VERSITA

CELLULAR & MOLECULAR BIOLOGY LETTERS
http://www.cmbl.org.pl

Received: 06 October 2010
Final form accepted: 07 March 2011
Published online: 20 March 2011

Volume 16 (2011) pp 264-278
DOI: 10.2478/s11658-011-0008-x

Short communication

PPI_SVM: PREDICTION OF PROTEIN-PROTEIN INTERACTIONS USING MACHINE LEARNING, DOMAIN-DOMAIN AFFINITIES AND FREQUENCY TABLES

PIYALI CHATTERJEE[1], SUBHADIP BASU[2], MAHANTAPAS KUNDU[2], MITA NASIPURI[2] and DARIUSZ PLEWCZYNSKI[3]*

[1]Department of Computer Science and Engineering, Netaji Subhash Engineering College, Garia, Kolkata - 700152, India, [2]Department of Computer Science and Engineering, Jadavpur University, Kolkata - 700032, India, [3]Interdisciplinary Centre for Mathematical and Computational Modelling, University of Warsaw, ul. Pawinskiego 5a, 02-106 Warsaw, Poland

Abstract: Protein-protein interactions (PPI) control most of the biological processes in a living cell. In order to fully understand protein functions, a knowledge of protein-protein interactions is necessary. Prediction of PPI is challenging, especially when the three-dimensional structure of interacting partners is not known. Recently, a novel prediction method was proposed by exploiting physical interactions of constituent domains. We propose here a novel knowledge-based prediction method, namely PPI_SVM, which predicts interactions between two protein sequences by exploiting their domain information. We trained a two-class support vector machine on the benchmarking set of pairs of interacting proteins extracted from the Database of Interacting Proteins (DIP). The method considers all possible combinations of constituent domains between two protein sequences, unlike most of the existing approaches. Moreover, it deals with both single-domain proteins and multi-

* Author for correspondence. e-mail: darman@icm.edu.pl

Abbreviations used: AP – appearance probability; BiFC – biomolecular fluorescence complementation; BIND – Biomolecular Interaction Network Database; DIP – Database of Interacting Proteins; DPI – dual polarization interferometry; FN – false negatives; FP – false positives; FPR – false positive rate; FRET – fluorescence resonance energy transfer; HMMs – hidden Markov models; IgG – Immunoglobulin G; IntAct – open source molecular interaction database; MINT – Molecular Interactions Database; MIPS – Mammalian Protein-Protein Interaction Database; PID – interacting domain pairs; PPI – protein-protein interactions; RBF – radial basis function; ROC – receiver operator curve; SVM – support vector machine; TAP – tandem affinity purification; TN – true negatives; TP – true positives; TPR – true positive rate

domain proteins; therefore it can be applied to the whole proteome in high-throughput studies. Our machine learning classifier, following a brainstorming approach, achieves accuracy of 86%, with specificity of 95%, and sensitivity of 75%, which are better results than most previous methods that sacrifice recall values in order to boost the overall precision. Our method has on average better sensitivity combined with good selectivity on the benchmarking dataset. The PPI_SVM source code, train/test datasets and supplementary files are available freely in the public domain at: http://code.google.com/p/cmater-bioinfo/.

Key words: Protein-protein interaction, Domain-frequency values, Domain-domain interaction affinity value, Proteome, Interactome, Brainstorming, Machine learning, Consensus, DIP, Protein domains, Sequences, Structures, Protein-protein complexes

INTRODUCTION

Understanding protein function is a major goal in the post-genomic era. It has been shown that proteins with similar functions are more likely to interact [1]. If the function of one protein is known, then the function of binding unannotated protein can be assigned. Protein-protein interactions are involved in many biological processes. For example, signals from the exterior of a cell are mediated to the inside of a cell by protein-protein interactions of the biomolecules. The signal transduction process plays a fundamental role in biological processes, and in many diseases (e.g. cancers). There are two types of protein-protein interactions. Proteins might bind to each other and form a stable protein-protein complex, for example in order to be transported from the cytoplasm to the nucleus, or vice versa (nuclear pore importins). On the other hand, a protein may interact only temporarily with another protein, in order to modify it (e.g. a protein kinase adds a phosphate to a target protein). Moreover, posttranslational modifications may change protein-protein interactions, allowing or prohibiting binding to their partners [2]. For example, several proteins with SH2 domains bind to other proteins only when they are phosphorylated on tyrosine residue, while bromodomains specifically recognize only acetylated lysines.

Presently, there are many biochemical and biophysical experimental methods to investigate protein-protein interactions [3-6]. For example, co-immunoprecipitation [3] is the biochemical technique of precipitating a protein antigen out of a solution using an antibody that specifically binds to that particular protein. Biomolecular fluorescence complementation (BiFC) [4] is another method for actually observing the interactions of proteins based on the association of the fluorescent protein fragments that are attached to the components of the same macromolecular complex. Tandem affinity purification [5] is a commonly used technique, which involves creating a fusion protein with a designed TAP tag on the end. The protein of interest with the TAP tag first

binds to beads coated with IgG protein, the TAP tag is then broken apart by an enzyme, and finally a part of the TAP tag binds reversibly to beads of a different type. The protein of interest is then washed through two affinity columns; and it can be examined for binding partners. Among different biophysical methods, dual polarization interferometry (DPI), surface plasmon resonance, and fluorescence resonance energy transfer (FRET) are typically used.

