(This is a sample cover image for this issue. The actual cover is not yet available at this time.)

Journal of Theoretical Biology 315 (2012) 64–70

Contents lists available at SciVerse ScienceDirect

Journal of Theoretical Biology

journal homepage: www.elsevier.com/locate/yjtbi

Interrogating noise in protein sequences from the perspective of protein–protein interactions prediction

Yongcui Wang [a,1], Xianwen Ren [b,1], Chunhua Zhang [c], Naiyang Deng [d,*], Xiangsun Zhang [e,*]

[a] Key Laboratory of Adaptation and Evolution of Plateau Biota, Northwest Institute of Plateau Biology, Chinese Academy of Science, Xining 810001, China
[b] MOH Key Laboratory of Systems Biology of Pathogens, Institute of Pathogen Biology, Chinese Academy of Medical Sciences and Peking Union Medical College, Beijing 100730, China
[c] Information School, Renmin University of China, Beijing 100872, China
[d] College of Science, China Agricultural University, Beijing 100083, China
[e] Academy of Mathematics and Systems Science, CAS, Beijing 100190, China

HIGHLIGHTS

- We evaluate the denoising techniques in protein–protein interactions (PPIs) prediction.
- Two kinds of denoising formulas efficient in phylogenetic trees construction are introduced.
- We integrate noise in protein sequences with a support vector machine.
- Three kinds of organisms PPIs datasets are used for validation.
- The denoising formulation cannot improve the PPIs prediction.

ARTICLE INFO

Article history:
Received 28 November 2011
Received in revised form
20 August 2012
Accepted 9 September 2012
Available online 18 September 2012

Keywords:
Bioinformatics
Denoising
Composition vector
Machine learning

ABSTRACT

The past decades witnessed extensive efforts to study the relationship among proteins. Particularly, sequence-based protein–protein interactions (PPIs) prediction is fundamentally important in speeding up the process of mapping interactomes of organisms. High-throughput experimental methodologies make many model organism's PPIs known, which allows us to apply machine learning methods to learn understandable rules from the available PPIs. Under the machine learning framework, the composition vectors are usually applied to encode proteins as real-value vectors. However, the composition vector value might be highly correlated to the distribution of amino acids, i.e., amino acids which are frequently observed in nature tend to have a large value of composition vectors. Thus formulation to estimate the noise induced by the background distribution of amino acids may be needed during representations. Here, we introduce two kinds of denoising composition vectors, which were successfully used in construction of phylogenetic trees, to eliminate the noise. When validating these two denoising composition vectors on *Escherichia coli* (*E. coli*), *Saccharomyces cerevisiae* (*S. cerevisiae*) and human PPIs datasets, surprisingly, the predictive performance is not improved, and even worse than non-denoised prediction. These results suggest that the noise in phylogenetic tree construction may be valuable information in PPIs prediction.

1. Introduction

Identification of the interactions among proteins is crucial to illustrate their functions, and further, can help scientists to understand the underlying mechanisms of many biological phenomena, such as cell cycles, apoptosis, signal transduction and pathogenesis of diseases. It has become one of the most challenging and important tasks in the post-proteomic researches. Various experimental techniques have been developed for large-scale protein-protein interactions (PPIs) analysis, including yeast two-hybrid systems (Fields and Song, 1989; Ito et al., 2001), mass spectrometry (Gavin et al., 2002; Ho et al., 2002), protein chip (Zhu et al., 2001) and so on. One computational idea is applying the machine learning methods to learn understandable rules from the available PPIs and furthermore to predict novel interactions

* Corresponding authors.
E-mail addresses: ycwang@nwipb.cas.cn (Y. Wang), renxwise@gmail.com (X. Ren), zhangchunhua@ruc.edu.cn (C. Zhang), dengnaiyang@cau.edu.cn, wangyc_82@163.com (N. Deng), zxs@amt.ac.cn (X. Zhang).
[1] These two authors contribute equally to the whole work.

0022-5193/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.jtbi.2012.09.007

(Deng et al., 2011; Hu et al., 2011; Ma et al., 2011; Qiu and Wang, 2012; Chou and Cai, 2006; Ren et al., 2011; Xia et al., 2010; Yang and Jiang, 2010; Zhang et al., 2011; Zhou, 2011). Comparing with costly and time-consuming biochemical experiments, computational methods for PPIs prediction have played an important role (Shen et al., 2007).

One key issue in machine learning is to extract protein attributes that are highly relevant to prediction of PPIs. Among the various attributes of proteins, the primal sequences are most popular because they are the most basic and the easiest to obtain due to the rapid development of genomic sequencing technologies. In addition, the primary sequences of proteins actually specify their structures that provide the molecular basis for PPIs. Therefore, protein primary sequences hold the promise to contain virtually sufficient information to construct the most universal predicting method (Shen et al., 2007).

How to encode the given protein sequences as the real-value vectors is the key to construct a universal sequence-based PPIs predictor. Many studies attempt to apply composition vectors to tackle this problem for various kinds of applications (Shen et al., 2007; Ben-Hur, 2005; Gomez et al., 2003; Bock and Gough, 2001; Najafabadi and Salavati, 2008; Leslie et al., 2002). Composition vectors have been used widely in predictions, for example, protein localization predictions. The subcellular localization of a protein allows further understanding its structure and molecular function (Arango-Argoty et al., 2011). Many types of composition vector have been successfully developed for protein subcellular localization prediction, such as PseAA composition (Chou, 2001; Chou and Cai, 2004; Chou and Shen, 2006), signal peptide (Hoglund et al., 2006), sequence domain (Chou and Cai, 2002), PSSM (Mak et al., 2008; Pierleoni et al., 2006), k-mer (Mei and Wang, 2010; Dijk et al., 2008), etc. However, the composition vector value might be highly correlated to the distribution of amino acids, i.e., amino acids which are frequently observed in nature tend to have a large value of composition vectors. Thus formulation to estimate the noise seems to be needed during representations. There are some works have discussed this problem, for example, Chang et al. have proposed a probability-based mechanism for transforming protein sequences into feature vectors to eliminate the noise of composition vector (Yu et al., 2010). However, when constructing one protein denoising composition vector, more than 10 thousand times permutation are needed, resulting in very large computational consumption. Chan et al. have proposed a low computational cost denoising mechanism, which is based on the principle of maximum entropy, to encode the proteins as real-value vectors (Chan et al., 2010). By using the angle-based distance measures on the denoising vectors, they have constructed well-grouped phylogenetic trees.

