BLAST to obtain the predicted secondary structure information and PSSM, respectively. Then we invoke our algorithm to perform the connectivity prediction (output).

TABLE I
THE FEATURE SETS USED IN OUR APPROACH.

Feature	size	Model for odd	Model for even
Distance of cysteine	1	Y	Y
Cysteine order	2	Y	N
Protein weight	1	Y	N
Protein length	1	Y	N
Amino acid composition	20	Y	N
PSSM around cystein	$(2k+1) \times 20 \times 2$	Y	Y
Secondary structure around cystein	$(2k+1) \times 3 \times 2$	Y	N

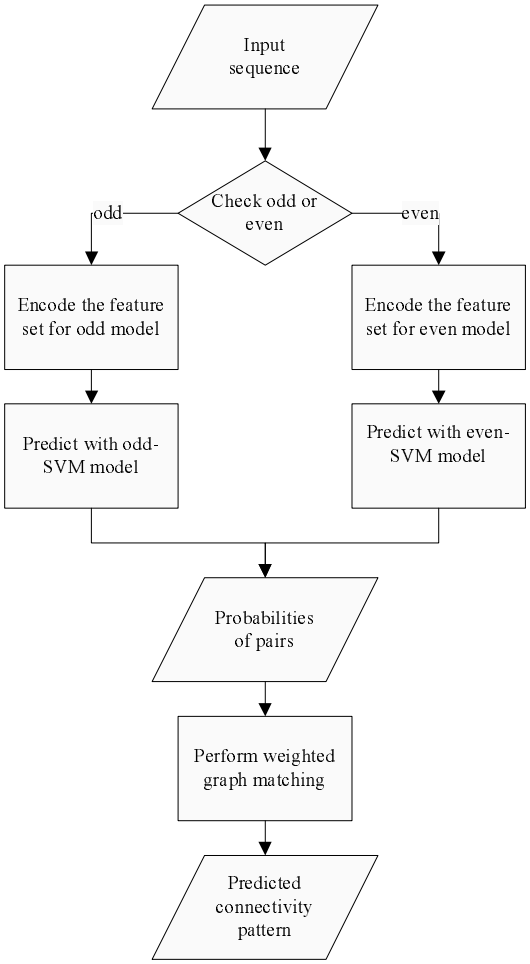

Fig. 1. The flowchart of our work for connectivity prediction.

IV. CONNECTIVITY PREDICTION EXPERIMENTS

In this chapter, we introduce the dataset used in our experiments and then explain how we compute the accuracy. Finally, we carry out the performance comparison of our work with others.

A. Dataset

For comparing with other works in connectivity prediction, we adopt the same dataset that was used by the previous works. The dataset is given as follows:

SP39: The dataset was derived from SWISS-PROT release No. 39. This dataset consists of 446 proteins and the number of disulfide bonds in each protein ranges from 2 to 5. To compare with Chung's work [6], we apply the same way to divide the dataset into 4 subsets for 4-fold cross-validation, in which the sequence identity between any two subsets is less than 30%. Moreover, to verify the correctness and remedy randomness, we also divide this dataset into 4 folds randomly and perform experiments. Totally, 10 different 4-fold cross validations are carried out.

B. Performance Evaluation

We use k-fold cross-validation to evaluate our result. The dataset D is split into k disjoint subsets D_1, D_2, ..., D_k. We take D_i, $1 \leq i \leq k$, for testing and the other k-1 subsets for training. This process is repeated k times until all subsets are tested.

We group the predicted results into 4 categories: TP (*true positive*), TN (*true negative*), FP (*false positive*), and FN (*false negative*). In pair-wise prediction, we adopt Q_c as the performance measure calculated as follows:

$$Q_c = \frac{P_c}{N_c} = \frac{TP}{TP + FN}, \tag{5}$$

where P_c denotes the number of bonded pairs that are correctly predicted and N_c denotes the total bonded pairs in the dataset.

For pattern prediction, we use Q_p to evaluate the performance. The formula is given as follows:

$$Q_p = \frac{P_p}{N_p}, \tag{6}$$

where P_p denotes the number of correctly predicted proteins and N_p denotes the total number of proteins in the dataset.

Finally, we perform k-fold cross-validation and calculate the average accuracy of all folds, which is presented as follows:

$$R = \frac{1}{k} \sum_{i=1}^{k} R_i. \tag{7}$$

where R_i could be Q_p or Q_c.

