

SEQUENCE-BASED PROTEIN-PROTEIN INTERACTION PREDICTION VIA SUPPORT VECTOR MACHINE*

Yongcui WANG · Jiguang WANG · Zhixia YANG · Naiyang DENG

DOI: 10.1007/s11424-010-0214-z

Received: 2 November 2009 / Revised: 29 January 2010

©The Editorial Office of JSSC & Springer-Verlag Berlin Heidelberg 2010

Abstract This paper develops sequence-based methods for identifying novel protein-protein interactions (PPIs) by means of support vector machines (SVMs). The authors encode proteins not only in the gene level but also in the amino acid level, and design a procedure to select negative training set for dealing with the training dataset imbalance problem, i.e., the number of interacting protein pairs is scarce relative to large scale non-interacting protein pairs. The proposed methods are validated on PPIs data of *Plasmodium falciparum* and *Escherichia coli*, and yields the predictive accuracy of 93.8% and 95.3%, respectively. The functional annotation analysis and database search indicate that our novel predictions are worthy of future experimental validation. The new methods will be useful supplementary tools for the future proteomics studies.

Key words Imbalance problem, protein-protein interactions, sequence-based, support vector machine.

1 Introduction

Identification of the interactions among proteins is crucial to illustrate their functions, and furthermore, it can help us to understand the mechanisms of some essential biological processes such as complex diseases, aging process, and so on^[1]. 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^[2–3], mass spectrometry^[4–5], protein chip^[6], and so on. Compared with

Yongcui WANG

College of Science, China Agricultural University, Beijing 100083, China; Key Laboratory of Adaptation and Evolution of Plateau Biota, Northwest Institute of Plateau Biology, Chinese Academy of Sciences, Xining 810008, China. Email: wangyc82@yahoo.cn.

Jiguang WANG

Institute of Systems Science, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190, China.

Zhixia YANG

College of Mathematics and Systems Science, Xinjiang University, Urumchi 830046, China.

Naiyang DENG

College of Science, China Agricultural University, Beijing 100083, China. Email: dengnaiyang@vip.163.com.

*This research is supported by the Key Project of the National Natural Science Foundation of China under Grant No. 10631070, the National Natural Science Foundation of China under Grant Nos. 10801112, 10971223, 11071252, and the Ph.D Graduate Start Research Foundation of Xinjiang University Funded Project under Grant No. BS080101.

these costly and time-consuming biochemical experiments, the computational methods have attracted much attention due to their low costing and competitive performance.

Current computational methods for PPIs prediction require a large amount of genomic data sources, such as, Gene Ontology (GO) annotations, gene expressions, evolution information. However, usually some of them are not available for some important genes. Sequence-based methods in the amino acid level then become popular because they only demand the information of amino acid sequences. And the highest accuracy of these methods is about 80%^[7], such as the methods by Martin, et al.^[8] and Chou and Cai^[9].

All above works focus on using the machine learning methods to learn understandable rules from these existing PPIs and furthermore to predict novel interactions. One key problem in machine learning is to extract features from protein pairs, and previous studies explore sequence features only in the amino acid level. However, the knowledge that codon usage is correlated with expression level has been widely accepted^[10], and the hypothesis of some function-specific codon preferences has been confirmed by experiments^[11]. Furthermore, Naiafjadi and Salavati^[12] proposed a sequence-based method by constructing the sequence features in the gene level instead of the amino acid level. By using a naïve Bayesian network to combine the frequencies of all codons, the encouraging predictive results were obtained. Inspired by these results, in this article, we encode the proteins in the gene level by means of codon. Besides codon usage, we note that Shen, et al.^[13] developed a conjoint triad feature (CTF) to encode proteins. And with support vector machine (SVM) as the classifier, they obtained a high prediction accuracy of 83.9% when predicting human PPIs. Following their work, we also introduce the CTF into our predictive model, and make a comparison between this CTF-based and the above codon-based encoding method. After encoding proteins as real-value feature vectors, we use SVM as the classifier to get the novel PPIs just like Shen, et al. did in [13]. SVMs are known to provide state-of-the-art performance in many applications^[14], in particular in computational biology^[15]. And identification of PPIs can be addressed as a two-classification problem: Determining whether a given pair of proteins is interacting or not. So, here, two-class SVM with codon usage and the CTF are used to predict PPIs.

Another problem in machine learning is to construct the gold-standard datasets. Gold-standard positive dataset is not a problem since many known PPIs have been deposited in some curated database, such as IntAct^[16], DIP^[17], BIND^[18], and HPRD^[19]. Gold-standard negative dataset generally cannot be obtained by experimental data and should be constructed approximately based on non-interacting protein pairs (unlabeled dataset). Furthermore, the gold-standard positive datasets are scarce relative to large scale unlabeled data. So, here, we select a gold-standard negative dataset from unlabeled dataset to deal with the training dataset imbalance problem. Then the two-class SVM is trained on the gold-standard positive dataset and the selected approximate gold-standard negative dataset. We validate the proposed methods on the PPIs data of *Plasmodium falciparum* (*P. falciparum*) and *Escherichia coli* (*E. coli*), and yields the predictive accuracy of 93.8% and 95.3%, respectively. They are further evaluated on *P. falciparum* and *E. coli* independent PPIs datasets, and achieve the test sensitivity of 74.7% and 84.6%, respectively. The functional annotation analysis and database search indicate that our novel predictions are worthy of future experimental validation.