All these experimental techniques have contributed tremendously to the creation of databases containing large sets of protein-protein interaction pairs, such as the Database of Interacting Proteins (DIP) [7], MIPS [8] BIND [9], IntAct [10], MINT [11] and many others. Yet, the high throughput techniques described above are labor intensive and time-consuming, especially when a huge volume of protein and protein-protein interaction data is involved. Therefore, several computational methods have been proposed to first analyze, then automatically predict protein-protein interaction, by exploiting physical and chemical effects on protein binding, or a thermodynamic description of binding kinetics. One of the most profound examples is the protein-protein docking approach for the prediction of protein-protein interactions based on the three-dimensional structures of interacting partners [12-13]. The docking algorithms are slow, so machine learning approaches were proposed in order to speed up the prediction time. Other large-scale computational methods, such as virtual screening techniques, easily deal with many protein-protein interactions within a short time window with acceptable accuracy. Among the most rapid algorithms are those that use the primary sequence of proteins, or their domain structure.

The support vector machine method employed by Bock et al. [14] uses protein primary structure and associated physicochemical properties such as charge, surface tension, and hydrophobicity to predict protein-protein interaction. Their training dataset was obtained from the Database of Interacting Proteins (DIP). Each protein sequence was annotated by a diverse set of features sensitive to local interaction sites, such as surface tension, charge, polar interaction, or sequential hydrophobicity profiles. Gomez et al. [15] used an attraction-repulsion model, where an interaction between a pair of proteins is represented as the sum of attractive and repulsive terms associated with small, domain- or motif-long sequence fragments along a protein chain. The first term is computed using a collection of hidden Markov models (HMMs) for protein domains extracted from the Pfam database. It not only exploits evolutionary conservation of three-dimensional protein structures, but also groups amino acids into four groups depending on their biochemical similarity. The support vector machine (SVM) algorithm is trained, and three-fold cross-validation is performed with ROC score equal to 0.7. Zaki et al. [16] extracted functional, structural or evolutionary relationships between protein sequences to identify inter-domain linker regions, finally detecting domain matches in protein sequences of interest. The main assumption is that two protein sequences may interact if they share similar domains. The overall prediction accuracy achieved by this method is close to 70%, with sensitivity of 0.61 and specificity of 0.7.

The protein-protein interactions can be decomposed into physical interactions between constituent domains of proteins. A domain or motif is defined as a structural and/or functional unit for which a specific sequence signature is conserved in evolution. Therefore, assuming that such motifs mediate the protein-protein interactions, the utilization of domain-domain interaction information is very promising. Wojcik et al. [17] introduced the profile method, which uses evolutionary information about interacting domains. A high quality protein interaction map with information about interacting domains is used to predict protein-protein interactions in other organisms. For example, the *E. coli* protein interaction map is derived from the reference *H. pylori* interactome. Kim et al. [18] proposed the statistical scoring system (PID matrix score) as a measure of the interaction probability (interactability) between different domains. They developed a database of interacting domain pairs (PID), which were extracted from the dataset of experimentally identified interacting protein pairs (DIP). Those pairs were cross-validated with InterPro and a database of protein families, domains and functional sites. The method is able to achieve about 50% sensitivity and 98% specificity.

In other studies [19] the one-class support vector machine algorithm is trained on protein pairs using their constituent domains. The feature vector of each protein includes the number of domains appearing in all the yeast protein. Then the feature vector for a particular protein is prepared by encoding "m" for the domain's indexed position when the protein has m pieces of that domain, otherwise 0. These two vectors for a protein pair are concatenated to form a single feature vector. The support vector machines (SVM) trained on the yeast data achieve accuracy of about 80%. A similar representation was used by Chen et al. [20], although all possible interactions between protein domains were analyzed instead of assuming a single interacting pair of domains for each protein. The forward-pruning decision tree model is able to predict interactions with sensitivity of 79% and specificity of 63%, whereas the multi-layer neural network achieves respectively 77.6% and 66%. In PreSPI [21] a probabilistic framework to predict interaction probability of proteins with a ranking method is developed. Protein pairs that are more likely to interact with each other are distinguished to achieve better accuracy, and the sensitivity is equal to 77%, with specificity of 95% for the DIP dataset. A similar domain approach is used to train a Bayesian kernel [22] with varied threshold, where both domain combinations and their appearance frequencies are obtained from the interacting and non-interacting sets of protein pairs. This information is stored in the appearance probability (AP) matrix, with minimum achieved accuracy equal to 80%, 77.4% sensitivity and 84% specificity, as benchmarked on a protein pair listed in the DIP core.