Following the previous works, in this paper, we introduce two types of low costly denoising formulas, which have clear probability assumptions and were successfully used in phylogenetic tree construction. We hypothesize that these two techniques may reveal some true sequence noise in proteins and may be useful to improve PPIs prediction. To test whether the PPIs prediction performance can be improved by these two denoising formulas or not, we introduce support vector machine (SVM) as the PPIs predictor. SVMs, which are motivated by statistical learning theory (Vapnik, 1995, 1998; Deng et al., 2012), have been proven successful on many different classification problems in bioinformatics (Noble, 2004). Identification of PPIs can be addressed as the two-classification problem: determining whether a given pair of proteins is interacting or not. Thus two-class SVM with the composition vectors and denoising composition vectors are used to predict *Escherichia coli* (*E. coli*), *Saccharomyces cerevisiae* (*S. cerevisiae*) and human PPIs, respectively. Surprisingly, for all three kinds of organisms on randomly and artificially negative datasets, the predictive performance of denoising composition vectors are not better than the primal composition vectors. These results suggest that, the denoising formulation efficient in phylogenetic trees construction cannot improve the PPIs prediction, i.e., what is noise is dependent on the applications.

According to a recent comprehensive review (Chou, 2011), to establish a really useful statistical predictor for a protein system, we need to consider the following procedures: (i) construct or select a valid benchmark dataset to train and test the predictor; (ii) formulate the protein samples with an effective mathematical expression that can truly reflect their intrinsic correlation with the target to be predicted; (iii) introduce or develop a powerful algorithm (or engine) to operate the prediction; (iv) properly perform cross-validation tests to objectively evaluate the anticipated accuracy of the predictor; (v) establish a user-friendly web-server for the predictor that is accessible to the public. We described these steps as follows.

We begin by introducing a popular composition vector in protein representation. Then we give the two kinds of denoising formulas which is efficient in phylogenetic trees construction. After that, by performing cross-validation tests, we compare two kinds of denoising formulas with primal composition vector regarding to their predictability, and show the predictive accuracy deterioration arising from denoising technology. Lastly, the discussions and conclusions are presented.

2. Materials and methods

Sequence-based attributes become popular in PPIs prediction not only because that the primal sequences are most basic and the easiest to obtain, but also owing to the assumption that knowledge of the amino acid sequence alone might be sufficient to estimate the evolutionary history, overall structure and function, and the interacting propensity between two proteins. Especially, Shen et al. have proposed a simple but effective feature encoding method, called conjoint triad feature (CTF) to represent the protein sequences (Shen et al., 2007). Shen et al. have shown that SVM with the CTF outperforms other sequence-based methods in human PPIs prediction. In addition, the CTF can be implemented in an economic way and contains no pre-defined parameters. Inspired by these observations, we first introduce the CTF and then apply the denoising approaches to formulate the denoising CTF vectors. In the Results section, we test the performance of the denoising CTF vectors on *E. coli*, *S. cerevisiae* and human randomly and artificially negative datasets.

2.1. Input feature vectors

We give the description on the CTF now. First, based on the dipoles and volumes of the side chains, the 20 amino acids are classified into seven classes: $\{A,G,V\}$, $\{I,L,F,P\}$, $\{Y,M,T,S\}$, $\{H,N,Q,W\}$, $\{R,K\}$, $\{D,E\}$, $\{C\}$. Second, a binary space (V,F) is applied to represent a given protein. The element of V, v_i represents a sort of triad type, and the element of F, f_i represents the frequency of type v_i appearing in the given protein sequence. Thus a 343 $(7\times7\times7)$-dimension vector is used to represent given protein, where each element of this vector is the frequency of the corresponding conjoint triad appearing in the protein sequence. The detailed definition and description for (V,F) are illustrated in SI in Fig. 3 in Shen et al. (2007).

2.1.1. Denoising vectors

The CTF considers the frequency of each conjoint triad type. However, the value of CTF's element might be highly correlated to the distribution of amino acids, i.e., triads that consist of amino

acid groups frequently observed in nature (e.g., groups 1 and 2) tend to have a large value of frequency. To deal with this problem, we introduce the denoising formula based on maximum entropy method to remove noises.

Given a conjoint triad type $\alpha_1\alpha_2\alpha_3$, the following two formulas proposed by Hao et al. (2003) and Yu et al. (2005) are applied to estimate the noise of $\alpha_1\alpha_2\alpha_3$:

Hao's formula:

$$q^{Hao}(\alpha_1\alpha_2\alpha_3)=\begin{cases}(f(\alpha_1\alpha_2)f(\alpha_2\alpha_3))/f(\alpha_2) & \text{if } f(\alpha_2)\neq 0,\\ 0 & \text{otherwise,}\end{cases} \quad (1)$$

where $f(u)$ represents the frequency of any string u appearing in the sequence. The formula (1) reveals the functional and evolutionary relatedness of word sequence, and was successfully applied for the phylogenetic analysis of prokaryotes, based on whole genome sequences (Hao et al., 2003).

Yu's formula:

$$q^{Yu}(\alpha_1\alpha_2\alpha_3)=(f(\alpha_1)f(\alpha_2\alpha_3)+f(\alpha_1\alpha_2)f(\alpha_3))/2, \quad (2)$$

where $f(u)$ represents the frequency of any string u appearing in the sequence. The formula (2) was commonly introduced in the area of complex and dynamic systems, and was successfully applied for the phylogenetic analysis of prokaryotes, chloroplasts and other phylogenetic problems, based on whole genome sequences (Yu et al., 2005). Both Hao's and Yu's methods provide an effective means to correctly infer phylogenetic trees from composition vectors of sequences. Their assumptions behind these two formulas are the random background of biological sequences.

