Hindawi Publishing Corporation
Journal of Applied Mathematics
Volume 2012, Article ID 897289, 23 pages
doi:10.1155/2012/897289

Research Article

Selecting Negative Samples for PPI Prediction Using Hierarchical Clustering Methodology

J. M. Urquiza,[1] I. Rojas,[1] H. Pomares,[1] J. Herrera,[1] J. P. Florido,[1] and O. Valenzuela[2]

[1] *Department of Computer Architecture and Computer Technology, University of Granada, 18017 Granada, Spain*
[2] *Department of Applied Mathematics, University of Granada, 18017 Granada, Spain*

Correspondence should be addressed to J. M. Urquiza, jurquiza@atc.ugr.es

Received 12 September 2011; Accepted 17 November 2011

Academic Editor: Venky Krishnan

Protein-protein interactions (PPIs) play a crucial role in cellular processes. In the present work, a new approach is proposed to construct a PPI predictor training a support vector machine model through a mutual information filter-wrapper parallel feature selection algorithm and an iterative and hierarchical clustering to select a relevance negative training set. By means of a selected suboptimum set of features, the constructed support vector machine model is able to classify PPIs with high accuracy in any positive and negative datasets.

1. Introduction

Protein-protein interactions (PPIs) play a greatly important role in almost any biological function carried out within the cell [1, 2]. In fact, an enormous effort has already been made to study biological protein networks in order to understand the main cell mechanisms [3–5]. The development of new technologies has improved the experimental techniques for detecting PPIs, such as coimmunoprecipitation (CoIP), yeast two-hybrid (Y2H), or mass spectrometry studies [6–9]. However, computational approaches have been implemented for predicting PPIs because of cost and time requirements associated with the experimental techniques [5].

Therefore, different computational methods have been applied in PPI prediction, some methods are Bayesian approaches [10–12], maximum likelihood estimation (MLE) [13, 14], maximum specificity set cover (MSSC) [4], decision trees [15, 16], and support vector machines (SVM) [15–18]. Many computational approaches use information from diverse sources at different levels [5]. In this way, predicting PPI models [4, 13, 15, 16, 19] have been built using domain information. Since interacting proteins are usually coexpressed and colocated in the same subcellular compartment [10], cell location patterns are also considered to be a valid criterion in prediction works. In other works, authors use functional similarity to predict interacting proteins [20]. Likewise, the concept of homology has been already used to generate prediction models [19, 21], homologs interactions databases [11], and negative datasets [22].

In the past years, these experimental methods [23] and computational approaches [22] have provided interactions for several organisms such as *Saccharomyces cerevisiae* (*S. cerevisiae* or Baker's yeast or simply yeast) [24–27], *Caenorhabditis elegans* (*C. elegans*) [28, 29], *Drosophila melanogaster* (*D. melanogaster* or fruit fly) [30, 31], including *Homo Sapiens* (*H. sapiens*) [3, 6, 32].

In spite of obtaining a huge amount of interaction data through high-throughput technologies, it is still difficult to compare them as they contain a large number of false positives [11, 22]. Some authors provide several reliable interaction sets, including diverse confidence levels. With this context, supervised learning methods used in PPI prediction require complete and reliable datasets formed by positive and negative samples. However, noninteracting pairs are rarely reported by experimentalists motivated by the difficulties associated in demonstrating noninteraction under all possible conditions. In fact, negative datasets have traditionally been created by randomly paired proteins [15, 33, 34] or by selecting pairs of proteins that are not sharing the same subcellular compartment [10]. Nonetheless, other works suggest that negative sets created on the basis of cell location alone lead to biased estimations in the predictive interacting models [17]. To solve this problem, Wu et al. [35] proposed a predictive interacting method by means of similarity semantic measures [36], based on gene ontology (GO) annotations [37], although they did not specify which ontology contributed most to the process of obtaining negative interactions. For this reason, Saeed and Deane [22] introduced a novel method to generate negative datasets, based on functional data, location, expression, and homology. These authors considered noninteracting pairs to be two proteins showing no overlapping between any of the features under consideration. In another work, Yu et al. [38] demonstrated that the accuracy of the PPI prediction works is significantly overestimated. The accuracy reported by the prediction model strongly depends on the structure of the selected training and testing datasets. The chosen negative pairs in the training data have a variable impact on the accuracy, and it can be artificially inflated by a bias towards dominant samples in the positive data [38]. In this way, Yu et al. [38] also presented a method for the selection of unbiased negative examples based on the frequency of the proteins involved in positive interactions in the dataset.

In this work, a novel method is presented for constructing an SVM classifier for PPI prediction, selecting negative dataset through clustering approach applied to 4 million negative pairs from Saeed and Deane [22]. This clustering approach is applied in an effort to avoid the impact of negative dataset on the accuracy of the classifier model. This new method is based on a new feature extraction and selection using well-known databases, applied specifically to a yeast organism model, since yeast is the most widely analysed organism and the one in which it is easiest to find data. New similarity semantic measures calculated from the features are proposed, and they demonstrate that their use improves the predictive power of trained classifiers. In addition, this classifier may return a confidence score for each PPI prediction through a modification of the SVM implementation. Firstly, features are extracted for positive and negative samples; then, a clustering approach is performed in order to obtain high-reliable noninteracting representative samples. Subsequently a parallel filter-wrapper feature technique selects the most relevance extracted features in order to obtain a reliable model. The algorithm called mRMR (minimal-redundancy-maximal-relevance criterion) [39] is used as filter and is based on the statistical concept of mutual information. This reduction in the number of features allows for a better training efficiency as the search space for most of the parameters of the model is also reduced [40, 41].

In a second part, with the purpose of validating the generalisation capability of our model, a group of highly reliable external datasets from [9] were classified using our method. These datasets to be validated were extracted using computational and experimental approaches together with information from the literature. The used models are SVM classifiers built using the most relevance selected features that characterise the protein-protein interaction as explained. They were trained using three training sets, the positive examples were kept, but the negative set was changed, each negative set was obtained by a specific method: (1) hierarchical clustering method presented in this paper, (2) randomly selection, and (3) using the approach proposed by Yu et al. [38].

The testing datasets were filtered for assessment to prevent biased results, that is, without any overlapping between the datasets used during the training stage. High sensitivity and specificity are obtained in both parts using this proposed approach, that is, the model trained using the negative set by the proposed hierarchical clustering method. The presented approach leads to the possibility of becoming a guide for experimentation, being a useful tool to save money and time.

2. Material and Methods

2.1. Material

Two types of datasets were used: training datasets to construct the models and testing datasets to assess the goodness of predictions. A supervised learning classifier as SVM requires positive and negative samples for training data. The positive and negative examples were extracted from Saeed and Deane [22], where authors provide a positive dataset composed of 4809 high-reliability interacting pairs of proteins and a high-quality negative set formed by more than 4 million noninteracting pairs. Two negative subsets of the size similar to that of the positive dataset were extracted from this negative set: one dataset is composed of randomly selected noninteraction pairs (4894) and the other one is created by means of the proposed hierarchical clustering approach presented in this paper in order to select the most representative negative samples (4988). The main goal of this negative dataset of clustered samples is to represent the whole negative space of more than 4 million examples avoiding biased results in PPI prediction. The third negative set used in this paper is created using the method proposed by Yu el at. [38], which is “balanced” to the taken positive set. A comparison of the PPI classification results training three models using these negative datasets is shown Section 3. During the training phase, the positive dataset is called gold standard positive (GSP) set and the used negative dataset is called gold standard negative (GSN) set.

In the case of testing datasets, these were selected for the sake of validating the generalisation capability of the proposed approach in PPI prediction. A group of reliable binary interaction datasets (LC-multiple, binary-GS, Uetz-screen, and Ito-core) were taken from Yu et al. [34]. These datasets have been obtained using several approaches from experimentally, computationally, and grouping datasets well known in the literature. These datasets can be freely downloaded from the website http://interactome.dfci.harvard.edu/. Besides, another group of used negative testing datasets is also described here. So all proposed testing datasets are the follwing.

(i) The LC-Multiple Dataset. It is composed of literature-curated interactions supported by two or more publications. There are 2855 positive interactions.