In summary, most of the existing computational methods consider only domain pairs (a single domain from one protein), and they assume that domain-domain interaction is independent. We propose a novel binary classification method to exploit additionally all possible combinations of domain pairs, therefore

validating and emphasizing their frequencies of co-interactions. The estimation of interaction probability is done by utilizing the individual domain occurrence values and calculated affinity value of domain pairs.

MATERIAL AND METHODS

A novel *PPI_SVM* two-class classification method is proposed here. Each protein pair belongs to either the "interaction" class (i.e. the two proteins interact with each other), or the "non-interaction" class (the two proteins do not interact with each other). Each protein pair is characterized by the list of domains for those two proteins. We used the DIP database (http://dip-mbi.ucla.edu/), which contains experimentally identified pairs of interacting proteins from various organisms, including yeast, *H. pylori*, and *Homo sapiens.* We used 9000 protein pairs from DIP, where almost 4080 unique domain types can be found.

Design of feature set

The likelihood of two proteins interacting depends on their structural composition, homology, etc. It has been found that the possibility of interaction between homologous protein pairs is higher than for those without any homology. This has been taken into account in designing the present feature set without using time-consuming similarity measuring tools such as FASTA, or PSI-BLAST. For that, we have analyzed interacting proteins' domain compositions in order to predict which domains are in contact in the protein-protein complex. In the case of multi-domain proteins we first consider all possible domain pair combinations. Then, we calculate two types of features, which are described in the following subsections: domain frequency and affinity values.

Domain frequency value (D^f)

Whenever a domain appears several times in different interacting protein pairs, we assume that the presence of the domain increases the chance of protein-protein interaction. The *domain frequency value* D_i^f for the domain *i* is computed by counting the occurrence of the *i*-th domain in our training dataset of protein-protein interaction pairs (9000 positive samples), and scaled to be in the range [0,1]. We identified 4080 unique Pfam domains among considered proteins, and indexed them with a number between 1 and 4080. For each protein D_i^f is therefore represented as a 4080 dimensional vector. The *j*-th element in the vector is assigned the frequency values D_j^f; if a protein has a *j* domain, the rest of the elements are set to *0*.

Protein pair interaction affinity value (Affinity)

The more often the domain pairs appear in the interacting protein pairs, the more likely the two domains interact with each other. This observation is calculated using the *protein pair interaction affinity value* (*Affinity)*. We consider all

possible interactions between different constituent domains observed in a particular protein pair. Each domain pair combination is searched among 4500 available positive samples. We compute the occurrence of a particular domain pair (D_i, D_j) in all interacting protein pairs and *Affinity* for two interacting domains, namely D_i, D_j divided by 100, is stored. A higher value of this feature indicates a higher intensity level of protein pair interaction, which normally occurs between homologous protein pairs. For a particular protein pair, there are several combinations of interacting domain pairs, each with corresponding $Affinity_{DiDj}$ values, from which the highest one is taken as the final interaction affinity value for a given protein pair.

Feature representation

First, a protein pair is represented as two vectors of real numbers, each having the dimension of 4080 elements that include the domain frequency values for each individual protein together with their $Affinity_{DiDj}$ data. Therefore, after combining those two vectors, a protein pair is described by 8161 features (see the first part of Fig. 1 for interacting proteins marked by class 1, and the second part of Fig. 1 for non-interacting ones from class 0). For example, if protein P_1 has two domains D_1, D_2, then D_1^f and D_2^f are calculated. If the second interacting protein P_2 has domain D_7, then D_7^f is calculated. Here, both possible domain combinations are analyzed: the first one for D_1, D_7, and another one for D_2, D_7. The maximum of these two affinity values is taken as the final Affinity value between two proteins P_1 and P_2. We assume that the more active domain pair determines the intensity of interaction of the protein pair. In the case of non-interacting proteins, which have a non-interacting domain pair (Fig. 1), the affinity value is set to 0.

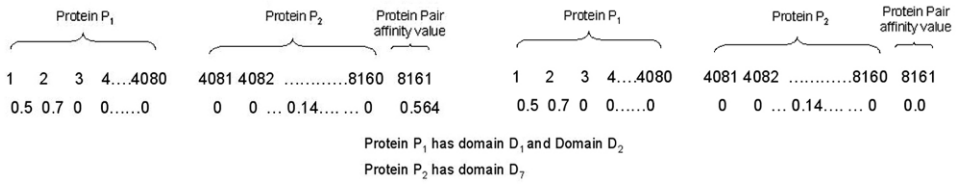

Fig. 1. Feature representation of interacting and non-interacting protein pair.