Then the input vector feeding to the SVM can be formulated as the signal-to-noise ratio:

$$s(\alpha_1\alpha_2\alpha_3)=(f(\alpha_1\alpha_2\alpha_3)-q(\alpha_1\alpha_2\alpha_3))/q(\alpha_1\alpha_2\alpha_3). \quad (3)$$

Comparing with the CTF, the element of the denoising vector becomes the signal-to-noise ratio s.

2.1.2. Protein pairs vectors

PPIs prediction treats each protein pair as the input, the vectors representing the protein pairs should be proposed. The concatenation operator is commonly used in protein pairs representation. However, the asymmetry problem will arise due to the fact that the prediction result will be different on protein pair A–B and B–A. To solve this problem, we concatenate the arithmetical and the geometric average of protein vectors to represent the protein pairs, i.e.

$$F_{AB}=((F_A+F_B)/2)\oplus\sqrt{F_A*F_B}, \quad (4)$$

where F_A,F_B represent the feature vector of protein A and B, the operator $*$ means the multiplication of the corresponding elements, and $\oplus$ represents the concatenation operator. The above representation method for protein pairs cannot only maintain symmetry (A–B identical to B–A), but also make the feature vectors representing proteins constructed uniquely from the protein pair representing vector (Ren et al., 2011).

2.2. Negative training datasets

With the above feature vector construction scheme, the PPIs prediction task is ready to be formalized as a classification problem with the publicly available PPIs the positive samples, and the others as the negative samples. The training data imbalance problem will arise, because there are only a relatively small number of known PPIs. This situation will make the SVM ineffective in determining the class boundary (Wu and Chang, 2003). To maintain a balance between training positive and negative datasets in SVM training procedure, that is, to make the number of negative dataset the same as the positive dataset, we introduce two types of negative datasets to train the SVM-based predictor. The first one is the randomly negative dataset. The randomly negative samples are sampled randomly from the complementary graph of the known PPIs network. Comparing with the method for generating the negative training dataset with the help of the functional annotation of proteins, this randomly generating scheme for negative training data can lead to unbiased estimates of prediction accuracy (Ben-Hur and Noble, 2006). The second one is the artificially negative dataset. The artificially negative samples are constructed by uShuffle based on the positive datasets (Jiang et al., 2008). uShuffle creates the negative proteins by generating uniform random permutations of positive sequences while preserving the exact k-let counts. Here, we let k be 1 and 2, that is preserving amino acid composition and binary composition, respectively.

2.3. Benchmark datasets and SVM implementation

Here, PPIs on three different organisms: *E. coli*, *S. cerevisiae* and human are used to validate the performance of the proposed predictive models. *E. coli* and *S. cerevisiae* PPIs datasets are first introduced in Table 1 in Najafabadi and Salavati (2008). Human PPIs dataset is proposed by Yungki Park (2009). The protein sequences are download from the RefSeq database of NCBI. In addition, the interactions which contain missing proteins in the corresponding proteome sequence datasets are excluded. Thus the number of interactions is 6954, 6635 and 38324 for *E. coil*, *S. cerevisiae* and human, respectively.

In statistical prediction, the following three cross-validation methods are often used to examine a predictor for its effectiveness in practical application: independent dataset test, sub-sampling (5-fold, 7-fold, or 10-fold cross-validation) test, and jackknife test (Chou and Zhang, 1995). However, as elucidated in Chou and Shen (2008) and demonstrated by Eqs. (28)–(32) of Chou (2011), among the three cross-validation methods, the jackknife test is deemed the least arbitrary that can always yield a unique result for a given benchmark dataset, and hence has been increasingly used and widely recognized by investigators to examine the accuracy of various predictors (Mohabatkar, 2010; Chou and Shen, 2010; Esmaeili et al., 2010; Georgiou et al., 2009; Mohabatkar et al., 2011; Chou et al., 2011; Xiao et al., 2011; Chou and Shen, 2010; Wang et al., 2011; Wu et al., 2011; Chou et al., 2012). However, in this study we used the independent dataset to test our model in order to reduce the computational time as done by some investigators using the SVM as the prediction engine.

We train the two-class SVM with denoising CTF and CTF by using *LibSVM* (Chang and Lin, 2011). In the implementation of two-class SVM, the RBF kernel function is used. The penalty parameter C and the RBF kernel parameter γ are optimized by grid search approach with 3-fold cross-validation. To evaluate the performance of our methods, we use the 10-fold cross-validation, that is, the gold-standard dataset is split into 10 subsets with roughly equal size. Each subset is then taken in turn as a test dataset, and train on the remaining nine datasets. The performances of our proposed methods are evaluated by the following evaluation criterions: AUC (area under the receiver operating curve (ROC) curve (Gribskov and Robinson, 1996), Accuracy (Acc) $=(TP+TN)/(TP+TN+FP+FN)$, Sensitivity (Sn) $=TP/(TP+FN)$, Specificity (Sp) $=TN/(TN+FP)$, Precision (Pre) $=TP/(TP+FP)$, and F-measure $=(2\times Sn\times Sp)/(Sn+Sp)$. Here TP is the number of protein pairs correctly predicted to interact, FP is the number of protein pairs predicted to interact but actually not. And TN is the number of protein pairs that do not interact and predicted correctly, FN is the number of protein pairs predicted not to interact but actually interact.