(ii) Binary-GS dataset. It is a binary gold standard set that was assembled through a computational quality reexamination that includes well-established complexes, as well as conditional interactions and well-documented direct physical interactions in the yeast proteome. There are 1253 positive interactions.

(iii) Uetz-screen. It is the union of sets found by Uetz et al. in a proteome-scale all-by-all screen [24]. There are 671 positive interactions.

(iv) Ito-core. It is Interactions found by Ito et al. that appear three times or more [25]. There are 829 positive interactions.

(v) *Random Negative Dataset 1, 2*. Due to the low number of noninteracting protein data within the RRS set, three negative subsets of similar size of the proposed GSP have been utilised. These set are denoted, random dataset negative 1 (4896 pairs) and random dataset negative 2 (4898 pairs), and were also randomly selected from the Saeed and Deane negative set [22].

(vi) *Negative Datasets Obtained Using the Proposed Hierarchical Clustering Approach*. The negative datasets obtained in the last step of the hierarchical clustering process were used as testing negative datasets. In total there are 9 datasets of 5000 examples (see Section 3).

For all the datasets, a feature extraction process was applied and the data obtained through this process were normalised in the range $[0, 1]$ to apply the proposed method. Furthermore, in a previous step to the evaluation of our model, those interactions from every testing dataset were filtered out to remove overlapping with the training set. In this way, the possible overestimated classification accuracy is prevented through a clustering process selecting a representative negative dataset and a filtering step.

2.2. Feature Extraction

Feature extraction process for the proposed datasets was applied using well-known databases in proteomics, especially for yeast model organism. The calculated features cover different proteomic information integrating diverse databases: Gene Ontology Annotation (GOA) Database [42], MIPS Comprehensive Yeast Genome Database (MIPS CYGD) [43], Homologous Interactions database (HINTdb) [11], 3D Interacting Domains database (3did) [44], and SwissPfam (SwissPfam is an annotated description of how Pfam domains map to possibly multidomain SwissProt entries) [45].

Essentially, the presented approach in this paper integrates distinct protein features to design a reliable classifier of PPIs. The importance of protein domains in predicting PPIs has been already proved [4, 13, 19], so the use of SwissPfam and 3did databases was included in this process. The MIPS CYGD catalogues that cover functional, complexes, phenotype, proteins, and subcellular compartments information about yeast make it a very useful tool in PPI analysis [10, 11]. Likewise, GO data has been successfully applied in classification models [46] and so has the usage of similarity measures supporting PPI prediction [35]. Furthermore, the “interlogs” concept helps to design new approaches in proteomics such as PPI prediction, classification, and creation of reliable PPI databases [11, 22, 28]. Therefore, the HINTdb database was included in our study.

The main step in this process is the extraction of a set of features that can be associated with all possible combinations of pairs of proteins. The fundamental idea about feature extraction here consists of computing how many common terms are shared between two proteins (a given pair) in any given database. Those features would be our "basic" features, with every feature being calculated as the number of common terms that are shared by a pair of proteins in a specific database.

Although the extraction process integrates several information sources, these features in themselves do not provide enough information to estimate whether any given pair of proteins are very likely to interact [10]. Thus, reinforcing the predictive power of classification models through a specific combination of features, two new similarity measures called local and global were incorporated in this process as "extended" features. The definition of these two similarity measures would be the following.

Given a pair of proteins (*protA*, *protB*) and leting A be the set of all terms linked for protein *protA* and B the set of terms linked for protein *protB* in a specific database, the local similarity measure for (*protA*, *protB*) is defined as

$$\mathrm{sim}_{\mathrm{local}} = \frac{\#(A \cap B)}{\#(A \cup B)}, \tag{2.1}$$

where $\#(A \cap B)$ represents the number of common terms in a specific database for (*protA*, *protB*) and $\#(A \cup B)$ is the total number of all terms in the union of sets A and B.

In a similar way, the global similarity measure is calculated as the ratio of common terms shared by a given pair (*protA*, *protB*) with respect to the sum of all terms in a specific database. This measure is calculated as

$$\mathrm{sim}_{\mathrm{global}} = \frac{\#(A \cap B)}{\#C}, \tag{2.2}$$

where C is the total number of terms in the complete database.

Hence, a further description of each considered database detailing the feature calculation and extraction for a given pair of proteins is summarised in Table 4. For the sake of clarity, in the following enumeration, the same information indicating between parenthesis the type of data (integer or real) and the order in the feature list is also explained.

(i) Gene Ontology Annotation (GOA) Database [42] that provides high-quality annotation of gene ontology (GO) [37]. The GO project was developed to give a controlled vocabulary for the annotations of molecular attributes in different model organisms. These annotations are classified in GOA into three structured ontologies that describe molecular function (F), biological process (P), and cellular component (C). Each ontology is organised as a directed acyclic graph (DAG). We extract the IDs (identifiers) of the GO terms associated with each protein calculating the common GO annotation terms between both proteins in the three ontologies (P, C, and F) (1st integer) and their local and global similarity measures (12th real, 13th real). Moreover, we considered each ontology separately (4th P integer, 5th C integer, and 6th F integer) and their respective local (15th real, 16th real, and 17th real) and global similarity measures (18th real, 19th real, and 20th real).

(ii) Homologous Interactions database (HINTdb) [11] is a collection of protein-protein interactions and their homologs in one or more model organisms. Homology refers to any similarity between characteristics that is because of their shared ancestry. The number of homologs between both proteins obtained from HintDB is the 2nd feature (integer).

(iii) MIPS Comprehensive Yeast Genome Database (MIPS CYGD) [43] gathers information on molecular structure and functional network in yeast. All catalogues are considered: functional, complexes, phenotype, proteins, and subcellular compartments. Considering each MIPS catalogue separately, the number of common terms (using the catalogue identifier) is calculated between both proteins (functional 7th integer, complexes 8th integer, proteins 9th integer, phenotypes 10th integer, and subcellular compartments 11th integer). Moreover, their local similarity measures are considered (21st real, 22nd real, 23rd real, 24th real, 25th real).

(iv) 3D Interacting Domains database (3did) [44] is a collection of domain-domain interactions in proteins for which high-resolution three-dimensional structures are known in the Protein Data Bank (PDB) [47]. 3did exploits structural information to support critical molecular details necessary for better understanding how interactions occur. This database also provides an overview of how similar in structure are interactions between different members of the same protein family. The database also stores gene ontology-based functional annotations and interactions between yeast proteins from large-scale interaction discovery analysis. The 3rd feature (integer) is calculated as the common Pfam domains between both proteins, extracted from SwissPfam, which are found in the 3did database. The 3rd feature divided by the total Pfam domains that are associated with both proteins is the 14th feature (real).

(v) SwissPfam [45] from UniProt database [48] is a compilation of domain structures from SWISSPROT and TrEMBL [45] according to Pfam [49]. Pfam is a database of protein families that stores their annotations and multiple sequence alignments created using hidden Markov models (HMM). No feature is directly associated with this database, but it is used in combination with the 3did database to calculate the 3rd and 14th features.

2.3. Feature Selection: Mutual Information and mRMR Criterion

In pattern recognition theory, patterns are represented by a set of variables (features) or measures. Such pattern is a point in an n-dimensional features space. The main goal is to select features that distinguish uniquely between patterns of different classes. Normally, the optimal set of features is unknown and commonly has an irrelevant number or redundant features. In this way, through a pattern recognition process, these irrelevant or redundant features are filtered out greatly improving the learning performance of classifiers [40, 41]. This reduction in the number of features, also known as *feature selection*, allows to simplify the model complexity, as it gives a better visualisation and understanding of used data [50]. In this work, we consider the PPI prediction as a classification problem, so each sample point represents a pair of proteins that must be classified into one out of two possible classes: noninteracting or interacting pair.

The feature selection algorithm can be classified in two groups: filter and wrapper [50, 51]. The filter methods choose a subset of features by means of a preprocessed step independently of used machine learning algorithm. The wrapper methods use the classifier performance to assess the goodness of the features subset. Other authors have utilised a combination of filter and wrapper algorithms [39]; in fact, in this work, a combination between filter and wrapper is used. First, a filter method is applied in order to obtain the relevance of features and subsequently a wrapper method is performed using support vector machine models from the obtained relevance order.