The paper is structured as follows. We begin by encoding the protein pairs by using sequence information both in the gene level and in the amino acid level. Then we introduce the procedure to select gold-standard negative dataset. We also compare our methods with existing methods on *P. falciparum* and *E. coli* PPIs datasets. Finally, the discussions and conclusions are presented.

2 Materials and Methods

2.1 Input Feature Vectors

Sequence-based feature in the amino acid level

Based on the dipoles and volumes of the side chains, the 20 amino acids can be classified into seven classes: $\{A, G, V\}$, $\{I, L, F, P\}$, $\{Y, M, T, S\}$, $\{H, N, Q, W\}$, $\{R, K\}$, $\{D, E\}$, $\{C\}$. Thus, a $343(7 \times 7 \times 7)$ -dimension vector is used to represent a given protein, where each element of this vector is the frequency for a kind of conjoint triad appearing in the corresponding protein sequence, and we call this kind of feature as the conjoint triad feature (CTF).

Sequence-based feature in the gene level

We represent a given open reading frame (ORF) by a 64-dimension vector, where each element of this vector is the frequency for a kind of codon appearing in the corresponding ORF, and we call this kind of feature as codon mode.

Another method for encoding ORF as a real-value input vector is incorporating 64 codons into 20 amino acids, that is, using a 20-dimensional vector to represent ORF, each element of this vector is the frequency of a sort of amino acid appearing in the corresponding ORF, and we call this kind of feature as codon merger mode.

There are two ways to encode protein-protein pair as the input vector:

1) Concatenating the protein pairs

A pair of protein A and protein B is represented by concatenating the protein feature vectors F_A and F_B . That is the input feature vector F_{AB} for a protein pair A-B is calculated as follows:

$$F_{AB} = F_A \oplus F_B, \quad (1)$$

where \oplus is the concatenation operator. To make predictive results for protein pair A-B identical to B-A, we train and test on both F_{AB} and F_{BA} , and report the average predictive results in numerical experiments.

2) Distance of protein pairs

A pair of protein A and protein B is represented by a distance vector. $D_k^{AB} = |f_k^A - f_k^B|$ is used to measure the distance between protein A and B. So the input feature vector D_{AB} for a protein pair A-B is calculated as follows:

$$D_{AB} = (f_1^{AB}, f_2^{AB}, \dots, f_m^{AB})^T, \quad f_k^{AB} = |f_k^A - f_k^B|, \quad k = 1, 2, \dots, m, \quad (2)$$

where f_k^A, f_k^B is the element of the protein feature vectors F_A and F_B , respectively, and $m = 343$ for the CTF, $m = 64$ for codon mode, $m = 20$ for codon merger mode.

Following our previous work^[20], for codon mode, we use the distance vector to represent a pair of proteins, while for codon merger mode, the concatenation operator is used. Specially, SVM_{codon} is used to denote the SVM with codon mode (using 64-dimensional vector to represent protein-protein pairs), and $SVM_{\text{codon merger}}$ is used to denote the SVM with codon merger mode (using 40-dimensional vector to represent protein-protein pairs). For the CTF, the distance vector is used for avoiding computational difficulty, and SVM_{CTF} is used to denote the SVM with the CTF.

2.2 Training SVM Using Labeled and Unlabeled Data

Identification of PPIs can be addressed as the two-classification problem: Determining whether a given pair of proteins is interacting or not. If we treat all the non-interacting protein-protein pairs (unlabeled pairs) as the negative dataset and all known interacting pairs as the gold-standard positive data, the imbalance problem will arise due to the gold-standard positive

dataset is scarce relative to large scale negative dataset. To maintain a balance between gold-standard positive and negative dataset in SVM training procedure, we select the gold-standard negative dataset which has the nearly same size of the gold-standard positive dataset, and then perform the two-class SVM. This gold-standard negative dataset should be a good representation of the entire negative dataset, so we select the data points which can hold on the main distribution of the whole dataset. Specially, we first calculate the mean vector of the whole negative data points; secondly, compute the distance between each data point and the mean vector; finally, select the data points far from the mean vector and make the chosen dataset with the nearly same size of the gold-standard positive dataset. After selecting the suitable negative dataset, we implement two-class SVM to predict PPIs. This method is denoted by SVM-SN.

We also randomly select the negative dataset, and then use two-class SVM on gold-standard positive dataset and this random negative dataset to perform the predictive task. It is denoted by SVM-random. We compare the performance of SVM-SN with the average results of SVM-random in experimental section.