3. Results

3.1. Overview of performances for denoising methods

We overall compare two kinds of denoising formulas with primal formula on all three kinds of organisms PPIs datasets under AUC criterion in Figs. 1–3. From these three figures, we can see that, on both randomly and artificially negative datasets, for all three kinds of organisms PPIs datasets, *Denoising*^{Yu} CTF obtains much higher AUC than *Denoising*^{Hao} CTF. However, the AUC obtained by both *Denoising*^{Yu} CTF and *Denoising*^{Hao} CTF are worse than obtained by the CTF. Although on *S. cerevisiae* artificially negative datasets, *Denoising*^{Yu} CTF can obtain the comparable AUCs with the CTF, the prediction performance are still not improved by this denoising formula. It means that the above two kinds of denoising formulas, which is useful to construct phylogenetic tree, cannot improve the performance of PPIs prediction.

3.2. The performance on the randomly negative datasets

We test the effect of two kinds of denoising CTF formulas (*Denoising*^{Hao} CTF and *Denoising*^{Yu} CTF) on *E. coli*, *S. cerevisiae* and human randomly negative datasets, respectively. The evaluation

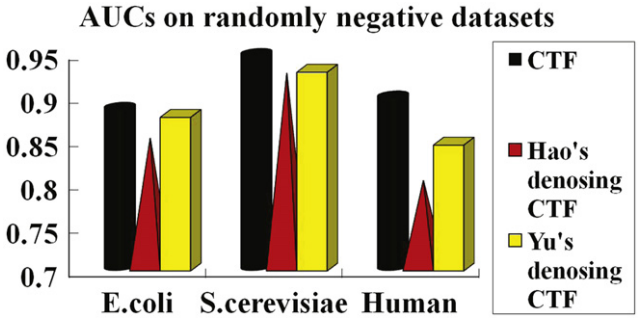

Fig. 1. The AUCs for various methods on randomly negative datasets.

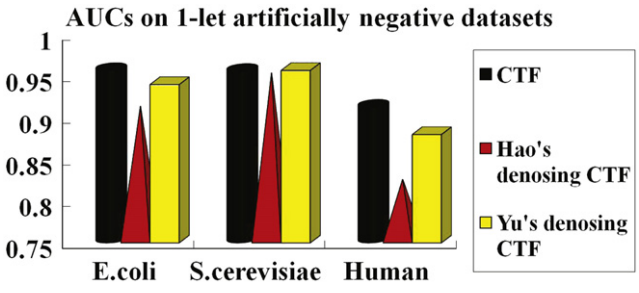

Fig. 2. The AUCs for various methods on 1-let artificially negative datasets.

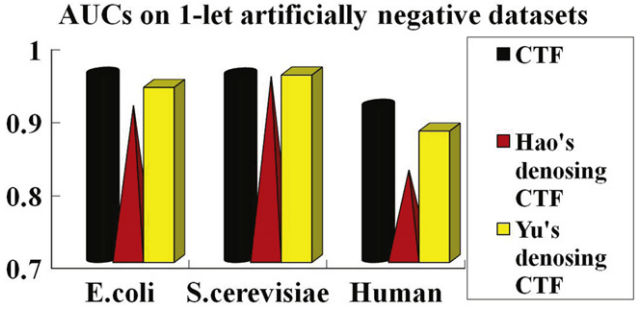

Fig. 3. The AUCs for various methods on 2-let artificially negative datasets.

Table 1
The performance comparison of denoising CTF with CTF on randomly generated negative set. The best predictions obtained are highlighted in bold.

Organism	Encoding methods	Evaluation criterions					
		AUC	Acc	Sn	Sp	Pre	F-measure
E. coli	CTF	**0.886**	**0.797**	**0.794**	**0.799**	**0.798**	**0.797**
	Denoising^{Hao} CTF	0.849	0.763	0.726	0.799	0.784	0.761
	Denoising^{Yu} CTF	0.877	0.791	0.782	0.799	0.796	0.791
S. cerevisiae	CTF	**0.948**	**0.880**	**0.879**	**0.927**	**0.909**	**0.882**
	Denoising^{Hao} CTF	0.924	0.853	0.806	0.899	0.889	0.851
	Denoising^{Yu} CTF	0.929	0.862	0.824	0.899	0.891	0.861
Human	CTF	**0.899**	**0.824**	**0.848**	**0.799**	**0.809**	**0.823**
	Denoising^{Hao} CTF	0.800	0.728	0.757	0.699	0.716	0.727
	Denoising^{Yu} CTF	0.845	0.765	0.730	0.799	0.785	0.763

criterions obtained by denoising CTF and CTF when the corresponding F-measure is the largest on *E. coli*, *S. cerevisiae* and human randomly negative datasets are shown in Table 1. From Table 1, we can see that, on all three kinds of organisms randomly negative datasets, *Denoising*^{Yu} CTF outperforms *Denoising*^{Hao} CTF with high AUC and nearly all other criterions. However, both *Denoising*^{Yu} CTF and *Denoising*^{Hao} CTF perform worse than the CTF. For example, on *E. coli* randomly negative dataset, comparing with the CTF, the AUC and Sn obtained by *Denoising*^{Yu} CTF decrease by 1%, and the other criterions decrease by more or less to a certain extent. On *S. cerevisiae* randomly negative datasets, comparing with the CTF, the AUC, Acc, Pre and F-measure obtained by *Denoising*^{Yu} CTF decrease by 2 or 3%, and the Sn drops by 7%. On human randomly negative datasets, comparing with the CTF, the AUC, Acc, Sn and F-measure obtained by *Denoising*^{Yu} CTF decrease by over 5%, and the Pre drops by 2%.

These results suggest that, on randomly negative dataset, the performance of *Denoising*^{Hao} CTF and *Denoising*^{Yu} CTF are not as good as that of CTF, and even worse than it. That is, these two kinds of denoising formulas, which is useful to construct phylogenetic tree, cannot improve the performance of PPIs prediction.