C. Experiments of Connectivity Prediction

According to the literature, the down-sampling technique is usually used to balance the negative samples and positive samples in the training set so that it prevents the classification

algorithm from favoring the data associated with the majority class. In Chung's work [6], it turns out that the best performance is achieved when the ratio between the numbers of negative and positive samples is 3. The parameter cost C and gamma γ for SVM training are 2 and 0.125, respectively.

Our initial thought is to develop our method based on Chung's work. Since only two feature sets were used in Chung's work, it is expected that we can add more useful features to achieve better performance. Accordingly, we include the features that was used in Song's work, such as distance of cysteines (D), PSSM (P), cysteine ordering (O), protein length (L), protein weight (W), amino acid composition (A) and secondary structure (S). The number of features is 521, including D and P, in Chung's work and 623, including D, L, W, O, A, P and S, in Song's work. Furthermore, we also refer to Zhu's work for feature selection. They suggested that the best result can be achieved by selecting only 150 features from those in Song's work.

We adopt Zhu's method to select 150 features and perform the experiment. Table II show the results obtained with Chung's data partition method. And Table III shows the results of randomly divided folds. Note that both tables employ the training dataset that is obtained by the down-sampling technique. The purpose of the experiment using randomly divided folds shown in Table III is to verify that our work is correct, since some other previous works also use randomly divided folds. The results of our experiment indicate that our performance is as good as the previous works, except for the result of 150 features in Zhu's selection. We conclude that down-sampling is not suitable for this reduced feature subset.

As Table II shows, the result obtained with the D+P feature set (in the first row) outperforms the other two feature sets. The effects of adding new features do not gain any improvement. Instead of adopting the down-sampling scheme, we build another training dataset, which contains all oxidized-oxidized pairs and discards other pairs (oxidized-reduced or reduced-reduced). This is because our aim is to predict disulfide bonds, one pair formed by at least one reduced cysteine is irrelevant for the training process. In our new training dataset, the ratios of negative and positive samples for different bonded sizes are different. For example, when the bonded sized is 3 (6 cysteines), the number of all possible negative bonded pairs is $C_2^6 - 3 = 12$. So, the negative-positive ratio is 4. We conduct another training process, in which the feature sets are the same as Table II but the positive/negative ratios are adjusted. The result corresponding to low identity partitions is shown in Table IV, and the result corresponding to randomly divided folds is shown in Table V.

The result in Table IV shows that results obtained by 4-fold cross-validation. Because the selected sequences in distinct folds are low in identity, it is expected that they are independent mutually. It shows that the accuracies of models involving either 521 features or 623 features are almost the same. Although getting elevated in accuracy (compared with Table II), the model with 150 selected features is still worse than others, either with the down-sampling scheme or with all oxidized-oxidized-pair samples.

The results in Table V are obtained from randomly partitioned 4-fold cross-validation experiments. It shows that the overall result associated with 150 selected features is still slightly worse than other two feature sets. It is not in accordance with what Zhu suggested that the classification rate corresponding to 150 features is higher than that associated with 623 features (the second feature set in the same table). Because the result derived from the randomly divided folds is inconsistent with what literatures declare, we thus do not consider this scheme furthermore. Consequently, we only focus on the folds divided with sequence identity less than 30%. In addition, because models trained with 150 features do not perform well in this independent test, we do not list it in the final comparisons.

By observing the results in Table IV, we find something interested. The accuracies of even bonded with 521 features are higher than those with 623 features, while the accuracies of odd bonded with 623 features are better than those with 521 features. It suggests that the secondary structure information is more useful in identifying disulfide bonds with odd-number pairs than those with even-number pairs. Therefore, we propose a new method which incorporates these two models. That is, we use the model with 521 features to predict proteins containing even disulfide bonds and use the other model with 623 features to predict proteins containing odd disulfide bonds. Table VI shows the result comparison between our hybrid model and Chung's work. The new model always adopts the most prominent model for prediction. Consequently, in most circumstances, the prediction accuracy is higher.

V. Conclusion

Chung [6] proposed the down-sampling scheme for reducing training samples (including oxidized-oxidized, oxidized-reduced and reduced-reduced) to get higher accuracy and to reduce execution time. However, according to our experience, oxidized-reduced and reduced-reduced pair samples seem to be irrelevant for connectivity pattern prediction. The connectivity patterns is in fact composed of only oxidized cysteines, so it seems reasonable not to consider the permutations and combinations of reduced cysteines.

It should be noted that when training with all samples, one way to speed up the process is to reduce features. Zhu's *et. al.* [19] select 150 features from original 623 features so that they make the process faster and achieve better performance. In our study, we find the selected feature set is not so helpful for the same dataset which is divided into four nearly independent folds. That is, the reduced feature set might not be useful in independent test. To overcome the situation, it would be better to choose another feature subset.