Different criteria have been applied to evaluate the goodness of a feature [50, 52]. In this case, the proposed filter features selection method is based on mutual information as relevance measure and redundancy between the features through minimal-redundancy-maximal-relevance criterion (mRMR) proposed by Peng et al. [39].

Let X and Y be two random continuous variables with marginal pdfs $p(x)$ and $p(y)$, respectively, and joint probability density function (pdf) $p(x, y)$. The mutual information between X and Y can be represented as [50, 53].

$$I(X, Y) = \iint p(x, y) \log \frac{p(x, y)}{p(x)p(y)} dxdy. \tag{2.3}$$

In the case of discrete variables, the integral operation is reduced to a summation operation. Let X and Y be two discrete variables with mathematical alphabets $\mathcal{X}$ and $\mathcal{Y}$, marginal probabilities $p(x)$ and $p(y)$, respectively, and a joint probability mass function $p(x, y)$. The MI between X and Y is expressed as [50]

$$I(X, Y) = \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x, y) \log \frac{p(x, y)}{p(x)p(y)}. \tag{2.4}$$

The mutual information (MI) has two principal properties that make it different from other dependency measures: (1) the capacity of measuring any relationship between variables and (2) its invariance under space transformations [50, 54].

For mRMR, authors considered mutual-information-based feature selection for both discrete and continuous data [39]. The MI for continuous variables was estimated using the Panzer Gaussian windows [39]. Peng et al. show that using a first-order incremental search (as a feature is selected in a time), the mRMR criterion is equal to maximum dependence, or, in other words, estimating the mutual information $I(C, S)$ between class variable C and subset of selected features S. In Peng el al. [39], for minimizing the classification error in the incremental search algorithm, mRMR method is combined with two wrapper schemes. In a first stage, the method is used with the purpose of finding the candidate feature set. In a second stage, backward and forward selections were applied in order to find the compact feature set through the candidate feature set that minimises the classification error.

Given class variable C, the initial set of features F, an individual feature $f_i \in F$, and a subset of selected features $S \subset F$, the mRMR criterion for the first-order incremental search can be expressed as the optimisation of the following condition [39, 50]:

$$I(C; f_i) = \frac{1}{|S|} \sum_{f_s \in S} I(f_s; f_i), \tag{2.5}$$

where $|S|$ is the cardinality of the selected feature set S, $f_s \in S$.

This filter mRMR method is a fast and efficient method because of its incremental nature, showing better feature selection and accuracy in classifier including wrapper approach [39, 50].

In this work, mRMR criterion method was used as filter algorithm with the purpose of obtaining the relevance of proposed features. Subsequently, an SVM model is trained for each incremental combination of features in ascending order of relevance. Such combination of features is applied adding a feature in a time according to the relevance, starting from the most relevant one, and adding the next most relevant one until feature 25. In total, 25 SVM models are trained using grid search to estimate the hyperparameters. A parallel approach was implemented for this filter-wrapper proposal because of memory and computational requirements, reducing the time to obtain the best combination of features that minimises the error classification.

2.4. Support Vector Machine

In machine learning theory, support vector machine (SVM) is related to supervised learning methods that analyse data and recognise patterns in regression analysis and classification problems. In fact, a support vector machine (SVM) is a classification and regression paradigm originally invented by Vladimir Vapnik [55, 56]. In the literature, the SVM is quite popular above all in classification and regression problems mainly due to its good generalisation capability and its good performance [57]. Although SVM method was originally designed for binary-class classification, a multiclass classification methodology was presented in Wu et al. [58]. In the case of this PPI classification, it is straightforward to apply the binary-class classification between interacting and noninteracting pairs of proteins.

For a given training set of instance-label pairs $\{\mathbf{x}_i, y_i\}$, $i = 1, \ldots, N$, with input data $\mathbf{x}_i \in \mathbb{R}^n$ and labelled output data $\mathrm{y}_i \in \{-1, +1\}$, a support vector machine resolves the next optimisation problem:

$$\begin{aligned} &\min_{\mathbf{w},b,\in} \quad \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{N}\xi_i, \\ &\text{subject to} \quad y_i\left(\mathbf{w}^T\phi(x_i) + b\right) \geq 1 - \xi_i, \xi_i \geq 0. \end{aligned} \tag{2.6}$$

So the training data vectors $\mathbf{x}_i$ are mapped into a higher-dimensional space through the ϕ function. C is the hyperparameter called penalty parameter of the error term, that is, it is a real positive constant that controls the amount of misclassification allowed in the model.

Taking the problem given in (2.6) into account, the dual form of an SVM can be obtained

$$\begin{aligned} &\min_{\alpha} \quad \frac{1}{2}\alpha^T Q\alpha - \mathbf{e}^T\alpha, \\ &\text{subject to} \quad y^T\alpha = 0, \\ &0 \leq \alpha_i \leq C, \quad i = 1, \ldots, N, \end{aligned} \tag{2.7}$$

where **e** is a vector composed of all ones (all-ones vector). Q is an N by N positive semi-definite matrix given by $Q_{ij} \equiv y_i y_j K(\mathbf{x}_i, \mathbf{x}_j)$. $K(\mathbf{x}_i, \mathbf{x}_j) \equiv \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j)$ is called the *kernel function* and allows the SVM algorithm to fit a maximum-margin hyperplane in a transformed feature space.

The classifier is a hyperplane in the high-dimensional feature space that may be nonlinear in the original input space. In this case, for the general nonlinear SVM classifier, the decision function can be expressed as

$$y(x) = \operatorname{sign}\left[\sum_{i=1}^{N} \alpha_i y_i K(x, x_i) + b\right], \tag{2.8}$$

where parameters α_i correspond to the solution of the quadratic programming problem that solves the maximum-margin optimisation problem. The training data points corresponding to nonzero α_i values are called *support vectors* [59] because they are the ones that are really required to define the separating hyperplane.

The most common kernel utilised in the literature is the radial basis function (RBF) or the Gaussian kernel [60]. It can be defined as

$$K(x, x_i) = \exp\left(-\gamma \|x - x_i\|^2\right), \quad \gamma > 0, \tag{2.9}$$

where parameter γ controls the region of influence of every support vector.

Training an SVM implies the optimization of the α_i and of the so-called hyperparameters of the model. These hyperparameters are usually calculated using gridsearch and cross-validation [59]. In the case of the RBF kernel, the hyperparameters C and γ are required to be optimised.

Furthermore, a score is proposed in the presented work for PPI prediction. This score is estimated using the difference of probabilities in absolute value returned by SVM model for each pair of proteins.

This score would be used as a measure of confidence in PPI classification. SVM classifies the pairs reporting two probability values that express the chance to belong to an interacting pair class or noninteracting pair class. These probabilities are obtained by the particularisation of the multiclass classification methodology introduced by Wu et al. [58] in the problem of PPI prediction (binary classification). In a general problem, given the observation $\mathbf{x}$ and the class label y, it is assumed that the estimated pairwise class probabilities $\mu_{ij} = P(y = i \mid y = i \text{ or } j, \mathbf{x})$ are available. Following the setting of the one-against-one approach for the general problem of multiclass problem with k classes, firstly, the pairwise class probabilities are estimated by r_{ij} with

$$\mathrm{r}_{ij} \approx P\left(y = i \mid y = i \text{ or } j, \mathbf{x}\right) \approx \frac{1}{1 + e^{A\hat{f} + B}}, \tag{2.10}$$

where A and B are estimated by minimizing the negative log-likelihood function using known training data and $\hat{f}$ are their decision values for these training data. In Zhang et al. [61], it is recalled that SVM decision can be easily clustered at ±1, making the estimate probability in (2.10) inaccurate. Therefore, ten-fold cross-validation was applied to obtain decision values in the experimental results. The next step is obtaining p_i from these r_{ij}, solving the following optimisation problem presented in Wu et al. [58].

The implementation for SVM was taken from the library LIBSVM [62] for Matlab (in this case R2010a). Specifically, C-SVM and RBF kernel was used in the development of the presented work.