2.3 Benchmark Datasets and SVM Implementation

Here, PPIs on two different organisms: *P. falciparum* and *E. coli* are used to validate the performance of the proposed predictive models. *P. falciparum* is eukaryotic, while *E. coli* is prokaryotic. The detailed information of these benchmark datasets can be found in Table 1 in [12]. The genome sequences for them can also be downloaded from [12], and the proteome sequences can be download in [21]. Specially, for *P. falciparum*, the positive and negative sets are the same as the ‘gold standard sets’ in [12], while for *E. coli*, we exclude the interactions which contain missing proteins in the corresponding genome and proteome sequence datasets. Thus, the number of interactions is 7689 and 6954 for *P. falciparum* and *E. coli*, respectively.

We train the SVM-SN and SVM-random by using LibSVM^[22]. In the implementation of SVM-SN and SVM-random, 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 benchmark dataset is split into 10 subsets of roughly equal size, each subset is then taken in turn as a test set, and we train SVM-SN and SVM-random on the remaining nine sets. The performances of our proposed methods are shown by receiver operating curve (ROC)^[23]. Furthermore, the evaluation criterions: AUC (area under the ROC curve), sensitivity= $TP/(TP + FN)$, specificity= $TN/(TN + FP)$, precision= $TP/(TP + FP)$ and accuracy = $(TP + TN)/(TP + TN + FP + FN)$ are also used to display the performance of the proposed predictive methods.

3 Results

In our previous work^[20], we shown that the SVM_{codon merger} outperforms the SVM_{codon} model not only on the randomly negative set but also on the well-chosen negative set, so we only compare the performance of the SVM_{CTF} and SVM_{codon merger} with PIC model^[12] on the benchmark datasets.

3.1 The Performance on *P. falciparum*

1) Comparison of predictive methods

We plot the ROC and the evaluation criterions for each method on *P. falciparum* in Figure 1. As shown in Figure 1, both the SVM_{codon merger} and SVM_{CTF} models outperform the PIC model not only on the randomly negative set but also on the well-chosen negative set, and for

the well-chosen negative set, the performance of SVM_{CTF} is comparable with SVM_{codon merger}, while for the randomly negative set, SVM_{CTF} performs a little well than SVM_{codon merger}. For example, a AUC of 0.978 is achieved by the SVM_{CTF}-SN, while a AUC of 0.975 is obtained by the SVM_{codon merger}-SN. The sensitivity of SVM_{CTF}-random is 0.869, and is 0.843 for SVM_{codon merger}-random.

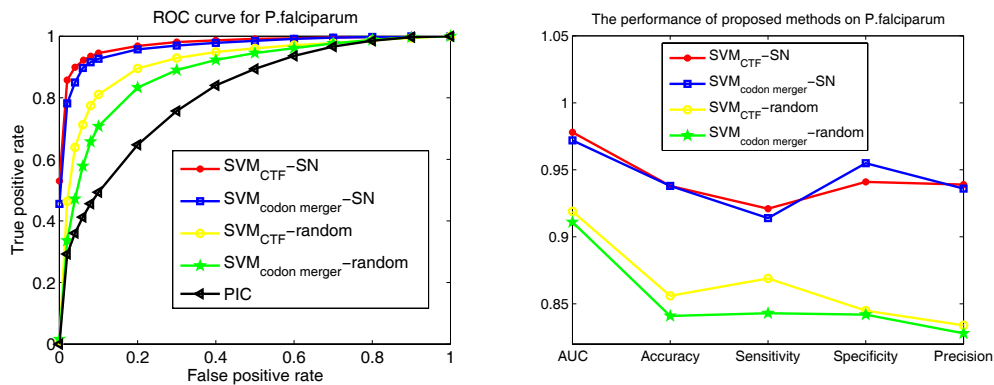


Figure 1 The performance of proposed methods on P. falciparum

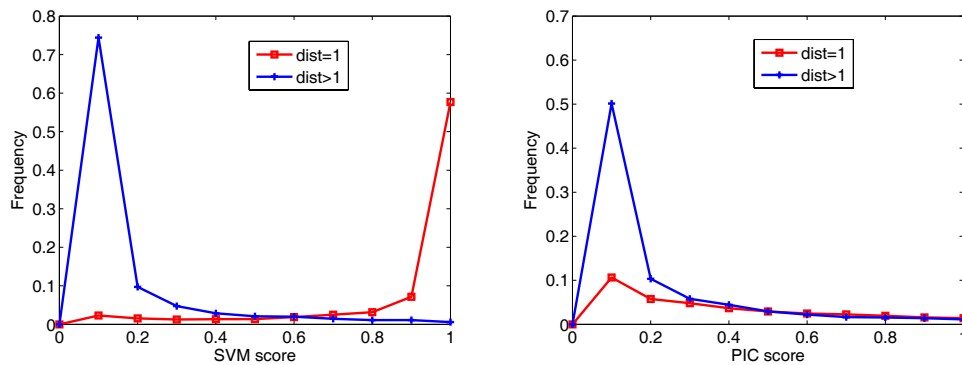


Figure 2 The relation between predictive score and network distance for SVM_{CTF}-SN (top) and PIC model (down) on P. falciparum PPIs Data

2) SVM scores partly capture the topological features of P. falciparum PPI network

In the previous subsection we show that our SVM_{CTF}-SN outperforms the PIC method. Here we would like to further explore the underlying rationale. Possible reason is that our SVM_{CTF}-SN method has implicitly learned important information not explicitly used for predicting. To show this, we study the correlation between the SVM scores obtained by SVM_{CTF}-SN model and the topological distances between proteins in PPI network. We define the network distance of a protein pair as the length of the shortest path between proteins in the network, and we define the SVM scores as the probabilistic output^[24] for each protein pair. We expect that SVM scores have a high value for interacting pairs in the network, and a low value for non-interacting pairs in the network.