The support vector machine classifier

The interacting and non-interacting protein pair can be represented as a point in the input feature space as described above. We consider all protein pairs in the training set, divided into two clusters: the first one represents the interacting, and the second one the non-interacting protein pairs. Both clusters are very well separated in the features space; therefore the application of any binary classifier is possible. In the machine learning research area typically SVM is used to find an optimal hyperplane or hypersurface in the binary classification problem

mapped into the high-dimensional features space. The classification of a given protein pair point is predicted depending on which side of the hyperplane (decision surface) it lies. The support vector machine developed by Vapnik [23] is known for its superb generalization abilities with binary classification data. Therefore we use it as a classifier for distinguishing interacting and non-interacting protein pairs. The first set of interacting protein pairs is used as positives for the training algorithm, whereas the non-interacting ones are negatives. The input training examples are nonlinearly mapped into a high-dimensional representations space, and in this space the separating hyperplane that maximizes the margin between the two classes (positives and negatives) is determined. The margin is calculated as the distance of the closest to the hyperplane points from two classes. The hyperplane that minimizes the margin value is proven to separate two classes optimally in comparison to other separating hyperplanes. The support vectors are identified as those data points that lie closest to the decision surface, and therefore that are the most difficult to classify. Given the training set of points $\{x_i, y_i\}_{i=1,2,\ldots p,}$ where x_i represents values of the input feature vector, and y represents the corresponding class label with two value $\bar{y}_i$, the separating hyperplane is represented as a linear combination of the training examples, and classification of an unknown test pattern $\bar{x}$ is done using the following equation (1):

$$f(\bar{x}) = \sum_{i=1}^{1} \alpha_i \, y_i \, k(\bar{x_i}, \bar{y_i}) + b \tag{1}$$

Where $k\,(\bar{x_i}, \bar{y_i})$ is the SVM kernel function, and b the bias that can be optimized on given training data. The optimal hyperplane is found by varying α_i, and the data point x_i corresponding to a given non-zero α_i is defined as the support vector. Finally, the sign of $f(\bar{x})$ function determines the class membership of input query point $\bar{x}$. Apparently, the quality of results depends on the selected kernel function, and therefore helps SVMs to handle nonlinearly separable pattern classes. Typically used kernel functions are polynomials of arbitrary degrees, Gaussian RBFs, etc. In practice, an SVM is implemented as a two-layer feed-forward neural network. Support vector machine (SVM) models are a close cousin of classical multilayer perceptron neural networks. Using a kernel function, SVMs are an alternative training method for polynomial, radial basis function and multi-layer perceptron classifiers in which the weights of the network are found by solving a quadratic programming problem with linear constraints, rather than by solving a non-convex, unconstrained minimization problem as in standard neural network training.

EXPERIMENTAL RESULTS

The protein interaction pairs were obtained from the Database of Interacting Proteins (DIP) [24], which was developed to store and organize information on binary protein-protein interactions from manually compiled and experimentally verified PPI datasets, for example in *Saccharomyces cerevisiae*. The database combines information from a variety of sources in order to create a high-quality dataset for protein-protein interactions of the organism of interest. The data of the DIP database is curated, both manually by expert curators, and by computational approaches that utilize biological knowledge about the protein-protein interaction networks. The most reliable, core subset of the DIP data contains 15 675 interactions of 4749 proteins for which the domain information is available. We have selected almost 9000 protein interaction pairs as our training and test dataset, where unique 4080 Pfam domains are involved. Unfortunately, there is no negative dataset representing non-interacting protein pairs readily available. Therefore, we created 9000 non-interacting protein pairs by exhaustive search and scanning of the 4749 protein pairs. We have in total 18 000 protein pairs, where the ratio of interacting and non-interacting samples is maintained at the ratio of 1:1. The 18 000 protein pairs are divided into 12 subsets (12-fold cross-validation) each containing 1500 positive and negative samples, i.e. 8.33% of the total number of positive/negative samples are used for training and the rest of the samples are used for testing. SVM classifiers are trained with variation of the kernels (linear, polynomial of degree 2, and radial basis function with gamma=.00123), and the Accuracy (%), Recall (Sensitivity) and Precision (Specificity) measures on the test data samples are calculated. We used SVM^{light} code implemented by Joachims et al. [25] (downloaded from http://svmlight.joachims.org web site).

Evaluation metrics

The training results are evaluated using standard measures, such as the *Accuracy*, *Recall*, and *Precision* values, which are explained below. We use a binary SVM classifier with three kernels, namely the linear kernel, polynomial kernel of order 2, and radial basis function kernel with gamma = 0.00123 (i.e. equal to 1.0 divided by the dimension of the feature vector):

$$\text{Accuracy} = (1 - \text{Error}) = \frac{TP+TN}{TP+FP+TN+FN} \quad (2)$$

$$\text{True positive rate / Recall / Sensitivity} \quad \frac{TP}{TP+FN} \quad (3)$$

$$\text{Precision / Specificity} \quad \frac{TP}{TP+FP} \quad (4)$$

$$\text{False positive rate / (1-specificity)} \quad \frac{FP}{FP+TN} \quad (5)$$

where TP is the number of true positives, FP is the number of false positives, TN is the number of true negatives, and FN is the number of false negatives. The recall (R) corresponds to the percentage of correct positive predictions, and the precision (P) measures the percentage of observed positives that are correctly

predicted. The true positive rate (TPR) is described as either the recall or sensitivity measure, and the false positive rate (FPR) estimates the false alarm rate or fall-out values. The performance of SVM models for each type of kernel is described by the recall R and the precision P. The R value measures the percentage of correct predictions, whereas P gives the percentage of observed positives that are correctly predicted.