2.5. Clustering Methodology

A clustering approach was applied to the negative dataset proposed by Saeed and Deane [22] in order to obtain a relevant, representative, and significant negative subset for training reliable SVM models. Saeed and Deane provide more than 4 million high-quality negative pairs. Therefore, after the feature extraction process applied to this large set of pairs, the set of data to consider would be represented as a matrix whose size is more than 4 million pairs (rows) and 25 features (columns). However, such amount of data is not feasible to train a model, and there is also an overrepresentation of negative data that hides the positive samples effect.

In order to reduce this amount of negative samples to select the most relevant noninteracting pairs, a "hierarchical" clustering approach is proposed in this section which is a iterative k-means process. Due to memory and computational requirements, the clustering data of 4 million noninteracting pairs were divided into subsets which are suitable to be computed by k-means. The k-means algorithm is applied to every subset. For each k-means, the k nearest samples to centroid are taken as the most representative pairs of that specific subset. Then, these representatives are joined again creating a number of new subsets. Thus, the same process of k-means for each subset is applied in an iterative procedure as explained below.

Therefore, in the following lines, a definition of classic k-means is given. In data mining, k-means clustering [63] is a method of cluster analysis that assigns n observations into k clusters where each observation belongs to the cluster with the nearest mean. Given a set of observations $(x_1, x_2, \ldots, x_n)$, where each observation or point is a d-dimensional real vector, n observations are then assigned into k sets $(k \leq n)$ $S = S_1, S_2, \ldots, S_k$ minimising the within-cluster sum of squares (WCSS) [63]:

$$\arg \min_{\mathbf{S}} \sum_{i=1}^{k} \sum_{\mathbf{x}_j \in S_i} \left\| \mathbf{x}_j - \boldsymbol{\mu}_i \right\|^2, \tag{2.11}$$

where μ_i is the mean of points in S_i.

Here, in the application of k-means, the used distance measure is the classical squared Euclidean distance and the clustering data is actually a matrix whose rows represent a pair of noninteracting proteins and columns represent the 25 considered features. The initial cluster centroid positions are randomly chosen from samples. Likewise, k is set to 5000 because it is a value similar to the size of the considered positive set (GSP) and also for computational performance of this "hierarchical" clustering approach.

In practice, the 4 million set was divided in subsets of 50000 pairs approximately (49665 samples) creating 84 subsets of negative samples. This division was carried out due to memory requirements of the available computing system, using the maximum allowed limit. A classical k-means clustering algorithm [63] was applied to each subset obtaining the 5000 most representative samples, that is, reducing 10% of data. Then, new subsets of 50000 negative samples were created adding the 5000 respective samples in order. And again the k-means algorithm is applied to the new subsets obtaining the 5000 most representative samples. This process is repeated until the last 5000 most representative samples that have a similar size to the proposed positive set (see Figure 1) are obtained. This approach is a "hierarchical" and iterative k-means-based clustering algorithm that can be run in a parallel computing platform (see Section 2.6) considering the k-means clustering independently in every iteration.

More formally, if we pay attention to Figure 1, we can see that in Iteration 1, given an initial group of subsets of 50000 pairs approximately $\mathbf{C} = C_1^1, C_2^1, \ldots, C_1^{84}$. As commented, the proposed "hierarchical" clustering approach is an iterative k-means process applied for each C_i^j where i is the iteration and j is the subset order. The resulting set for the k-means method is called R_i^j using the same indices i and j from the input subset C_i^j. Thus, R_i^j is formed by the set of the 5000 most representative negative samples from C_i^j selected by k-means. In the next iterations, C_{i+1}^j is the subset formed by the summation of the 10 sets of the 5000 most representative negative samples R_i^j. When it is not possible to apply the summation of every 10 subsets R_i^j because there is an inferior number of subsets, the summation is composed by the maximum number of subsets until completing all considered data. In general, $C_i^j = \sum_{m=(j-1)*10+1}^{j*10} R_{i-1}^m$ given the iteration i and the subset j. In this paper, 3 iterations were executed until obtaining the set of the 5000 most representative negative samples from the whole set of more than 4 million negative samples. Iteration 2 shows that there were 9 subsets $C_2^1, C_2^2, \ldots, C_2^9$ where C_2^9 contains 20000 pairs. The resulting subsets by k-means $R_2^1, R_2^2, \ldots, R_2^9$ create a new C_3^1 of 45000 elements. In the final step, R_3^1 is obtained in Iteration 3, which will be used as part of a training set as a representation of the negative space from the whole negative set. The $R_2^1, R_2^2, \ldots, R_2^9$ will be used as testing set in Section 3, and after a filtering process from the training set, they are called $R_3^{\text{test\_1}}$, $R_3^{\text{test\_2}}$, $R_2^{\text{test\_3}}$, $R_3^{\text{test\_4}}$, $R_3^{\text{test\_5}}$, $R_3^{\text{test\_6}}$, $R_3^{\text{test\_7}}$, $R_3^{\text{test\_8}}$, and $R_3^{\text{test\_9}}$.

With this process, the main goal of obtaining a representative negative dataset and not biased from a high-quality negative set is fulfilled.

2.6. Parallelism Approach

The filter/wrapper feature selection proposed in this work demands high computational resources. The classical and simple master-slave approach was adopted [64], a master process sends tasks and data to the slave process, and the master process receives results from slaves and controls the finalisation of the tasks. In our case, the tasks are to train SVM model including grid search for hyperparameters. Therefore, the master process sends the next data for slave processes: the selected features and the training and testing datasets. In addition, the "hierarchical" k-means clustering algorithm from the previous section could be implemented in a parallel computing platform using this approach.

The implementation of this approach was carried out using MPIMEX [65], a new interface that allows MATLAB standalone applications to call MPI (message passing interface) standard routines (it was developed in our research group). MPI is a library specification for message passing, proposed as a standard by a broadly based committee of vendors, implementers, and users as defined in http://www.mcs.anl.gov/research/projects/mpi/.

This parallel approach was running in a cluster of computers. This cluster was formed by 13 nodes with dual processors Intel Xeon E5320 2.86 GHz, 4 GB RAM memory, and 250 GB HDD. All nodes are connected using Gigabit Ethernet. The installed operating system is

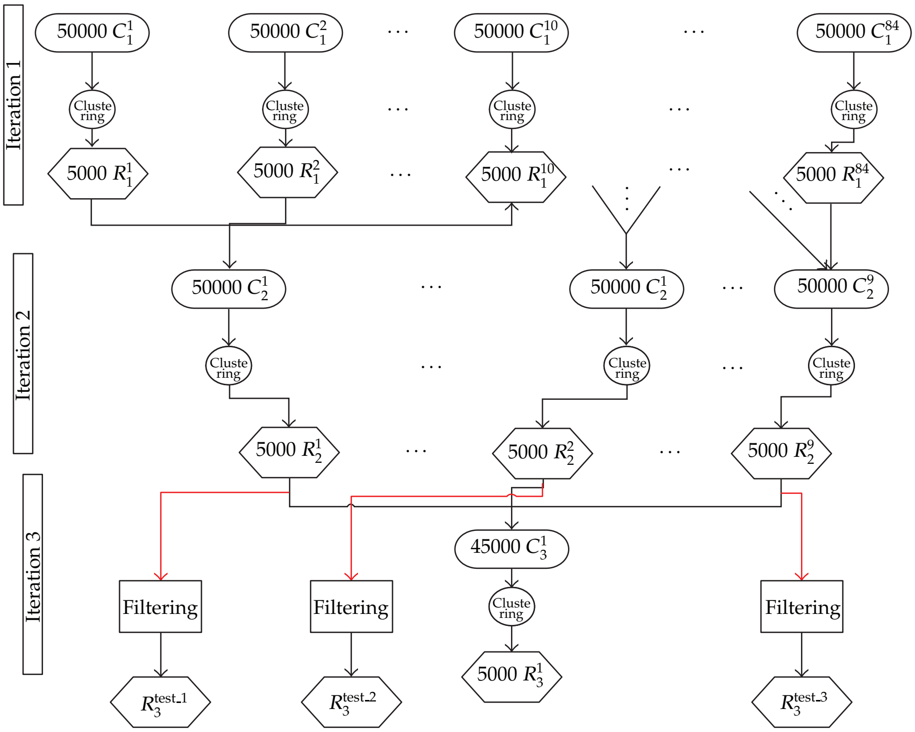

Figure 1: Diagram for the proposed "hierarchical" k-means-based clustering algorithm applied. It is an iterative k-means process. The application in this problem would be the selection of the 5000 most representative negative samples of the whole set.