To examine the relationship between SVM scores and the network distances, we divide the distances (dist) for protein pairs in the network into two groups: dist= 1 and dist> 1, which indicates the interacting pairs and non-interacting pairs, respectively. We plot the distributions of SVM scores with respect to each distance group in Figure 2. Figure 2 shows that SVM scores

and network distances are somehow correlated, i.e., the higher the SVM score is, the protein pair will be more closer in the PPI network. E.g., for the distance group $\text{dist} = 1$, most of the interacting protein pairs have the SVM scores of 1, while for distance group $\text{dist} > 1$, the SVM score is about 0.1. While, the PIC score cannot reflect this correlation, due to both two distance group have nearly the same peak. These results verify that SVM_{CTF-SN} might have implicitly learned important information about the topology of PPI network. These facts also partly explain the underlying reason that SVM_{CTF-SN} is effective and efficient in *P. falciparum* PPIs prediction.

3) The performance of proposed methods on *P. falciparum* independent PPIs Data

We evaluate the performance of SVM_{CTF-SN}, SVM_{codon-SN} and SVM_{codon merger-SN} on an independent dataset^[25] which is generalized by yeast two-hybrid experiments. This dataset includes 2,823 interactions covering 1,267 proteins. We generate the negative dataset by using the positive dataset: for example, AB and CD are interaction pairs, thus AC, AD, BC and BD could be the negative pairs^[13], that is there are 11,288 ($4 \times (2823 - 1)$) non-interactions which could be incorporated into the test dataset. Two test datasets are used to test the generalization ability of our methods: the first dataset contains 2,823 interactions and randomly selected 2,823 non-interactions, while the second one contains 2,823 interactions and the entire 11,288 non-interactions. We train SVM_{codon merger-SN}, SVM_{codon-SN} and SVM_{CTF-SN} on the gold-standard positive set and well-chosen negative set, and test on two test datasets respectively. Because the ability of predictive model to uncover the novel interacting pairs is the people most focus on, we show the test sensitivity in the left of Figure 3. From it, we can see that although two test datasets contain positive (Pos) and negative (Neg) data points with different ratio (1:1 and 1:4), there have been a little difference on the test results between them for both three methods.

In our previous work^[20], we show that, although with respect to the low FPR (false positive rate), the TPR (true positive rate) of SVM_{codon merger-SN} is higher than that of SVM_{codon-SN} on both two test datasets, both codon and codon merger are not suitable for the physical interaction. Fortunately, SVM_{CTF-SN} achieves a test sensitivity of about 75% on both two test datasets (the left of Figure 3), which is much higher than that of SVM_{codon-SN} and SVM_{codon merger-SN}.

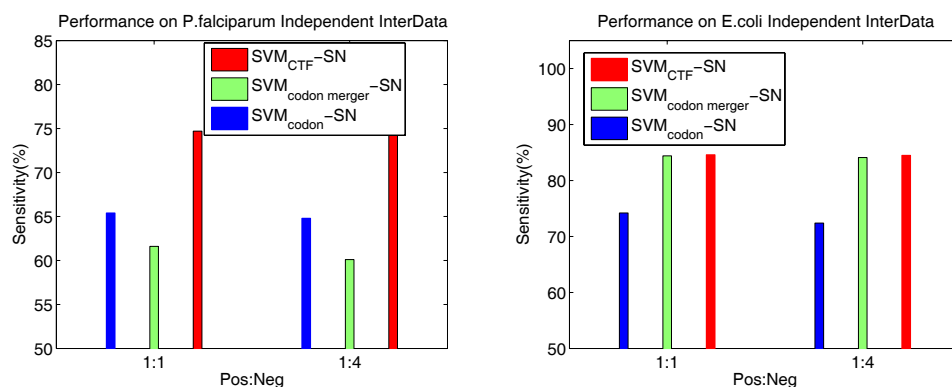


Figure 3 The performance of proposed methods on *P. falciparum* (left) and *E. coli* (right) independent PPIs datasets

We confirm the top ten predicted interacting pairs by GO function annotation analysis and database search. Since the most proteins do not interact with each other in the real-world^[26–27], we list the predicted interactions obtained by SVM_{CTF-SN} on the second test

dataset (Pos: Neg=1:4) in Table 1. We find evidences for all top ten predicted interactions in Gene DB^[28] and PlasmoDB^[29]. These results may suggest that SVM_{CTF}-SN can help to discover novel physical interaction on *P. falciparum*.