3.3. The performance on the artificially negative datasets

We then test the effect of two kinds of denoising CTF formulas on *E. coil*, *S. cerevisiae* and human artificially negative datasets, respectively. The evaluation criterions obtained by denoising CTF and CTF when the corresponding F-measure is the largest on three kinds of organisms artificially negative datasets are listed in Table 2. Table 2 show that, on all three kinds of organisms artificially negative datasets, *Denoising*^{Yu} CTF outperforms *Denoising*^{Hao} CTF with high AUC and all other criterions. However, both *Denoising*^{Yu} CTF and *Denoising*^{Hao} CTF perform worse than the CTF. For example, for *E. coli*, on 1-let dataset, comparing with the CTF, the AUC obtained by *Denoising*^{Yu} CTF decreases by 1%, Acc and F-measure decrease by 3%, Sn drops by 6%, and Sp and Pre are nearly same as the CTF obtained. On 2-let dataset, comparing with the CTF, AUC, Acc, Sn, Pre and F-measure obtained by *Denoising*^{Yu} CTF decrease by more than 1%, and Pre is nearly same as the CTF obtained. These results suggest that, on *E. coil* artificially negative dataset, by introducing the denoising formulas which is useful to construct phylogenetic tree, the PPIs prediction performance cannot improved.

On *S. cerevisiae*1-let datasets, comparing with the CTF, although Acc, Sn, Pre and F-measure obtained by *Denoising*^{Yu} CTF increase by 1%, the AUC increases only 0.1%. On 2-let dataset, comparing with the CTF, although Acc, Sn and F-measure obtained by *Denoising*^{Yu} CTF increase by 2–4%, the AUC only has

Table 2
The performance comparison of denoising CTF with CTF on shuffled negative set. The best predictions obtained are highlighted in bold.

Organism	Encoding methods	Evaluation criterions					
		AUC	Acc	Sn	Sp	Pre	F-measure
E. coli	1let-CTF	**0.957**	**0.891**	**0.882**	**0.899**	**0.898**	**0.891**
	1-let-*Denoising*^{Hao} CTF	0.910	0.824	0.848	0.799	0.809	0.823
	1-let-*Denoising*^{Yu} CTF	0.940	0.860	0.820	0.899	0.891	0.858
	2-let-CTF	**0.936**	**0.856**	**0.892**	**0.899**	**0.890**	**0.853**
	2-let-*Denoising*^{Hao} CTF	0.904	0.818	0.836	0.799	0.807	0.818
	2-let-*Denoising*^{Yu} CTF	0.927	0.841	0.882	0.879	0.887	0.839
S. cerevisiae	1let-CTF	0.956	0.884	0.868	**0.899**	0.896	0.883
	1-let-*Denoising*^{Hao} CTF	0.950	0.885	0.871	0.879	0.897	0.885
	1-let-*Denoising*^{Yu} CTF	**0.957**	**0.899**	**0.879**	0.879	**0.919**	**0.899**
	2-let-CTF	0.936	0.850	0.801	0.899	0.888	0.847
	2-let-*Denoising*^{Hao} CTF	0.935	0.866	0.837	0.899	0.893	0.867
	2-let-*Denoising*^{Yu} CTF	**0.937**	**0.872**	**0.845**	**0.899**	**0.894**	**0.871**
Human	1let-CTF	**0.913**	**0.831**	**0.863**	**0.799**	**0.811**	**0.830**
	1-let-*Denoising*^{Hao} CTF	0.822	0.748	0.795	0.799	0.776	0.744
	1-let-*Denoising*^{Yu} CTF	0.880	0.802	0.804	0.799	0.800	0.802
	2-let-CTF	**0.831**	**0.756**	**0.812**	**0.699**	**0.730**	**0.751**
	2-let-*Denoising*^{Hao} CTF	0.770	0.705	0.716	0.699	0.701	0.705
	2-let-*Denoising*^{Yu} CTF	0.798	0.726	0.753	0.699	0.715	0.725

0.1% improvement. These results suggest that, on *S. cerevisiae* artificially negative dataset, the *Denoising*^{Yu} CTF outperforms the CTF with 0.1% AUC improvement. However, this little improvement is insufficient to support the fact that the prediction performance can be improved by introducing the denoising procedure.

On human 1-let datasets, comparing with the CTF, AUC, Acc, Sn and F-measure obtained by *Denoising*^{Yu} CTF decrease by over 3%, the Pre decreases 1%. On 2-let dataset, comparing with the CTF, AUC, Acc, Sn and F-measure obtained by *Denoising*^{Yu} CTF decrease by over 3%, the Pre drops nearly 2%. These results suggest that, on human artificially negative dataset, the CTF also outperforms the *Denoising*^{Yu} CTF with over 3% improvement.

3.4. The performance of denoising formulas on the gene level

The two kinds of denoising formulas were proposed on the gene level in Chan et al. (2010). By introducing these denoising formulas, the well-grouped phylogenetic trees have been constructed. Therefore, we test the effect of these two denoising formulas on the gene level on *E. coli*, *S. cerevisiae* and human randomly negative datasets, respectively. That is, we encode protein sequences by codon composition, and introduce the denoising formulas (1) and (2) as the noise of the codon composition, respectively, then apply the signal-to-noise ratio as the input vectors for representing the proteins. Eq. (4) is also applied as the representation vector for protein pairs, and it is denoted as denoising codon. The evaluation criterions obtained by denoising codon and codon composition when the corresponding F-measure is the largest on all three kinds of organisms randomly negative datasets are shown in Table 3. From Table 3, we can see that, on both *E. coli* and *S. cerevisiae* PPIs datasets, *Denoising*^{Yu} codon outperforms *Denoising*^{Hao} codon with high AUC and all other criterions. However, both *Denoising*^{Yu} codon and *Denoising*^{Hao} codon perform worse than codon itself. For example, on *E. coli* randomly negative dataset, comparing with codon composition, the AUC, Acc and F-measure obtained by *Denoising*^{Yu} codon decrease by 3%, Sn drops by more than 7%, and Pre drops by 1%. On *S. cerevisiae* randomly negative datasets, comparing with the codon composition, the AUC, Acc and F-measure obtained by *Denoising*^{Yu} codon decrease by 3%, Sn drops by nearly 7%, and Pre drops by 1%. These results suggest that, on gene level, the PPIs prediction performance also cannot improved by introducing the denoising methods.