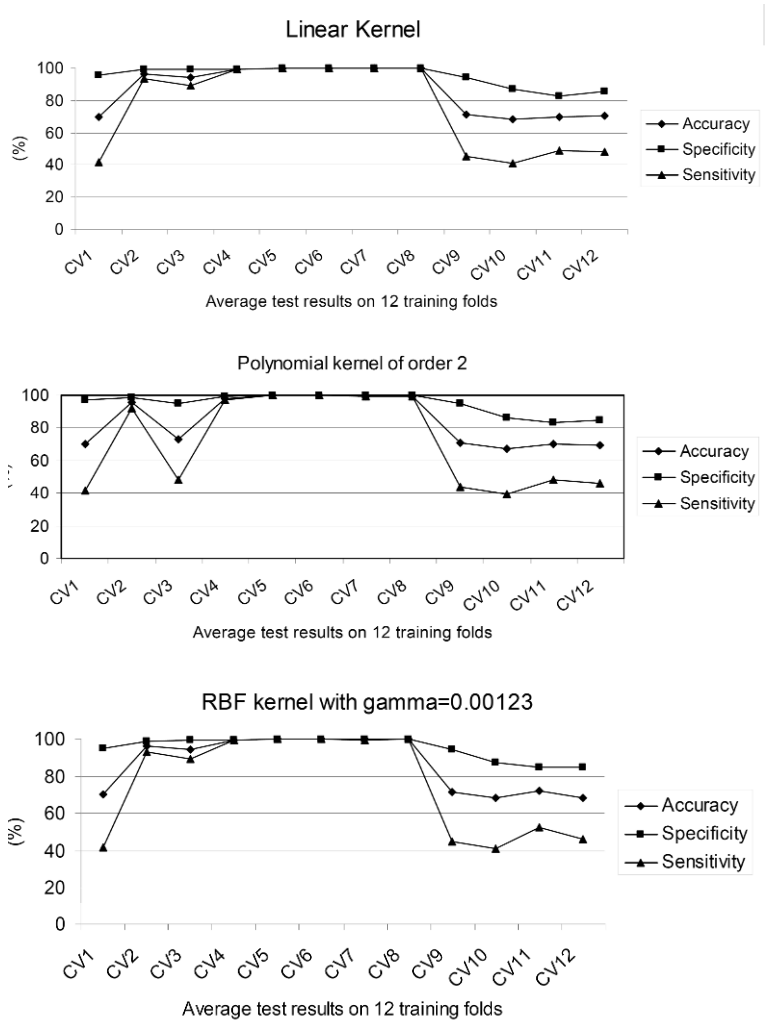

Fig. 2. Performance measures with linear, polynomial and RBF kernel.

Performance analysis

We performed 12-fold cross-validation of results, on protein pairs selected from a subset of the DIP dataset. We selected 9000 protein pairs, with 4500 interacting protein pairs and 4500 non-interacting ones. The dataset is divided

into 12 different folds, where each fold contains 750 interacting pairs as positive items and 750 non-interacting pairs as negative ones. Each dataset is trained with three different learning kernels: linear, polynomial of order 2, and the radial basis function with gamma equal to 0.00123. Each of these 12 subsets is sequentially evaluated 12 times using models trained on the other subsets which are tabulated in detail in the supplementary material: SupplTab. 1. A-M, SupplTab. 2. A-M and SupplTab. 3. A-M at http://dx.doi.org/10.2478/s11658-011-0008-x. *Accuracy*, *Precision* and *Recall* measures for each training and test dataset combination are presented in Fig. 2. The linear kernel for training and test combination numbers (1, 9, 10, 11, 12) achieves accuracies in the range 69-70%, whereas for other combinations it is around 90%. On average the linear kernel classifier achieves accuracy of almost 86%, precision (specificity) of 95.2%, and recall (sensitivity) of 75.7%. The polynomial kernel of order 2, training and test combination numbers (1, 3, 9, 10, 11, and 12) give 69-72% accuracy on test sets, whereas other sets give 90% accuracy. This classifier achieves on average 84.4% accuracy, with precision of 94.9%, and recall measure of 71.2%. Another kernel, namely the radial basis function kernel, for training on (1, 9, 10, 11, and 12) combinations achieves less accuracy than other combinations. Yet, on average it achieves better accuracy than the polynomial kernel, i.e. equal to 86%, with precision of 95.4% and recall of 75.6%.

Discovering potentially interacting domain pair

The domain pairs can be categorized by computing the *Domain-domain interaction affinity value (*$Affinity_{DiDj}$*)* according to their active participation in protein-protein interactions. The interaction probability between two proteins can be supported by the presence of these frequently interacting domains as measured by the $Affinity_{DiDj}$ coefficient. These interacting domain pairs can be categorized into three groups: high, medium, and low probability of interaction. We list in Tab. 1 a selection of high probability interacting domain pairs. We are also able to draw an interaction network for interacting domain pairs. Fig. 3 shows an interaction network for some interacting domains.