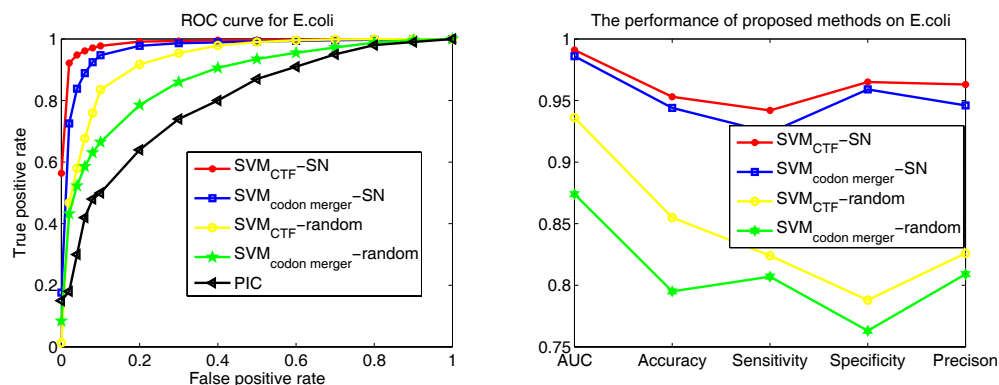


Figure 4 The performance of proposed methods for *E. coli*. SVM_{CTF} outperforms other methods on all evaluation criterions

Table 1 The top ten novel predictions by our SVM_{CTF}-SN method on *P. falciparum* independent PPI data

Rank	Protein pair	Annotation	Evidence
1	PFI0975c	Location: chromosome Pf3D7_09	Gene DB
	PF14_0726	Location chromosome Pf3D7_09	
2	PFB0815w	Calcium-dependent protein kinase	PlasmoDB
	PFF1440w	SET domain protein	
3	PF14_0280	Phosphotyrosyl phosphatase activator	PlasmoDB
	PF10_0262	Conserved Plasmodium protein, unknown function	
4	PFE1325w	Iconserved Plasmodium protein, unknown function	PlasmoDB
	PFD1155w	Erythrocyte binding antigen	
5	PF14_0614	Hydrolase activity	PlasmoDB
	PFL1930w	Nucleotide binding	
6	PFD0100c	Surface-associated interspersed gene	PlasmoDB
	PFL0815w	DNA-binding chaperone	
7	PFI0225w	Ubiquitin thiolesterase activity	PlasmoDB
	MAL7P1.167	Conserved Plasmodium protein, unknown function	
8	PF14_0495	Rhoptry neck protein	PlasmoDB
	PF10_0324	Conserved Plasmodium protein, unknown function	
9	PF07_0043	60S ribosomal protein	PlasmoDB
	PFE0040c	Mature parasite-infected erythrocyte surface antigen	
10	PF11_0504	Plasmodium exported protein (hyp11), unknown function	PlasmoDB
	PFL1855w	Cell cycle control proteine	

3.2 The Performance on *E. coli*

1) Comparison of predictive methods

For *E. coli*, the ROC curves and evaluation criterions are drawn for each method in Figure 4. Following the *P. falciparum* subsection, we only show the performance of SVM_{CTF}, SVM_{codon merger} and the PIC model. Both SVM_{CTF} and SVM_{codon merger} outperform PIC model not only on the randomly negative set but also on the well-chosen negative set, while

SVM_{CTF} performs better than $SVM_{codon\ merger}$ not only on randomly selected negative set but also on the well-chosen negative set.

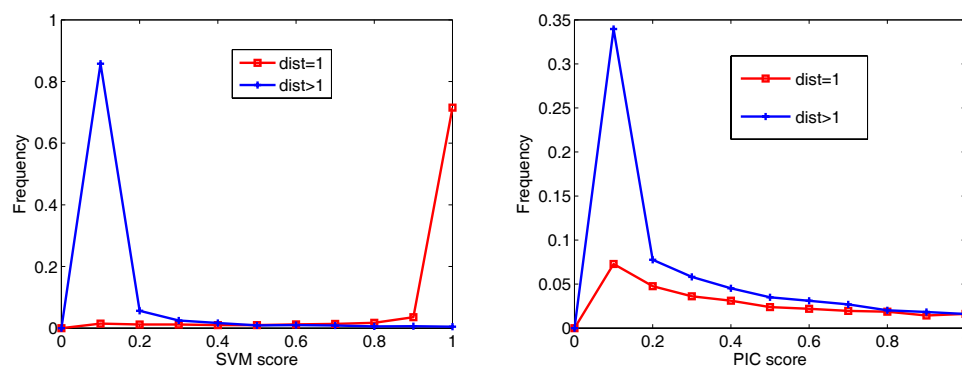


Figure 5 The relation between predictive score and network distance for SVM_{CTF-SN} (top) and PIC model (down) on *E. coli* PPIs Data

2) SVM scores partly capture the topological features of *E. coli* PPI network

We test whether SVM_{CTF-SN} can also capture the topological features in the *E. coli* PPI network. We plot the distributions of SVM scores with respect to each distance group ($dist=1$ and $dist>1$) in Figure 5. Figure 5 shows that the SVM score is more closely associated with network distance than the PIC score. E.g., for the distance group $dist=1$, most of the interacting protein pairs have the SVM scores of 1, while for distance group $dist>1$, the SVM score is about 0.1. While, both two distance group have nearly the same highest PIC score. These results verify that SVM_{CTF-SN} may also have implicitly learned the topology of *E. coli* PPI network.