Linux CentOS 4.6 (rocks). This cluster was purchased using public funds from the Spanish Ministry of Education Project TIN 2007-60587. The time of execution was reduced from 16 days in a single computer to 32 hours to train all the SVM models.

3. Results and Discussion

The results consist of two parts. In the first part, a "suboptimal" set of features is selected through the filter/wrapper feature selection process using the parallel approach. The training data for RBF-SVM model is composed by a GSP set and for a GSN set which is the set which resulted from applying iterative clustering approach as explained in section Material and Methods. In the second part, taking this suboptimal set of features, three RBF-SVM classifiers are constructed using three training sets, respectively. All training sets have the same GSP set for positive examples. In one case, the GSN set is the negative set obtained using the hierarchical clustering method from the first part and, in a second case, the GSN set is a randomly selected negative set as commented. In the third case, the GSN set was created using the approach proposed by Yu el al. [38], it is a "balanced" set to GSP. Subsequently, a comparison of the results obtained of three RBF-SVM classifiers trained with all the proposed negative datasets is discussed.

Previously the filter/wrapper feature selection process, the feature extraction process is applied to all available datasets. The 25 features were also extracted for the 4 million negative set from Saeed and Deane [22], but, due to computational requirements, the whole set was divided into 84 subsets of 50000 samples approximately. In order to obtain a representative negative dataset of the whole negative space, the iterative k-means clustering approach was applied to these 84 subsets as explained in the Section 2.5. In total, three iterations selecting 5000 negative representative samples were realized using the clustering approach. In the first iteration, the Euclidean k-means method was applied to the 84 subsets creating 5000 centroids, and 9 new subsets (8 subsets of 50000 and the last one of 20000 negative examples) were obtained adding the selected 5000 negative representative samples of each previous subset. In the second iteration, the k-means was applied again to the 9 new subsets taking 5000 new negative representative samples of each subset and creating another new subset of 45000 samples (the representatives of 9-subset summation). In the third and last iteration, the last 5000 most representative negative samples taken as GSN set for training data were obtained from clustering the previous subset. The taken negative pairs were selected using the minimum Euclidean distance to the centroid of each cluster. A diagram of this process is represented in Figure 1.

In this way, the considered data (GSP and clustered GSN sets) was used to apply the presented paralleled filter/wrapper feature selection process. Because of memory requirements in the construction of the 25 SVM models, this data was randomly divided into 70% for training SVM and 30% for testing the performance of obtained models. Hence, four randomly divisions of data as 4 training/test datasets were used in this feature selection approach in a cluster of computers as commented in Section 2.6. In order to obtain the best hyperparameters for SVM models, gridsearch and 10-fold cross-validation were implemented. In Figure 2, the accuracy, sensitivity, and specificity obtained using the order of feature relevance reported by mRMR filter method are shown for all 25 SVM models. It can be observed that an excess of information may lead to overfitting, that is, the interaction information decreases when adding more features to the models, specially for testing case. The last added features were considered for mRMR method as more irrelevant or redundant than the features in the first positions. In Figure 2, it can be observed that the performance does not significantly improve after reaching 6 features, it even gets worse due to an excess of information, so the suboptimum selected set is composed of those 6 features: 13th referring to *global similarity measure for* 1st *feature, common GO terms using all ontologies*, 3rd referring to *number of SwissPfam domains for a pair in 3did*, 10th referring to *common terms for the two proteins in MIPS phenotype catalogue*, 8th referring to *common terms for the two proteins in MIPS functional catalogue*, 7th referring to *common terms for the two proteins in MIPS complexes catalogue,* and 2nd referring to *number of shared homologous proteins between a pair of proteins*.

In the selected suboptimum set, the features concerning protein complex, phenotypes, and functional data from MIPS CYGD catalogues have already been used successfully and proved themselves to be reliable in interacting prediction analysis [10, 35, 66–69]. Note that global similarity measures were also included in this suboptimum set of features with the purpose of improving the performance of the classifier in PPI prediction. At the same time, domain information (3rd feature) has provided a suitable framework in PPI prediction works [4, 13]. Moreover, the second feature refers to homology whose relevance has been shown in previous publications [11, 19, 21, 22].

In order to check if the SVM models trained with 6 features are significant, a ROC (receiver operating characteristic) was plotted using the confidence score presented in this work, previously explained in Section 2. The ROC curve shows the sensitivity values with respect to 1-specificity values. The used statistics to measure the goodness of the classification was the widely extended AUC (area under curve) [70, 71]. This statistics represents the probability

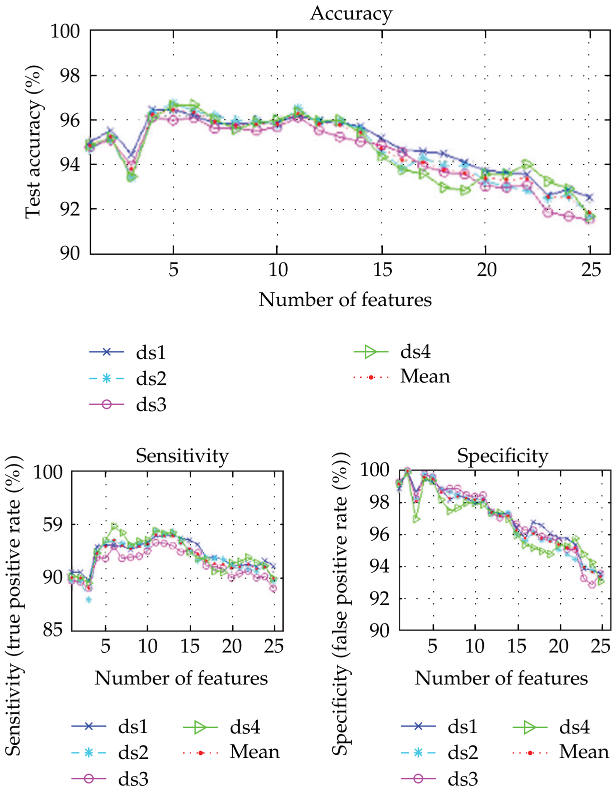

Figure 2: Sensitivity, specificity, and test accuracy for the four randomly partitioned datasets and their average values.

Table 1: Results for ROC: Area Under Curve (AUC).

Training and test group	6-feature SVM	25-feature SVM
1st	0.808	0.672
2nd	0.812	0.725
3rd	0.836	0.698
4th	0.846	0.619
Mean	0.826	0.678
Std. deviation	0.016	0.039

The ROC curve was constructed using our proposed confidence score for the four randomised sets (70% training, 30% test). The RBF kernel SVMs were trained using 6 features and 25 features. Std. standard.

that a classifier will rank a randomly chosen positive instance higher than a randomly chosen negative one. In Figure 3 and Table 1, the results for 6-feature SVM model and 25-feature SVM model showing better performance of the SVM trained with a suboptimum set are shown. As we mentioned, this reduction in the number of features implies a significant saving in memory, calculation, and other computational requirements, obtaining an overfitting utilising the whole set.