Comparison with other existing methods

In Fig. 4 we compare the accuracy of the *PPI_SVM* method with other previously proposed methods that also exploit domain information to predict PPI. Our tool achieves comparable accuracy with PreSPI [21], the domain-based approach by Chen et al. [20], and the Bayesian kernel by Alshawl et al. [22].

PreSPI achieves 77% recall, 95% precision values. The Bayesian kernel approach by Alshawl et al. also reaches almost 77.4% recall, 83.9% precision. Recall and precision reported by the domain-based approach [20] are equal respectively to 79.3% and 62.8%. Our method, namely PPI_SVM, achieves 86% accuracy, with recall/sensitivity of 75.65% and precision/specificity of 95.35%. The PID approach by Kim et al. [8] has values of 50% sensitivity and 98% specificity. Zaki's [6] algorithm has 60% sensitivity, and 70.26% specificity.

The overall accuracy achieved by our method is 86%, which is greater than both the Bayesian kernel method (80%) and PPI prediction by Zaki (70%). Detailed results of the performance measures achieved by this selection of different methods are shown in Fig. 4.

Tab. 1. List of highly active domain pairs

Domain #1	Domain #2
PF00069: Protein kinase domain (IPR000719)	PF00069: Protein kinase domain (IPR000719)
PF00010: DNA-binding domain (IPR001092)	PF00010: DNA-binding domain (IPR001092)
PF00048: Small cytokines (IPR001811)	PF00001:Transmembrane receptor (IPR000276)
PF01423: LSM domain (IPR001163)	PF01423: LSM domain (IPR001163)
PF00001: Transmembrane receptor (IPR000276)	PF00048: Small cytokines (IPR001811)
PF00006: Nucleotide-binding domain (IPR000194)	PF00231: ATP synthase (IPR000131)
PF00306: ATP synthase domain (IPR000793)	PF00231: ATP synthase (IPR000131)
PF00137: ATP synthase subunit C(IPR002379)	PF00231: ATP synthase (IPR000131)
PF00010: DNA-binding domain (IPR001092)	PF07527: Hairy Orange (IPR003650)
PF07527: Hairy Orange (IPR003650)	PF00010: DNA-binding domain (IPR001092)
PF00018: SH3 domain (IPR001452)	PF00018: SH3 domain (IPR001452)
PF00118: SH3 domain (IPR001452)	PF00400: WD domain (IPR001680)
PF00227: Proteasome (IPR001353)	PF00227: Proteasome (IPR001353)
PF07714: Protein tyrosine kinase (IPR001245)	PF00017: SH2 domain (IPR000980)

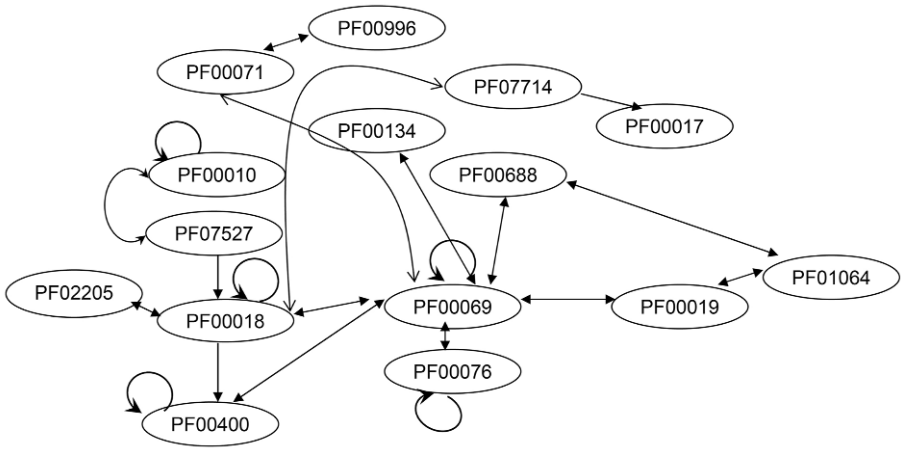

Fig. 3. Domain-domain interaction network.

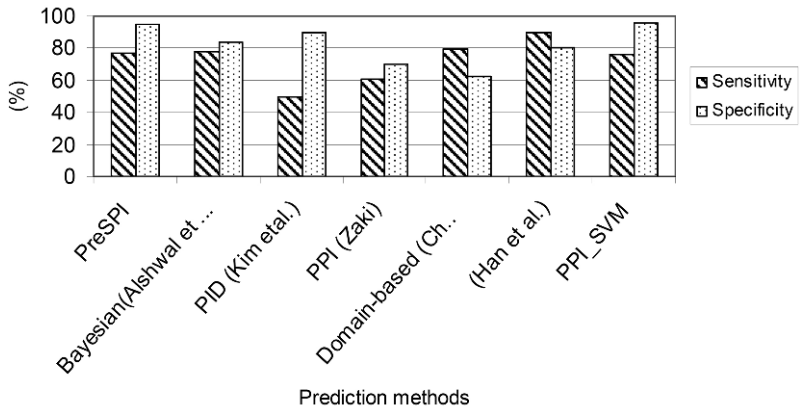

Fig. 4. Comparison of PPI predictive accuracy with other existing methods.