3) The performance of proposed methods on *E. coli* independent PPIs Data

We evaluate the performance of SVM_{CTF-SN} , $SVM_{codon-SN}$, and $SVM_{codon\ merger-SN}$ on an independent *E. coli* PPI dataset which is collected by Su, et al.^[30]. This dataset contains 14,536 experimented physical interactions, by deleting the interactions which is present in the benchmark dataset, remains 10,529 interactions. We generate the negative set as the same way as in *P. falciparum* subsection, and 42,112 non-interactions are generated. We use two test datasets to test the performance of our methods: The first test dataset contains 10,529 interactions and randomly selected 10,529 non-interactions, while the second one contains 10,529 interactions and 42,112 non-interactions. We train $SVM_{codon-SN}$, $SVM_{codon\ merger-SN}$, and SVM_{CTF-SN} on the gold-standard positive and well chosen gold-standard negative *E. coli* PPI dataset, and test on these two test datasets respectively. The sensitivity for each method on each test dataset is shown in Figure 3 (right). On the two test dataset, both SVM_{CTF-SN} and $SVM_{codon\ merger-SN}$ perform well than $SVM_{codon-SN}$, and the average sensitivity is 84.6% and 84.4% for SVM_{CTF-SN} and $SVM_{codon\ merger-SN}$, respectively, while is 74.6% for $SVM_{codon-SN}$.

Following the *P. falciparum* subsection work, we confirm the top ten predicted interacting pairs by GO function annotation analysis and database search. We also list the predicted interactions obtained by SVM_{CTF-SN} on the second test dataset (Pos: Neg=1:4) in Table 2. We also find evidences for all top ten predicted interactions in EcoCyc^[31] and EcID^[32]. These results may suggest that SVM_{CTF-SN} can also help to discover novel physical interaction on *E. coli*.

Table 2 The top ten novel predictions by our SVM_{CTF-SN} method for E. coli independent PPI data

Rank	Protein pair	Annotation	Evidence
1	rpsK	Ribosomal protein S11	EcoCyc
	rpsD	Ribosomal protein S4	
2	ygdE	Ribose methyl-transferase	EciD
	ygeR	Uncharacterized lipoprotein	
3	rpsE	RNA binding	EcoCyc
	rplR	RNA binding	
4	cbrC	CreB-regulated gene C protein	EciD
	nadE	Nitrogen regulatory protein	
5	rpsK	Ribosomal protein S11	EciD
	rplQ	Ribosomal protein L17	
6	fliA	RNA polymerase sigma factor for flagellar operon	EciD
	rpoC	DNA-directed RNA polymerase subunit beta	
7	pth	Peptidyl-tRNA hydrolase	EciD
	rpsB	Ribosomal protein S2	
8	rpsL	Ribosomal protein S12	EciD
	rpsD	Ribosomal protein S4	
9	ygdE	Putative RNA 2-O-ribose methyltransferase ygdE	EciD
	yqjI	Uncharacterized protein	
10	nagC	N-acetylglucosamine repressor	EcoCyc
	nagB	Glucosamine-6-phosphate isomerase	

4 Discussion and Conclusion

In this paper, the sequence-based methods are proposed to predict PPIs. We extract sequence features both in the gene level and in the amino acid level. Specially, codon, codon merger mode and the CTF are used to represent proteins, and the distance and concatenation operator are applied to encode a pair of proteins as the feature vector. By implementing SVM_{codon merger-SN} and SVM_{CTF-SN} model on imbalance problem, the significant improvement in prediction can be obtained on both two kinds of organisms. For testing the generalization ability of SVM_{codon merger-SN} and SVM_{CTF-SN}, we train on the gold-standard positive datasets and well-chosen negative dataset and test on the independent interactions of *P. falciparum* and *E. coli*, respectively. For *P. falciparum*, SVM_{CTF-SN} achieves the test sensitivity of 74.7% on physical interaction generalized by yeast two-hybrid experiments. For *E. coli*, SVM_{CTF-SN} gets the test sensitivity of 84.6% on the experimental physical interactions. The good generalization ability of SVM_{CTF-SN} can be seen.

To explain why our SVM_{CTF-SN} model works well, we correlate the SVM scores obtained by SVM_{CTF-SN} with the distances of protein pairs in the PPI network, and then we reveal significant correlations between SVM scores and network distances for both two organisms, that is, the protein pair with higher SVM score tends to be closer in the PPI network (Figure 2, Figure 5). These results further exhibit the usefulness of our methods for predicting the interacting and non-interacting protein pairs.