In the second part, the behaviour of our approach is tested using the selected subset of the six most relevant characteristics. Three RBF-SVM models are built with three training sets, sharing the same GSP but with a different GSN. In one case, the GSN is the negative set from

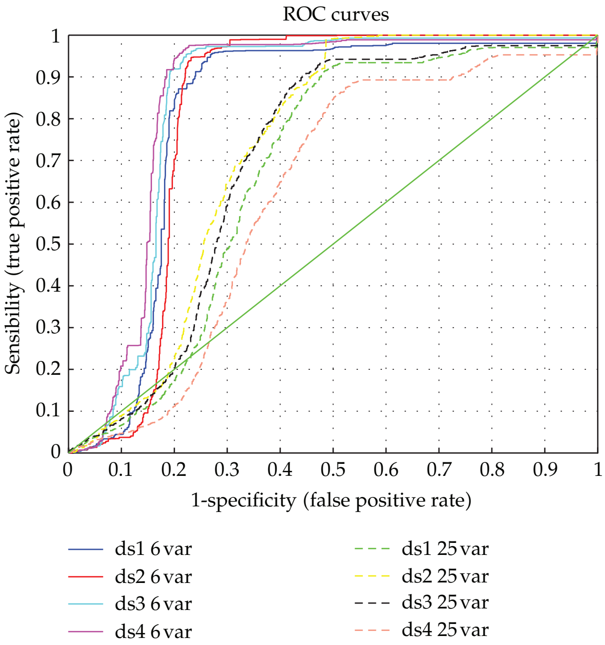

Figure 3: ROC curve for the four randomly partitioned groups (6 and 26 features).

the first part created using the proposed hierarchical clustering approach presented in this paper (it is also called clustered training dataset). In the second case, the GSN is a randomly selected negative set (called random training dataset), and, in the last case, the GSN is a negative set "balanced" to GSP set obtained using the approach by Yu et al. [38]. This third GSN is created using a selection of unbiased negative examples based on the frequency of the proteins in the positive set. The testing datasets, detailed in Section 2, cover both positive and negative sets and they were obtained in different ways: experimentally, from the literature, and computationally. Additionally, in order to make a reliable comparison, previous to the evaluation of our models, the interactions for each testing dataset were filtered out to avoid overlapping with the respective training set. The new sizes of the testing datasets are shown in Table 2.

Therefore, the results of these models are shown in Table 3 and Figure 4 for positive datasets and Figure 5 for negative datasets. In general, the SVM model trained using the negative set generated by the proposed hierarchical clustering approach presented in this paper has a better performance in comparison with the rest of models, that is, the models that used the randomly selected negative set and the balanced negative set. Globally, the obtained results were slightly worse in the experimental datasets than in the computational and literature datasets. The models classify the literature-extracted dataset "LC-multiple" with a range between 93 and 95% of accuracy. For the computationally obtained "binary-GS" dataset, the classifiers obtain a range of accuracy between 92 and 95%. Between the experimental datasets "Uetz-screen" [24] and "Ito-core" [25], the reported accuracies are sightly lower than for the previous datasets with ranges of 72–81% and 76–80%, respectively, for the case of the models trained with the negative set from the clustering approach and the negative set from the random selection. Nevertheless, in the case of the model trained using the "balanced" negative set, the accuracies for both datasets are about 50%. However, if we can consider the nature

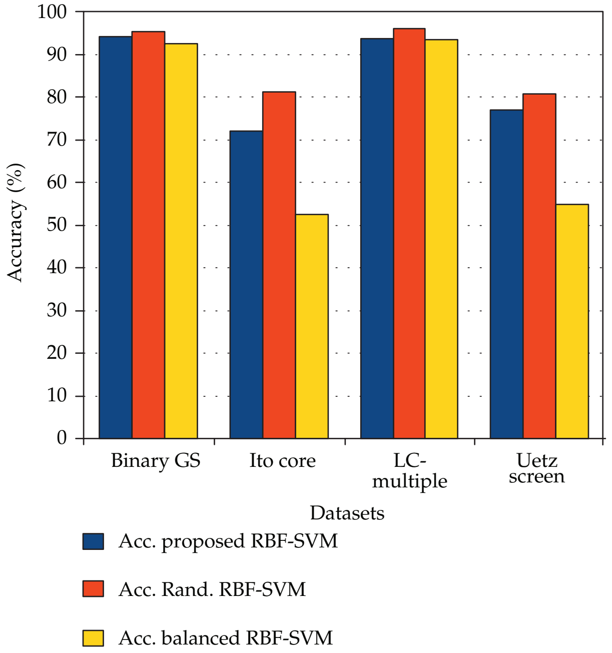

Figure 4: Comparison of accuracy obtained in positive datasets for the three trained models: the SVM model trained using the training set formed by the GSP set and the GSN set obtained using the proposed hierarchical clustering method (clustered), the SVM model trained using the training set where the GSN set was randomly selected (Rand. RBF-SVM) and the balanced RBF-SVM is the SVM model trained using the training set formed by the GSP set and the GSN set obtained using the approach to create a "balanced" negative set by Yu et al. [38].

Table 2: New sizes of datasets after filtering process.

Datasets	Size of filtering training set with GSN set obtained using the presented hierarchical clustering	Size of filtering training set with randomly selected GSN set	Size of filtering training set with "balanced" GSN set obtained from the approach by Yu et al. [38]
Binary-GS	933	937	987
Ito-core	680	686	700
LC-multiple	2362	2380	2468
Uetz-screen	574	584	594
Random negative dataset 1	4893	4894	4894
Random negative dataset 2	4895	4894	4898
$R_3^{\text{test\_1}}$	4735	4995	4992
$R_3^{\text{test\_2}}$	4788	4995	4994
$R_3^{\text{test\_3}}$	4814	4991	4991
$R_3^{\text{test\_4}}$	4844	4987	4992
$R_3^{\text{test\_5}}$	4854	4983	4986
$R_3^{\text{test\_6}}$	4816	4991	4994
$R_3^{\text{test\_7}}$	4837	4985	4990
$R_3^{\text{test\_8}}$	4797	4994	4994
$R_3^{\text{test\_9}}$	4873	4994	4996

Table 3: Accuracy using the 6 most relevant features for three RBF-SVM models.

Datasets	Acc. Our proposal RBF-SVM	Acc. Rand. RBF-SVM	Acc. "balanced" RBF-SVM	% relative difference for our proposal versus "Rand" model	% relative difference for our proposal versus "balanced" model
Binary-GS	94,111	95,411	92,401	−1,381	1,817
Ito-core	72,059	81,195	52,571	−12,678	27,045
LC-multiple	93,750	95,924	93,517	−2,319	0,249
Uetz-screen	76,857	80,822	54,882	−5,159	28,592
Random negative dataset 1	72,211	38,353	6,537	46,888	90,947
Random negative dataset 2	71,951	37,937	37,444	47,274	47,959
$R_3^{\text{test\_1}}$	58,184	29,349	1,883	49,558	96,764
$R_3^{\text{test\_2}}$	63,596	30,150	1,882	52,591	97,041
$R_3^{\text{test\_3}}$	96,469	69,365	1,683	28,096	98,255
$R_3^{\text{test\_4}}$	62,221	31,061	1,522	50,080	97,554
$R_3^{\text{test\_5}}$	61,248	29,862	1,364	51,244	97,773
$R_3^{\text{test\_6}}$	64,992	33,120	1,702	49,040	97,381
$R_3^{\text{test\_7}}$	64,441	31,454	1,824	51,189	97,170
$R_3^{\text{test\_8}}$	94,705	67,821	1,702	28,387	98,203
$R_3^{\text{test\_9}}$	64,334	31,237	1,061	51,446	98,351

Acc. is the accuracy of the RBF SVM model. *Our proposal RBF-SVM* is the SVM model trained using the training set formed by the GSP set and the GSN set obtained using the proposed hierarchical clustering method. *Rand. RBF-SVM* is the SVM model trained using the training set where the GSN set was randomly selected. *"balanced" RBF-SVM* is the SVM model trained using the training set formed by the GSP set and the GSN set obtained using the approach to create a "balanced" negative set by Yu et al. [38]. % relative difference is the percentage of relative difference using "our proposal RBF-SVM" as basis.

and complexity of the filtering in experimental data, the obtained accuracy is still satisfactory at least in the case of the model trained using the negative set from the clustering approach. Referring to the different negative datasets in the training data, the model trained using the negative set extracted by clustering method attained better results than the model trained using a randomly selected negative set. The obtained minimum relative difference is about 28% compared to the randomly selected negative set, and the maximum difference is about 90% in the case of the model trained using the "balanced" negative set. The negative set obtained by the "hierarchical" approach has a relevant representation of the negative search space from a large high-reliability negative set from Saeed and Deane [22]. But in the case of the "balanced" negative set, this is not happening, the negative set is "balanced" to the positive side in the training data but it is not enough to recognise any negative case. Hence, the obtained results of the model predicting negative datasets are worse than the results in the classification of positive datasets. Nonetheless, the difficulty and complexity to predict negatives make the results still acceptable. It can be observed that the relative difference in positive datasets is better for the model trained with the randomly selected negative set but that difference is not so strong, it can even be a slightly overestimation. The accuracy could be artificially inflated by a bias towards dominant samples in the positive data as Yu et al. showed [38]. With such a suboptimum set of features, an SVM model is able to classify PPIs with relative notorious accuracy in any positive and negative datasets.