Table 3
The performance comparison of denoising codon composition with codon composition on randomly generated negative set. The best predictions obtained are highlighted in bold.

Organism	Encoding methods	Evaluation criterions					
		AUC	Acc	Sn	Sp	Pre	F-measure
E. coli	Codon	**0.897**	**0.812**	**0.825**	**0.799**	**0.805**	**0.812**
	Denoising^{Hao} codon	0.855	0.766	0.732	0.799	0.785	0.764
	Denoising^{Yu} codon	0.868	0.775	0.751	0.799	0.789	0.774
S. cerevisiae	Codon	**0.942**	**0.881**	**0.863**	**0.899**	**0.896**	**0.881**
	Denoising^{Hao} codon	0.887	0.811	0.783	0.899	0.802	0.806
	Denoising^{Yu} codon	0.911	0.847	0.794	0.899	0.888	0.843
Human	Codon	0.740	0.677	0.655	0.699	0.685	0.676
	Denoising^{Hao} codon	0.693	0.642	0.584	0.699	0.660	0.636
	Denoising^{Yu} codon	**0.743**	**0.680**	**0.660**	**0.699**	**0.687**	**0.679**

On human PPIs datasets, *Denoising*^{Yu} codon also outperforms *Denoising*^{Hao} codon with high AUC and all other criterions. While, *Denoising*^{Yu} codon obtain nearly same AUC and other criterions. For example, comparing with the codon, AUC, Acc, Pre and F-measure obtained by *Denoising*^{Yu} codon increase by 0.3 percent, the Sn increases only 0.5%. These results suggest that, on *Human* PPIs dataset, the *Denoising*^{Yu} formula on gene level has comparable performance with original formula. However, the prediction performance are still not improved by introducing the *Denoising*^{Yu} formula.

4. Discussion and conclusion

In this paper, we introduce the denoising idea which is proved to be useful in construction of phylogenetic trees into the PPIs prediction. Specially, we first encode the given protein sequence by the composition vector, and then introduce two denoising formulas proved to be useful in phylogenetic tree construction as

the noise vector, finally apply the signal-to-noise ratio as the input vector for representing the given protein. The concatenation of arithmetical and geometric average of protein vectors are used as the protein pair representation vector, which can not only maintain symmetry, but also make the protein representing vectors constructed uniquely from the protein pair representing vector. We test the effect of the denoising vectors on *E. coli*, *S. cerevisiae* and human randomly and artificially negative datasets, and compare it with the primal composition vectors. The evaluation criterions obtained by both two denoising vectors are not improved. These results suggest that, although the denoising methods can improve the performance of phylogenetic trees construction, it cannot improve the performance of PPIs prediction. That is, what is noise is dependent on the applications.

The reason that the lower accuracy of denoising methods here may be that the CTF first classifies 20 amino acids into seven classes based on the dipoles and volumes of the side chains, and then apply conjoint triad (3-string) composition to represent the given protein. That is, the denoising is already done by reducing the dimension, and further denoising will make information shrink. For testing this assumption, we do the experiments on composition vector without fusion of amino acids (that is a $20\times20\times20$ vector) and corresponding denoising vector. For concise, we take the *Helicobacter pylori* PPIs dataset as an example. And we found that the denosing methods also perform worse than composition vector without fusion of amino acids. Specially, when we test composition vector without fusion of amino acids (a $20\times20\times20$ vector) on *Helicobacter pylori* PPIs dataset, AUC reaches 0.80, while validated on Yu's denoising vector, AUC drops to 0.79.

The two types of denoising techniques we chosen have clear probability assumptions and did make a success in phylogenetic tree construction. We hypothesized that these two techniques may reveal some true sequence noise in proteins and may be useful to improve PPIs prediction. However, the computational results did not support our hypothesis. Thus, we concluded that 'noise' in phylogenetics may be 'information' in PPIs. This conclusion needs further confirmation by excluding the influence of the selection of denoising techniques in the future.

Since user-friendly and publicly accessible web-servers represent the future direction for developing practically more useful models, simulated methods, or predictors (Chou and Shen, 2009), we shall make efforts in our future work to provide a web-server for the method presented in this paper.

Acknowledgment

This work is supported by the National Natural Science Foundation of China (Nos. 10801131 and 10971223).

References

Arango-Argoty, G.A., Jaramillo-Garzón, J.A., Röthlisberger S., Castellanos-Dominguez, C.G., 2011. Prediction of protein subcellular localization based on variable-length motifs detection and dissimilarity based classification. In: Conference Proceedings: IEEE Engineering in Medicine and Biology Society, pp. 945–948.

Ben-Hur, A., 2005. Kernel methods for predicting protein–protein interactions. Bioinformatics 21, i38–i46.

Ben-Hur, A., Noble, W.S., 2006. Choosing negative examples for the prediction of protein–protein interactions. BMC Bioinf. 7 (Suppl. 1), S2.

Bock, J.R., Gough, D.A., 2001. Predicting protein–protein interactions from primary structure. Bioinformatics 17, 455–460.

Chan, R.H.F., Wang, R.W., Wong, J.C.F., 2010. Maximum Entropy Method for Composition Vector Method. Published Online: 23 December 2010 http://dx.doi.org/10.1002/9780470892107.ch27.

Chang, C.-C., Lin, C.-J., 2011. LIBSVM: a library for support vector machines. ACM Trans. Intell. Syst. Technol. 2:27, 1–27.

Chou, K.C., Zhang, C.T., 1995. Review: prediction of protein structural classes. Crit. Rev. Biochem. Mol. Biol. 30, 275–349.

Chou, K.C., 2001. Prediction of protein cellular attributes using pseudo amino acid composition. Proteins: Struct. Funct. Genet. 43, 246–255.