We compare our SVM_{CTF-SN} with the works in [13]. For SVM_{CTF-SN}, although we introduce the CTF encoding methods^[13], the kernel and predictive model are different with the method in [13]. Specially, after using the CTF to encode proteins, the authors in [13] used the S-kernel function, while we use distance between protein pairs to represent the given protein pairs, and then the RBF kernel function is devoted, that is, a kind of pairwise kernel^[33]

($K(AB, CD) = K(sim(A, B), sim(C, D))$, while $sim(x, y)$ represents the similarity between protein x and y , here $sim(x, y)$ is distance vector) is introduced here. To deal with the imbalance training problem, Shen, et al.^[13] randomly selected the training negative dataset, while we design an automatic procedure to select the training negative dataset. The experiment results show our SVM_{CTF}-SN model can not only successfully predict the known interacting protein pairs but also uncover the potential interacting pairs. Future study can also focus on incorporating a more efficient pairwise kernel into the predictive model.

Efficient feature construction is important in determining the performance of a predictive method. The results in this article suggest that the CTF display its promising prospects. That's because that the CTF considers not only properties of one amino acid but also its vicinal amino acids and treats any three continuous amino acids as an unit, that is, it contains not only the composition of amino acids but also sequence-order effect. While both codon and codon merger mode contain only amino acid composition information. Thus, future work can focus on how to improve feature extracting method, including introducing other encoding features which consider the sequence-order information, such as pseudo amino acid composition (Pse-AAC)^[34] and amphiphilic pseudo-amino acid composition (Am-Pse-AAC)^[35]. Another way to improve the feature construction methods is to integrate more genome and proteome data sources such as gene expression profiling into encoding features. The more information is incorporated into the predictive model, the more better performance will be obtained. In our previous work^[20], we concatenated 14 microarray experiments into SVM_{codon merger}-SN model, the improved predictive performance can be obtained (Figure 1 in [20]). So we believe that if we integrate them by more accurate integrating methods, such as Bayesian model, the performance will be further improved. Besides Bayesian model, we can also use efficient kernel methods to fuse different information^[36]. In addition, we can define a different similarity measure for each data source and thereby incorporate more prior information into the design of the classifier^[37].

There is still plenty room for the improvement on the definition and selection of the gold-standard negative dataset. We note that this is a formidable challenge to our method as well as to any interaction prediction method. Although the non-interacting pairs are well-chosen in our procedure, there have been no evidence to exhibit the confidence for these chosen pairs. Future work can use more efficient methods to select and approximate the gold-standard negative dataset with the help of the functional annotation of proteins. In addition, since the intrinsic reason for constructing the gold-standard negative dataset is the imbalance problem in training SVM. Thus, another way to deal with this problem is to introduce the SVM-based models specially designed for imbalance classification problem, such as SVM with an offset^[38], Twin SVM^[39], Nonparallel plane proximal classifier (NPPC)^[40], an so on.

Acknowledgments

Thank Dr. Yong Wang from Institute of Systems Science, Academy of Mathematics and Systems Science for kind discussion and good suggestions.

References

- [1] J. Wang, S. Zhang, Y. Wang, et al., Disease-aging network reveals significant roles of aging genes in connecting genetic diseases, *PLoS Computational Biology*, 2009, **5**(9): e1000521.
- [2] S. Fields and O. Song, A novel genetic system to detect protein-protein interactions, *Nature*, 1989, **340**: 245–246.
- [3] T. Ito, T. Chiba, R. Ozawa, et al., A comprehensive two-hybrid analysis to explore the yeast protein interactome, *Proceedings of the National Academy of Sciences*, 2001, **98**: 4569–4574.