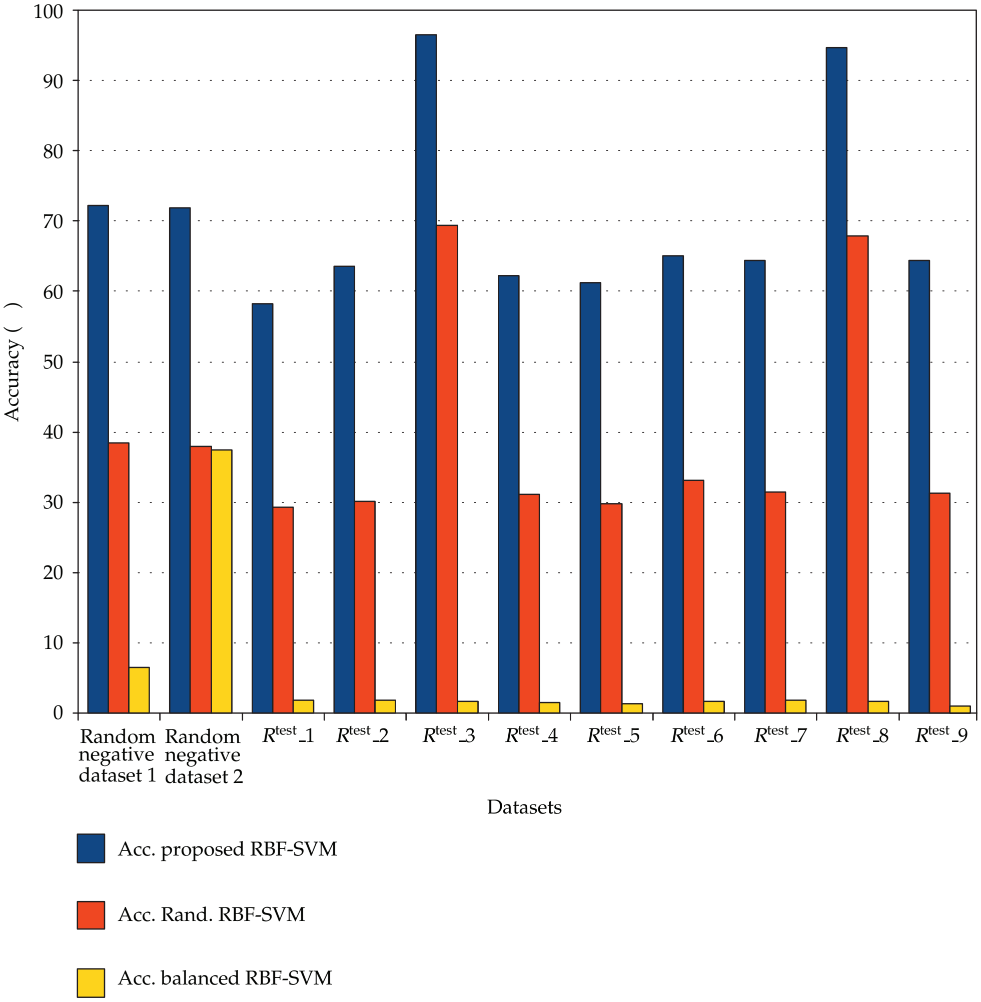

Figure 5: Comparison of accuracy obtained in negative datasets for the two trained models: the SVM model trained using the training set formed by the GSP set and the GSN set obtained using the proposed hierarchical clustering method (clustered) and the SVM model trained using the training set where the GSN set was randomly selected (Rand. RBF-SVM) and the balanced RBF-SVM is the SVM model trained using the training set formed by the GSP set and the GSN set obtained using the approach to create a "balanced" negative. Please note that R^{test}_1, R^{test}_2, R^{test}_3, R^{test}_4, R^{test}_5, R^{test}_6, R^{test}_7, R^{test}_8, and R^{test}_9 correspond to: $R_3^{\text{test\_1}}$, $R_3^{\text{test\_2}}$, $R_2^{\text{test\_3}}$, $R_3^{\text{test\_4}}$, $R_3^{\text{test\_5}}$, $R_3^{\text{test\_6}}$, $R_3^{\text{test\_7}}$, $R_3^{\text{test\_8}}$, and $R_3^{\text{test\_9}}$. And the "balanced" negative set is created using the approach by Yu et al. [38].

First, in Patil and Nakamura [19], the authors used a Bayesian approach, previously proposed by Jansen et al. [10] with only three features for the filtering out of high-throughput datasets of the organisms *Saccharomyces cerevisiae* (Yeast), *Caenorhabditis elegans*, *Drosophila melanogaster*, and *Homo sapiens*. Their model was able to obtain a sensibility of 89.7% and a specificity of 62.9%, being only capable of attaining a prediction accuracy of 56.3% for true interactions for the datasets Y2H, external to the model. For two datasets called "Ito" and "Uetz" (see Table 3), the presented model trained with the negative set from clustering method reported classification rates between 76 and 93%. In Jiang and Keating [72], a mixed framework is proposed combining high-quality data filtering with decision trees in PPI prediction,

Table 4: Description of the 25 extracted features.

Number	Description	Type
1st	$\#(A_{GOA} \cap B_{GOA})$ from GOA DB taking 3 ontologies together (P,F,C)	Integer
2nd	Number of homologs for $(protA, ProtB)$ from HINTdb	integer
3rd	$\#[(A_{SPFAM} \cap 3DID) + (B_{SPFAM} \cap 3DID)]$, A and B are domains extracted form SwissPfam, 3DID is 3did database	Integer
4th	$\#(A_{GOA-P} \cap B_{GOA-P})$ from GOA DB taking Biological Process ontology	Integer
5th	$\#(A_{GOA-C} \cap B_{GOA-C})$ from GOA DB taking Cellular Compartment ontology	integer
6th	$\#(A_{GOA-F} \cap B_{GOA-F})$ from GOA DB taking Molecular Function ontology	integer
7th	$\#(A_{MIPS-F} \cap B_{MIPS-F})$ from functional MIPS catalogue identifiers	integer
8th	$\#(A_{MIPS-C} \cap B_{MPIS-C})$ from complexes MIPS catalogue identifiers	integer
9th	$\#(A_{MIPS-P} \cap B_{MIPS-P})$ from proteins MIPS catalogue identifiers	integer
10th	$\#(A_{MPIS-FE} \cap B_{MPIS-FE})$ from phenotypes MIPS catalogue identifiers	integer
11th	$\#(A_{MPIS-FCC} \cap B_{MIPS-FCC})$ from subcellular compartments MIPS catalogue identifiers	integer
12th	Local similarity of 1st feature	real
13th	Global similarity of 1st feature	real
14th	$\#[((A_{SPFAM} \cap 3DID) + (B_{SPFAM} \cap 3DID))]/\#(A_{SPFAM} \cup B_{SPFAM})$	Real
15th	Local similarity of 4th feature	real
16th	Local similarity of 5th feature	real
17th	Local similarity of 6th feature	real
18th	Global similarity of 4th feature	real
19th	Global similarity of 5th feature	Real
20th	Global similarity of 6th feature	Real
21th	Local similarity of 7th feature	Real
22th	Local similarity of 8th feature	Real
23th	Local similarity of 9th feature	Real
24th	Local similarity of 10th feature	Real
25th	Local similarity of 11th feature	Real

Symbol # indicates the number of elements in a set. See (2.1) and (2.2).

taking as the base the notation of all GO ontologies, aiming an accuracy in a range of 65–78%. From there, we incorporated that information in combination with other features to improve the generalisation of our approach. Other similarity measures have been proposed, mainly based on the GO annotations, for example, the works by Wu et al. [35] that were able to detect the 35% of the cellular complexes from the MIPS CYGD catalogues or the work by Wang et al. [36] for the validation of gene expression analysis. Nevertheless, the authors did not take into account the cellular component ontology because it was considered that this ontology includes ambiguous annotations that may lead to error. In this paper, we opted for proposing a set of similarity measures that permit their generalisation to a wide range of databases in the obtaining of our prediction model.