DISCUSSION

Proteins interact with each other through specific intermolecular interactions that are localized to specific structural domains within each protein. So, interaction between protein pairs comprises interactions among their constituent domains. Appearance of a particular domain in interacting protein pairs and participation of interacting domain pairs in PPI are very important in this respect. A protein may consist of multiple domains. For an interacting protein pair, it needs to be investigated which domain pair in what intensity among all possible pairs of them is actively interacting to make the protein pair interact. The novelty of this work lies in the use of two unique features, namely domain frequency, and protein pair interaction affinity values. We consider all possible combinations of constituent domains between two proteins; therefore our method can be applied not only for single-domain proteins, but also multi-domain protein-protein interaction, which is frequently observed in real biological data [26]. Not only considering all possible interactions between constituent domains of interacting protein pairs but also determining interaction affinity between a protein pair by distinguishing the dominant interacting domain pair is also significant in this work. In addition, for binary classification, the choice of support vector machine with different kernels is also important. The support vector machine algorithm with radial-basis function kernel achieves over 86% in accuracy, with precision of 95.35% and recall of 75.65%. The linear kernel reports similar results: accuracy of 86%, precision (specificity) 95.24%, and recall (sensitivity) equal to 75.71%. The polynomial kernel on the other hand provides lower accuracy of 69%, with precision 84.96% and the recall measure 46%. Our method is comparable to the PreSPI method, or other domain-domain interaction methods. PreSPI involves a probabilistic framework to predict the interaction probability of proteins and develops an interaction possibility ranking method for multiple protein pairs. Following the Brainstorming approach [27], we achieved considerable predictive accuracy with the use of these features without using any

probabilistic computation. It dominates over classical tools such as PID, or PPI by Zaki et al. Moreover, we are able to provide a list of highly interacting domain pairs, which is useful to form a domain interaction map to be further used in order to predict PPI [26].

Although the proposed domain combination based prediction method certainly improves the prediction accuracy of the conventional domain based prediction method, it has limitations. Domain-domain interactions are not the only factor in determining all the details of complex protein–protein interactions. In addition, there is no information on the sets of non-interacting pairs. Hence, we artificially created random protein pairing and used it to form a set of non-interacting protein pairs. This could limit the accuracy of prediction, since it might contain some interacting protein pairs that have not yet been discovered. Since we used a subset of the DIP dataset the number of domains in the feature vector is dependent upon the selection of proteins. Since the feature vector of protein is constituted by unique domains (i.e., 4080 unique domains) of selected proteins, PPI_SVM only predicts interaction of those proteins which should have domains within our unique domain sets. Different types of protein features such as solvent accessibility, subcellular localization, and hydrophobicity can be utilized along with domain information to improve its performance [27]. This can be a future work to achieve better predictive accuracy.

Acknowledgements. The authors are thankful to the "Center for Microprocessor Application for Training Education and Research" of the Computer Science & Engineering Department, Jadavpur University, India, for providing infrastructure facilities during progress of the work. P. Chatterjee is thankful to Netaji Subhash Engineering College, Garia for permitting her to carry out research work. Research of S. Basu is partially supported by BOYSCAST Fellowship (SR/BY/E-15/09) from DST, Government of India. Research of D. Plewczynski is supported by the Polish Ministry of Education and Science (N301 159735, N518 409238 and others), and the Interdisciplinary Centre for Mathematical and Computational Modelling (ICM), University of Warsaw (grant no. G36-24).

REFERENCES

1. Ito, T., Tashiro, K., Muta, S., Ozawa, R., Chiba, T., Nishizawa, M., Yamamoto, K., Kuhara, S. and Sakaki, Y. Toward a protein-protein interaction map of the budding yeast: a comprehensive system to examine two-hybrid interactions in all possible combinations between the yeast proteins. **Proc. Natl. Acad. Sci. USA** 97 (2000) 1143-1147.
2. Plewczynski, D. and Basu, S. AMS 3.0: prediction of post-translational modifications. **BMC Bioinformatics** 11 (2010) 210 DOI: 10.1186/1471-2105-11-210.
3. Gharakhanian, E., Takahashi, J., Clever, J. and Kasamatsu, H. In vitro assay for protein-protein interaction: carboxyl-terminal 40 residues of simian virus