Song's *et. al.*[16] suggested that the predicted secondary information plays an important role in connectivity prediction. Hence, it is believed that the more accurate the secondary structure, the more helpful for disulfide bond prediction. Since the secondary information is obtained by prediction tools, it

TABLE II
PREDICTION ACCURACIES WITH THE DOWN-SAMPLING SCHEME.

Feature set	$B=2$		$B=3$		$B=4$		$B=5$		$B=2\cdots5$	
	Q_p	Q_c	Q_p	Q_c	Q_p	Q_c	Q_p	Q_c	Q_p	Q_c
$D+P$[a]	80.8	80.8	57.5	66.2	54.5	70.7	42.2	61.8	63.5	70.1
$D+P+L+W+O+A+S$	76.9	76.9	54.1	63.5	57.6	67.7	37.8	57.8	61.2	66.8
$150\ features\ in\ Zhu's\ work$	77.6	77.6	50.7	60.7	38.4	53.8	33.3	51.1	55.6	61.0

[a] From Chung *et al.* [6].

TABLE III
PREDICTION ACCURACIES CORRESPONDING TO RANDOMLY DIVIDED FOLDS WITH THE DOWN-SAMPLING SCHEME.

Feature set	$B=2$		$B=3$		$B=4$		$B=5$		$B=2\cdots5$	
	Q_p	Q_c	Q_p	Q_c	Q_p	Q_c	Q_p	Q_c	Q_p	Q_c
$D+P$[a]	87.7	87.7	68.8	75.3	79.9	85.5	50.9	63.6	76.0	79.2
$D+P+L+W+O+A+S$	87.6	87.6	70.3	76.9	77.1	82.0	51.1	63.1	75.9	78.6
$150\ features\ in\ Zhu's\ work$	84.4	84.4	67.1	74.2	73.9	80.3	46.0	59.7	72.5	75.9

TABLE IV
PREDICTION ACCURACIES OBTAINED BY THE TRAINING DATASET CONTAINING ALL OXIDIZED-OXIDIZED PAIRS.

Feature set	$B=2$		$B=3$		$B=4$		$B=5$		$B=2\cdots5$	
	Q_p	Q_c	Q_p	Q_c	Q_p	Q_c	Q_p	Q_c	Q_p	Q_c
$D+P$	84.0	84.0	53.4	63.7	55.6	66.9	46.7	60.4	63.9	68.7
$D+P+L+W+O+A+S$	78.8	78.8	60.2	69.2	53.5	64.4	44.4	62.2	63.7	69.1
$150\ features\ in\ Zhu's\ selection$	78.8	78.8	51.4	62.1	48.5	68.2	33.3	52.9	58.5	64.6

TABLE V
PREDICTION ACCURACIES CORRESPONDING TO RANDOMLY DIVIDED FOLDS OBTAINED BY THE TRAINING DATASET CONTAINING ALL OXIDIZED-OXIDIZED PAIRS.

Feature set	$B=2$		$B=3$		$B=4$		$B=5$		$B=2\cdots5$	
	Q_p	Q_c	Q_p	Q_c	Q_p	Q_c	Q_p	Q_c	Q_p	Q_c
$D+P$	88.5	88.5	72.5	78.8	77.0	82.4	52.0	62.0	77.0	79.8
$D+P+L+W+O+A+S$	88.9	88.9	70.5	77.5	77.6	82.9	52.2	66.1	76.7	79.8
$150\ features\ in\ Zhu's\ selection$	86.5	86.5	71.8	76.8	79.2	83.6	49.1	61.4	76.2	78.4

might not guarantee to be fully correct. Consequently, the provided information might be somewhat limited.

We finally propose a hybrid model in our work. According to different bonding configurations, we construct our model with different feature sets and parameters for SVM. We roughly divide the proteins into odd or even bonded. For an odd-bonded protein, the predicted secondary structure information is included to build the SVM model. But, the information is not included for an even-bonded protein. By this approach, we get better result in the SP39 dataset. In this sense, sequences under various bonding configurations may somewhat different in physical or chemical properties. Consequently, proteins associated with different configurations may have their individually preferential model for prediction.

REFERENCES

[1] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipmanl, "Basic local alignment search tool," *Journal of Molecular Biology*, Vol. 215, No. 3, pp. 403–410, 1990.

[2] S. F. Altschul, T. L. Madden, A. A. Schaffer, J. Zhang, Z. Zhang, W. Miller, and D. J. Lipman, "Gapped blast and psi-blast: a new generation of protein database search programs," *Nucleic Acids Research*, Vol. 25, No. 17, pp. 3389–3402, 1997.