Chou, K.C., 2011. Some remarks on protein attribute prediction and pseudo amino acid composition (50th Anniversary Year Review). J. Theoret. Biol. 273, 236–247.

Chou, K.C., Cai, Y.D., 2002. Using functional domain composition and support vector machines for prediction of protein subcellular location. J. Biol. Chem. 277, 45765–45769.

Chou, K.C., Cai, Y.D., 2004. Prediction of protein subcellular locations by GOC-FunDCPseAA predictor. Biochem. Biophys. Res. Commun. 320, 1236–1239.

Chou, K.C., Cai, Y.D., 2006. Predicting protein–protein interactions from sequences in a hybridization space. J. Prot. Res. 5, 316–322.

Chou, K.C., Shen, H.B., 2006. Predicting eukaryotic protein subcellular location by fusing optimized evidence: theoretic K-nearest neighbor classifiers. J. Prot. Res. 5, 1888–1897.

Chou, K.C., Shen, H.B., 2008. Cell-PLoc: a package of Web servers for predicting subcellular localization of proteins in various organisms (updated version: Cell-PLoc 2.0: an improved package of web-servers for predicting subcellular localization of proteins in various organisms, Natural Science, 2010, 2, 1090–1103). Nature Prot. 3, 153–162.

Chou, K.C., Shen, H.B., 2009. Review: recent advances in developing webservers for predicting protein attributes. Natural Sci. 2, 63–92. (Openly Accessible at 〈http://www.scirp.org/journal/NS/〉).

Chou, K.C., Shen, H.B., 2010. A new method for predicting the subcellular localization of eukaryotic proteins with both single and multiple sites Euk-mPLoc 2.0. PLoS One 5, e9931.

Chou, K.C., Shen, H.B., 2010. Plant-mPLoc: a top-down strategy to augment the power for predicting plant protein subcellular localization. PLoS One 5, e11335.

Chou, K.C., Wu, Z.C., Xiao, X., 2011. iLoc-Euk: a multi-label classifier for predicting the subcellular localization of singleplex and multiplex eukaryotic proteins. PLoS One 6, e18258.

Chou, K.C., Wu, Z.C., Xiao, X., 2012. iLoc-Hum: using accumulation-label scaleto predict subcellular locations of human proteins with both single and multiple sites. Mol. Biosyst. 8 (2), 629–641.

Deng, L., Guan, J.H., Dong, Q.W., Zhou, S.G., 2011. SemiHS: an iterative semi-supervised approach for predicting protein–protein interaction hot spots. Protein Pept. Lett. 18, 896–905.

Deng, N.Y., Tian, Y.J., Zhang, C.H., 2012. Support Vector Machines: Theory, Algorithms, and Extensions. Science Press, Beijing.

Dijk, A., Bosch, D., Braak, C., Krol, A., Ham, R., 2008. Predicting sub-Golgilocalization of type II membrane proteins. Bioinformatics 24 (16).

Esmaeili, M., Mohabatkar, H., Mohsenzadeh, S., 2010. Using the concept of Chou's pseudo amino acid composition for risk type prediction of human papilloma viruses. J. Theoret. Biol. 263, 203–209.

Fields, S., Song, O., 1989. A novel genetic system to detect protein–protein interactions. Nature 340, 245–246.

Gavin, A.C., Boche, M., Krause, R., Grandi, P., Marzioch, M., Bauer, A., Schultz, J., Rick, J., Michon, A., Cruciat, C., 2002. Functional organization of the yeast proteome by systematic analysis of protein complexes. Nature 415, 141–147.

Georgiou, D.N., Karakasidis, T.E., Nieto, J.J., Torres, A., 2009. Use of fuzzy clustering technique and matrices to classify amino acids and its impact to Chou's pseudo amino acid composition. J. Theoret. Biol. 257, 17–26.

Gomez, S.M., Noble, W.S., Rzhetsky, A., 2003. Learning to predict protein–protein interactions from protein sequences. Bioinformatics 19, 1875–1881.

Gribskov, M., Robinson, N.L., 1996. Use of receiver operating characteristic (roc) analysis to evaluate sequence matching. Comput. Chem. 20, 25–33.

Hao, B.L., Qi, J., Wang, B., 2003. Prokaryotic phylogeny based on complete genomes without sequence alignment. Mod. Phys. Lett. B 2, 1–4.

Ho, Y., Gruhler, A., Heilbut, A., Bader, G.D., Moore, L., Adams, S., Millar, A., Taylor, P., Bennett, K., Boutilier, K., Yang, L., Wolting, C., Donaldson, I., Schandorff, S., Shewnarane, J., Vo, M., Taggart, J., Goudreault, M., Muskat, B., Alfarano, C., Dewar, D., Lin, Z., Michalickova, K., Willems, A.R., Sassi, H., Nielsen, P.A., Rasmussen, K.J., Andersen, J.R., Johansen, L.E., Hansen, L.H., Jespersen, H., Podtelejnikov, A., Nielsen, E., Crawford, J., Poulsen, V., Sørensen, B.D., Matthiesen, J., Hendrickson, R.C., Gleeson, F., Pawson, T., Moran, M.F., Durocher, D., Mann, M., Hogue, C.W., Figeys, D., Tyers, M., 2002. Systematic identification of protein complexes in *Saccharomyces cerevisiae* by mass spectrometry. Nature 415, 180–183.

Hoglund, A., Donnes, P., Blum, T., Adolph, H., Kohlbacher, O., 2006. MultiLoc: prediction of protein subcellular localization using N-terminal targeting sequences, sequence motifs and amino acid composition. Bioinformatics 22 (10), 1158–1165.

Hu, L., Huang, T., Shi, X., Lu, W.C., Cai, Y.D., Chou, K.C., 2011. Predicting functions of proteins in mouse based on weighted protein–protein interaction network and protein hybrid properties. PLoS One 6, e14556.

Ito, T., Chiba, T., Ozawa, R., Yoshida, M., Hattori, M., Sakaki, Y., 2001. A comprehensive two-hybrid analysis to explore the yeast protein interactome. Proc. Natl. Acad. Sci. 98, 4569–4574.