40 structural protein VP3 contain a determinant for interaction with VP1. **Proc. Natl. Acad. Sci. USA** 85 (1998) 6607-6611.
4. Hu, C.D., Chinenov, Y. and Kerppola, T.K. Visualization of interactions among bZIP and Rel family proteins in living cells using bimolecular fluorescence complementation. **Mol. Cell.** 9 (2002) 789-798.
5. Rigaut, G., Shevchenko, A., Rutz, B., Wilm, M., Mann, M. and Seraphin, B. A generic protein purification method for protein complex characterization and proteome exploration. **Nat. Biotechnol.** 17 (1999) 1030-1032.
6. Klingström, T. and Plewczynski D. Protein-protein interaction and pathway databases, a graphical review. **Brief. Bioinform.** (2010) DOI: 10.1093/bib/bbq064.
7. Salwinski, L., Miller, C.S., Smith, A.J., Pettit, F.K., Bowie, J.U. and Eisenberg, E. The Database of Interacting Proteins: 2004 update. **Nucleic Acids Res.** 32 (2004) 449-451.
8. Pagel, P., Kovac, S., Oesterheld, M., Brauner, B., Dunger-Kaltenbach, I., Frishman, G., Montrone, C., Mark, P., Stümpflen, V., Mewes, H.W., Ruepp, A. and Frishman, D. The MIPS mammalian protein-protein interaction database. **Bioinformatics** 21 (2005) 832-834.
9. Bader, G.D., Betel, D. and Hogue, C.W. BIND: the Biomolecular Interaction Network Database. **Nucleic Acids Res.** 31 (2003) 248-250.
10. Aranda, B., Achuthan, P., Alam-Faruque, Y., Armean, I., Bridge, A., Derow, C., Feuermann, M., Ghanbarian, A.T., Kerrien, S., Khadake, J., Kerssemakers, J., Leroy, C., Menden, M., Michaut, M., Montecchi-Palazzi, L., Neuhauser, L.N., Orchard, S., Perreau, V., Roechert, B., van Eijk, K. and Hermjakob, H. The IntAct molecular interaction database in 2010. **Nucleic Acids Res.** 38 (2009) 525-531.
11. Ceol, A., Chatr, Aryamontri, A., Licata, L., Peluso, D., Briganti, L., Perfetto, L., Castagnoli, L. and Cesareni, G. MINT, the molecular interaction database: 2009 update. **Nucleic Acids Res.** 38 (2010) 532-539.
12. Plewczynski, D., Łaźniewski, M., Augustyniak, R. and Ginalski, K. Can we trust docking results? Evaluation of seven commonly used programs on PDBbind database. **J. Comput. Chem.** 32 (2011) 742-755.
13. Plewczynski, D., Łaźniewski, M., von Grotthuss, M., Rychlewski, L. and Ginalski, K. VoteDock: Consensus docking method for prediction of protein-ligand interactions. **J. Comput. Chem.** 32 (2011) 568-581.
14. Bock, J.R. and Gough, A.D., A. Predicting protein–protein interactions from primary structure. **Bioinformatics** 17 (2001) 455-460.
15. Gomez, S.M., Noble, W.S. and Rzhetsky, A. Learning to predict protein–protein interactions from protein sequences. **Bioinformatics** 19 (2003) 1875-1881.
16. Zaki, N. Prediction of protein–protein interactions using pairwise alignment and inter-domain linker region. **Engin. Letter** 16 (2008) 505-511.
17. Wojcik, J. and Schachter, V. Protein-protein interaction map inference using interacting domain profile pairs. **Bioinformatics** 17 (2001) 296-305.

18. Kim, W.K., Park, J. and Suh, J.K. Large scale statistical prediction of protein-protein interaction by potentially interacting domain (PID) pair. **Genome Inform.** 13 (2002) 42-50.
19. Alashwal, H., Deris, S. and Othman, R.M. One-class support vector machines for protein-protein interactions prediction. **J. Biomed. Sci.** 1 (2006) 120-127.
20. Chen, X.W. and Liu, M. Domain-based predictive models for protein-protein interaction prediction. **Eurasip Jasp.** 1 (2006) 1-8. DOI: 10.1155/ASP/2006/32767.
21. Han, D.S., Kim, H.S., Jang, W.H., Lee, S.D. and Suh, J.K. PreSPI: a domain combination based prediction system for protein-protein interaction. **Nucleic Acids Res.** 132 (2004) 6312-6320.
22. Alashwal, H., Deris, S. and Othman, R.M. A Bayesian kernel for the Prediction of Protein-Protein Interactions. **World Academy of Science, Engineering and Technology** 51 (2009) 928-933.
23. Vapnik, V. **The nature of statistical learning theory,** Springer-Verlag, New York, 1995.
24. Xenarios, I., Salwinski, L., Duan, X.J., Higney, P., Kim, S.M. and Eisenberg, D. DIP, the Database of Interacting Proteins: a research tool for studying cellular networks of protein interactions. **Nucleic Acids Res.** 30 (2002) 303-305.
25. Joachims, T. Making Large-Scale SVM Learning Practical. in: **Advances in Kernel Methods - Support Vector Learning** (Schölkopf, B.**,** Burges. C. and Smola.A., Eds.), MIT Press Cambridge, 1999, 169-284.
26. Plewczynski, D. and Ginalski, K. The interactome: Predicting the protein-protein interactions in cells. **Cell. Mol. Biol. Lett.** 14 (2009) 1-22.
27. Plewczynski D. Brainstorming: weighted voting prediction of inhibitors for protein targets. **J. Mol. Model.** (2010) DOI 10.1007/s00894-010-0854-x.