- [4] A. C. Gavin, M. Boche, R. Krause, et al., Functional organization of the yeast proteome by systematic analysis of protein complexes, *Nature*, 2002, **415**: 141–147.
- [5] Y. Ho, A. Gruhler, A. Heilbut, et al., Systematic identification of protein complexes in *Saccharomyces cerevisiae* by mass spectrometry, *Nature*, 2002, **415**: 180–183.
- [6] H. Zhu, M. Bilgin, R. Bangham, et al., Global analysis of protein activities using proteome chips, *Science*, 2001, **193**: 2101–2105.
- [7] Y. Z. Guo, L. Z. Yu, Z. N. Wen, and M. L. Li, Using support vector machine combined with auto covariance to predict protein-protein interactions from protein sequences, *Nucleic Acids Research*, 2008, **36**: 3025–3030.
- [8] S. Martin, D. Roe, and J. L. Faulon, Predicting protein-protein interactions using signature products, *Bioinformatics*, 2005, **21**: 218–226.
- [9] K. C. Chou and Y. D. Cai, Predicting protein-protein interactions from sequences in a hybridization space, *Journal of Proteome Research*, 2006, **5**: 316–322.
- [10] R. Jansen, H. J. Bussemaker, and M. Gerstein, Revisiting the codon adaptation index from a whole-genome perspective: Analyzing the relationship between gene expression and codon occurrence in yeast using a variety of models, *Nucleic Acids Research*, 2003, **31**: 2242–2251.
- [11] K. A. Dittmar, M. A. Sorensen, J. Elf, et al., Selective charging of *tRNA* isoacceptors induced by amino-acid starvation, *EMBO Reports*, 2005, **6**: 151–157.
- [12] H. S. Najafabadi and R. Salavati, Sequence-based prediction of protein-protein interactions by means of codon usage, *Genome Biology*, 2008, **9**: R87–R95.
- [13] J. W. Shen, J. Zhang, X. M. Luo, et al., Predicting protein-protein interactions based only on sequences information, *Proceedings of the National Academy of Sciences*, 2007, **104**: 4337–4341.
- [14] B. Schölkopf and A. J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*, MIT Press, Cambridge, MA, 2002.
- [15] B. Schölkopf, K. Tsuda, and J. P. Vert, *Kernel Methods in Computational Biology*, MIT Press, Cambridge, MA, 2004, 71–92.
- [16] S. Kerrien, Y. Alam-Faruque, B. Aranda, et al., IntAct-open source resource for molecular interaction data, *Nucleic Acids Research*, 2007, **35**: D561–D565.
- [17] L. Salwinski, C. S. Miller, A. J. Smith, et al., The database of interacting proteins: 2004 update, *Nucleic Acids Research*, 2004, **32**: D449–D451.
- [18] G. D. Bader, I. Donaldson, C. Wolting, et al., BIND: The Biomolecular Interaction Network Database, *Nucleic Acids Research*, 2003, **31**: 248–250.
- [19] G. R. Mishra, M. Suresh, K. Kumaran, et al., Human protein reference database–2006 update, *Nucleic Acids Research*, 2006, **34**: 411–414.
- [20] Y. C. Wang, J. G. Wang, Z. X. Yang, et al., Prediction of protein-protein interaction based only on coding sequences, *Proceedings of the 8th International Symposium on Optimization and Systems Biology*, Zhangjiajie, 2009, 151–158.
- [21] <http://www.genedb.org/>.
- [22] C. W. Hsu, C. C. Chang, and C. J. Lin, A practical guide to Support Vector Classification, 2007, URL: <http://www.csie.ntu.edu.tw/~cjlin>.
- [23] M. Gribskov and N. L. Robinson, Use of receiver operating characteristic (roc) analysis to evaluate sequence matching, *Computers and Chemistry*, 1996, **20**: 25–33.
- [24] J. Platt, Probabilistic outputs for support vector machines and comparison to regularized likelihood methods, *Advances in Large Margin Classifiers*, 1999: 61–74.
- [25] D. J. LaCount, M. Vignali, R. Chettier, et al., A protein interaction network of the malaria parasite *Plasmodium falciparum*, *Nature*, 2005, **10**: 103–107.
- [26] G. D. Bader and C. W. Hogue, Analyzing yeast protein-protein interaction data obtained from different sources, *Nature Biotechnology*, 2002, **20**: 991–997.
- [27] A. Kumar and M. Snyder, Protein complexes take the bait, *Nature*, 2002, **415**: 123–124.
- [28] C. Hertz-Fowler, C. S. Peacock, V. Wood, et al., GeneDB: A resource for prokaryotic and eukaryotic organisms, *Nucleic Acids Research*, 2004, **32**: D339–D343.
- [29] C. Aurrecochea, J. Brestelli, B. P. Brunk, et al., PlasmoDB: A functional genomic database for malaria parasites, *Nucleic Acids Research*, 2009, **37**: D539–D543.

-
- [30] C. Su, J. M. Peregrin-Alvarez, G. Butland, et al., Bacteriome.org—an integrated protein interaction database for *E. coli*, *Nucleic Acids Research*, 2008, **36**: D632–D636.
- [31] I. M. Keseler, C. Bonavides-Martínez, J. Collado-Vides, et al., EcoCyc: A comprehensive view of *Escherichia coli* biology, *Nucleic Acids Research*, 2009, **37**: D464–D470.
- [32] E. Andres Leon, I. Ezkurdia, B. García, et al., EcID. A database for the inference of functional interactions in *E. coli*, *Nucleic Acids Research*, 2009, **37**: D629–D635.
- [33] A. Ben-Hur and W. S. Noble, Kernel methods for predicting protein-protein interactions, *Bioinformatics*, 2005, **21**: i38–i46.
- [34] K. C. Chou, Prediction of protein cellular attributes using pseudo-amino acid composition, *Proteins: Structure, Function, and Genetics*, 2001, **43**: 246–255.
- [35] K. C. Chou, Using amphiphilic pseudo amino acid composition to predict enzyme subfamily classes, *Bioinformatics*, 2005, **21**: 10–19.
- [36] G. R. Lanckriet, M. Deng, N. Cristianini, et al., Kernel-based data fusion and its application to protein function prediction in yeast, *Pacific Symposium on Biocomputing*, 2004.
- [37] Y. Guan, C. Myers, D. Hess, et al., Predicting gene function in a hierarchical context with an ensemble of classifiers, *Genome Biology*, 2008, **9**(S3).
- [38] B. Li, J. Hu, K. Hirasawa, et al., Support vector machine with fuzzy decision-making for real-world data classification, *IEEE World Congress on Computational Intelligence*, Int. Joint Conf. on Neural Networks, Canada, 2006.
- [39] R. Jayadeva Khemchandani and S. Chandra, Twin support vector machines for pattern classification, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2007, **29**: 905–910.
- [40] S. Ghorai, A. Mukherjee, and P. K. Dutta, Nonparallel plane proximal classifier, *Signal Processing*, 2008, **89**: 510–522.