Jiang, M., Anderson, J., Gillespie, J., Mayne, M., 2008. uShuffle: a useful tool for shuffling biological sequences while preserving the *k*-let counts. BMC Bioinf. 9, 192.

Leslie, C., Eskin, E., Noble, W.S., 2002. The spectrum kernel: a string kernel for SVM protein classification. In: Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing, pp. 564–575.

Ma, D.C., Diao, Y.B., Guo, Y.Z., Li, Y.Z., Zhang, Y.Q., Wu, J., Li, M.L., 2011. A novel method to predict protein–protein interactions based on the information of

protein–protein interaction networks and protein sequence. Protein Pept. Lett. 18, 906–911.
Mak, M., Guo, J., Kung, S., 2008. PairProSVM: protein subcellular localization based on local pairwise profile alignment and SVM. IEEE/ACM Trans. Comput. Biol. Bioinf. 5 (3), 416–422.
Mei, S., Wang, Fei, 2010. Amino acid classification based spectrum kernel fusion for protein subnuclear localization. BMC Bioinf. 11 (Suppl 1), S17.
Mohabatkar, H., 2010. Prediction of cyclin proteins using Chou's pseudo amino acid composition. Protein Pept. Lett. 17, 1207–1214.
Mohabatkar, H., Mohammad Beigi, M., Esmaeili, A., 2011. Prediction of GABA(A) receptor proteins using the concept of Chou's pseudo-amino acid composition and support vector machine. J. Theoret. Biol. 281, 18–23.
Najafabadi, H., Salavati, R., 2008. Sequence-based prediction of protein–protein interactions by means of codon usage. Genome Biol. 9, R87.
Noble, W.S., 2004. Support vector machine applications in computational biology. In: Schoelkopf, B., Tsuda, K., Vert, J.-P. (Eds.), Kernel Methods in Computational Biology. MIT Press, Cambridge, MA, pp. 71–92.
Pierleoni, A., Luigi, P., Fariselli, P., Casadio, R., 2006. BaCelLo: a balanced localization predictor. Bioinformatics 22 (14), e408–e416.
Qiu, Z., Wang, X., 2012. Prediction of protein–protein interaction sites using patch-based residue characterization. J. Theoret. Biol. 293C, 143–150.
Ren, L.H., Shen, Y.Z., Ding, Y.S., Chou, K.C., 2011. Bio-entity network for analysis of protein–protein interaction networks. Asian J. Control 13, 726–737.
Ren, X.W., Wang, Y.C., Wang, Y., Zhang, X.S., Deng, N.Y., 2011. Improving accuracy of protein–protein interaction prediction by considering the converse problem for sequence representation. BMC Bioinf. 12, 409.
Shen, J.W., Zhang, J., Luo, X.M., Zhu, W.L., Yu, K.Q., Chen, K.X., Li, Y.X., Jiang, H.L., 2007. Predicting protein–protein interactions based only on sequences information. Proc. Natl. Acad. Sci. 104, 4337–4341.
Vapnik, V., 1995. The Nature of Statistical Learning Theory. Springer, New York.
Vapnik, V., 1998. Statistical Learning Theory. Wiley.
Wang, P., Xiao, X., Chou, K.C., 2011. NR-2L: a two-level predictor for identifying nuclear receptor subfamilies based on sequence-derived features. PLoS One 6, e23505.
Wu, G., Chang, E.Y., 2003. Class-boundary alignment for imbalanced dataset learning. In: ICML 2003 Workshop on Learning from Imbalanced Data Sets.
Wu, Z.C., Xiao, X., Chou, K.C., 2011. iLoc-gpos: a multi-layer classifier for predicting the subcellular localization of singleplex and multiplex gram-positive bacterial proteins. Protein Pept. Lett. 19, DOI: BSP/ PPL/ E pub/0380 [pii].
Xia, J.F., Han, K., Huang, D.S., 2010. Sequence-based prediction of protein–protein interactions by means of rotation forest and autocorrelation descriptor. Protein Pept. Lett. 17, 137–145.
Xiao, X., Wu, Z.C., Chou, K.C., 2011. A multi-label classifier for predicting the subcellular localization of gram-negative bacterial proteins with both single and multiple sites. PLoS One 6, e20592.
Yang, J., Jiang, X.F., 2010. A novel approach to predict protein–protein interactions related to Alzheimer's disease based on complex network. Protein Pept. Lett. 17, 356–366.
Yu, C.Y., Chou, L.C., Chang, D.T.H., 2010. Research article predicting protein–protein interactions in unbalanced data using the primary structure of proteins. BMC Bioinf. 11, 167.
Yu, Z.G., Zhou, L.Q., Anh, V., Chu, K.H., Long, S.C., Deng, J.Q., 2005. Phylogeny of prokaryotes and chloroplasts revealed by a simple composition approach on all protein sequences from whole genome without sequence alignment. J. Mol. Evol. 60, 538–545.
Park, Yungki, 2009. Critical assessment of sequence-based protein–protein interaction prediction methods that do not require homologous protein sequences. BMC Bioinf. 10, 419.
Zhang, Y.N., Pan, X.Y., Huang, Y., Shen, H.B., 2011. Adaptive compressive learning for prediction of protein–protein interactions from primary sequence. J. Theoret. Biol. 283, 44–52.
Zhou, G.P., 2011. The disposition of the LZCC protein residues in wenxiang diagram provides new insights into the protein–protein interaction mechanism. J. Theoret. Biol. 284, 142–148.
Zhu, H., Bilgin, M., Bangham, R., Hall, D., Casamayor, A., Bertone, P., Lan, N., Jansen, R., Bidlingmaier, S., Houfek, T., Mitchell, T., Miller, P., Dean, R.A., Gerstein, M., Snyder, M., 2001. Global analysis of protein activities using proteome chips. Science 193, 2101–2105.