[3] C. C. Chang and C. J. Lin, *LIBSVM: A library for support vector machines*. National Taiwan University, No. 1, Roosevelt Rd. Sec. 4, Taipei, Taiwan 106, ROC, 2001. Software available at http://www.csie.ntu.edu.tw/∼cjlin/libsvm.

[4] Y.-C. Chen and J.-K. Hwang, "Prediction of disulfide connectivity from protein sequences," Vol. 61, pp. 507–512, 2005.

[5] Y.-C. Chen, Y.-S. Lin, C.-J. Lin, and J.-K. Hwang, "Prediction of the bonding states of cysteines using the support vector machines based on multiple feature vectors and cysteine state sequences," Vol. 55, pp. 1036–1042, 2004.

[6] W.-C. Chung, "A multi-phase approach for disulfide bond prediction," Master Thesis, Department of Computer Science and Engineering, National Sun Yat-Sen University, Kaohsiung, Taiwan, 2009.

[7] P. Fariselli and R. Casadio, "Prediction of disulfide connectivity in proteins," Vol. 17, No. 10, pp. 957–964, 2001.

[8] P. Frasconi, A. Passerini, and A. Vullo, "A two-stage svm

TABLE VI
PREDICTION ACCURACIES COMPARED WITH CHUNG'S WORK.

Methods	$B=2$		$B=3$		$B=4$		$B=5$		$B=2\cdots5$	
	Q_p	Q_c	Q_p	Q_c	Q_p	Q_c	Q_p	Q_c	Q_p	Q_c
$Down-sampling$[a]	80.8	80.8	57.5	66.2	54.5	70.7	42.2	61.8	63.5	70.1
$Our\ method$	84.0	84.0	60.2	69.2	55.6	66.9	44.4	62.2	65.9	70.8

[a] From Chung *et al.* [6].

architecture for predicting the disulfide bonding state of cysteines," *Neural Networks for Signal Processing, 2002. Proceedings of the 2002 12th IEEE Workshop on*, pp. 25–34, 2002.

[9] Jayavardhana Rama G. L., A. P. Shilton, M. M. Parker, and M. Palaniswami, "Prediction of cystine connectivity using SVM," *Bioinformation*, Vol. 1, No. 2, pp. 69–74, 2005.

[10] D. T. Jones, "Protein secondary structure prediction based on position-specific scoring matrices," *Journal of Molecular Biology*, Vol. 292, No. 2, pp. 195–202, 1999.

[11] W. Kabsch and C. Sander, "Dictionary of protein secondary structure: Pattern recognition of hydrogen-bonded and geometrical features," *Biopolymers*, Vol. 22, pp. 2577–2637, 1983.

[12] C.-Y. Lin, C.-B. Yang, C.-Y. Hor, and K.-S. Huang, "Disulfide bonding state prediction with svm based on protein types," *Bio-Inspired Computing: Theories and Applications*, pp. 1436–1442, 2010.

[13] H.-L. Liu and S.-C. Chen, "Prediction of disulfide connectivity in proteins with support vector machine," Vol. 38, No. 1, pp. 63–70, 2007.

[14] C.-H. Lu, Y.-C. Chen, C.-S. Yu, and J.-K. Hwang, "Predicting disulfide connectivity patterns," Vol. 67, pp. 262–270, 2007.

[15] R. Singh, "A review of algorithmic techniques for disulfide-bond determination," *Brief Funct Genomic Proteomic*, Vol. 7, No. 2, pp. 157–172, 2008.

[16] J. Song, Z. Yuan, H. Tan, T. Huber, and K. Burrage, "Predicting disulfide connectivity from protein sequence using multiple sequence feature vectors and secondary structure," *Bioinformatics*, Vol. 23, No. 23, pp. 3147–3154, 2007.

[17] C.-H. Tsai, B.-J. Chen, C.-H. Chan, H.-L. Liu, and C.-Y. Kao, "Improving disulfide connectivity prediction with sequential distance between oxidized cysteines," Vol. 21, No. 24, pp. 4416–4419, 2005.

[18] M. Vincent, A. Passerini, M. Labbe, and P. Frasconi, "A simplified approach to disulfide connectivity prediction from protein sequences," *BMC Bioinformatics*, Vol. 9, No. 1, p. 20, 2008.

[19] L. Zhu, J. Yang, J.-N. Song, K.-C. Chou, and H.-B. Shen, "Cysteine separations profiles (csp) on protein sequences infer disulfide connectivity," *Journal of Computational Chemistry*, Vol. 31, No. 7, pp. 1415–1420, 2009.