4. Conclusion

In this work, a new approach to build an SVM classifier in PPI prediction is presented. The approach has several notorious processes: a feature extraction using well-known databases, a new filter-wrapper feature selection implemented in a master-slave parallel approach, and a reliable and representative negative dataset for training by the means of "hierarchical" k-means clustering. The filter method is based on the statistical concept of mutual information using mRMR criterion, which is a reliable and quick method. In addition, a confidence score is presented through a modification of SVM model implementation. A comparison between a randomly selected negative dataset, a "balanced" negative set obtained using Yu et al. approach [38], and a negative dataset obtained using the "hierarchical" k-means clustering method presented in this paper is done where the model training using the set resulted by the clustering approach has better performance. This comparison also allowed us to check the generalisation capacity of the presented approach for the sake of the evaluation of previously filtered external datasets. Hence, a fair negative selection method is presented avoiding the overestimation in the classification of PPIs.

For further work, a hierarchical parallel clustering could improve the performance of a classifier with the purpose of obtaining a balanced negative dataset using a more complex clustering algorithm. We consider applying this approach to other model organisms as *Homo sapiens*. A parallel approach was applied, which, by making a better load balancing, would be suitable to reduce time computation in the filter/wrapper feature selection approach.

In summary, we conclude that by combining data from several databases, using reliable positive and clustered negative samples for training, supporting a set of widely applicable similarity measures to the feature extraction process, and using mutual information methods for feature selection and RBF-SVM models capable of returning a confidence score, we have presented a reliable approach to the validation of protein-protein interaction datasets.

Acknowledgments

J. M. Urquiza is supported by the FPU Research Grant AP2006-01748 from the Spanish Ministry of Education. This paper has been partially supported by the Spanish CICYT Project SAF2010-20558 and by the Regional Excellence Projects P07-TIC-02768 and P09-TIC-175476.

References

[1] V. Deshmukh, C. Cannings, and A. Thomas, "Estimating the parameters of a model for protein-protein interaction graphs," *Mathematical Medicine and Biology*, vol. 23, no. 4, pp. 279–295, 2006.

[2] D. J. Higham, G. Kalna, and J. K. Vass, "Spectral analysis of two-signed microarray expression data," *Mathematical Medicine and Biology*, vol. 24, no. 2, pp. 131–148, 2007.

[3] J. F. Rual, K. Venkatesan, T. Hao et al., "Towards a proteome-scale map of the human protein-protein interaction network," *Nature*, vol. 437, no. 7062, pp. 1173–1178, 2005.

[4] C. Huang, F. Morcos, S. P. Kanaan, S. Wuchty, D. Z. Chen, and J. A. Izaguirre, "Predicting protein-protein interactions from protein domains using a set cover approach," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 4, no. 1, pp. 78–87, 2007.

[5] A. J. González and L. Liao, "Predicting domain-domain interaction based on domain profiles with feature selection and support vector machines," *BMC Bioinformatics*, vol. 11, article 537, 2010.

[6] U. Stelzl, U. Worm, M. Lalowski et al., "A human protein-protein interaction network: a resource for annotating the proteome," *Cell*, vol. 122, no. 6, pp. 957–968, 2005.

[7] K. Venkatesan, J. F. Rual, A. Vazquez et al., "An empirical framework for binary interactome mapping," *Nature Methods*, vol. 6, no. 1, pp. 83–90, 2009.

[8] P. Braun, M. Tasan, M. Dreze et al., "An experimentally derived confidence score for binary protein-protein interactions," *Nature Methods*, vol. 6, no. 1, pp. 91–97, 2009.

[9] J. Yu and R. L. Finley, "Combining multiple positive training sets to generate confidence scores for protein-protein interactions," *Bioinformatics*, vol. 25, no. 1, pp. 105–111, 2009.

[10] R. Jansen, H. Yu, D. Greenbaum et al., "A bayesian networks approach for predicting protein-protein interactions from genomic data," *Science*, vol. 302, no. 5644, pp. 449–453, 2003.

[11] A. Patil and H. Nakamura, "HINT—a database of annotated protein-protein interactions and their homologs," *Biophysics*, vol. 1, pp. 21–24, 2005.

[12] I. Kim, Y. Liu, and H. Zhao, "Bayesian methods for predicting interacting protein pairs using domain information," *Biometrics*, vol. 63, no. 3, pp. 824–833, 2007.

[13] M. Deng, S. Mehta, F. Sun, and T. Chen, "Inferring domain-domain interactions from protein-protein interactions," *Genome Research*, vol. 12, no. 10, pp. 1540–1548, 2002.

[14] I. Iossifov, M. Krauthammer, C. Friedman et al., "Probabilistic inference of molecular networks from noisy data sources," *Bioinformatics*, vol. 20, no. 8, pp. 1205–1213, 2004.

[15] L. V. Zhang, S. L. Wong, O. D. King, and F. P. Roth, "Predicting co-complexed protein pairs using genomic and proteomic data integration," *BMC Bioinformatics*, vol. 5, article no. 38, 2004.

[16] Y. Liu, I. Kim, and H. Zhao, "Protein interaction predictions from diverse sources," *Drug Discovery Today*, vol. 13, no. 9-10, pp. 409–416, 2008.

[17] A. Ben-Hur and W. S. Noble, "Kernel methods for predicting protein-protein interactions," *Bioinformatics*, vol. 21, no. 1, pp. i38–i46, 2005.

[18] R. A. Craig and L. Liao, "Improving protein-protein interaction prediction based on phylogenetic information using a least-squares support vector machine," *Annals of the New York Academy of Sciences*, vol. 1115, pp. 154–167, 2007.

[19] A. Patil and H. Nakamura, "Filtering high-throughput protein-protein interaction data using a combination of genomic features," *BMC Bioinformatics*, vol. 6, article no. 100, 2005.

[20] F. Azuaje, H. Wang, H. Zheng, O. Bodenreider, and A. Chesneau, "Predictive integration of gene ontology-driven similarity and functional interactions," in *Proceedings of the 6th IEEE International Conference on Data Mining*, pp. 114–119, 2006.

[21] C. M. Deane, L. Salwiński, I. Xenarios, and D. Eisenberg, "Protein interactions: two methods for assessment of the reliability of high throughput observations," *Molecular and Cellular Proteomics*, vol. 1, no. 5, pp. 349–356, 2002.

[22] R. Saeed and C. Deane, "An assessment of the uses of homologous interactions," *Bioinformatics*, vol. 24, no. 5, pp. 689–695, 2008.

[23] M. Pellegrini, D. Haynor, and J. M. Johnson, "Protein interaction networks," *Expert Review of Proteomics*, vol. 1, no. 2, pp. 239–249, 2004.

[24] P. Uetz, L. Glot, G. Cagney et al., "A comprehensive analysis of protein-protein interactions in Saccharomyces cerevisiae," *Nature*, vol. 403, no. 6770, pp. 623–627, 2000.

[25] T. Ito, T. Chiba, R. Ozawa, M. Yoshida, M. Hattori, and Y. Sakaki, "A comprehensive two-hybrid analysis to explore the yeast protein interactome," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 98, no. 8, pp. 4569–4574, 2001.

[26] A. C. Gavin, M. Bösche, R. Krause et al., "Functional organization of the yeast proteome by systematic analysis of protein complexes," *Nature*, vol. 415, no. 6868, pp. 141–147, 2002.

[27] Y. Ho, A. Gruhler, A. Heilbut et al., "Systematic identification of protein complexes in Saccharomyces cerevisiae by mass spectrometry," *Nature*, vol. 415, no. 6868, pp. 180–183, 2002.

[28] A. J. M. Walhout, R. Sordella, X. Lu et al., "Protein interaction mapping in C. elegans Using proteins involved in vulval development," *Science*, vol. 287, no. 5450, pp. 116–122, 2000.

[29] S. Li, C. M. Armstrong, N. Bertin et al., "A map of the interactome network of the metazoan C. elegans," *Science*, vol. 303, no. 5657, pp. 540–543, 2004.

[30] L. Giot, J. S. Bader, C. Brouwer et al., "A protein interaction map of Drosophila melanogaster," *Science*, vol. 302, no. 5651, pp. 1727–1736, 2003.

[31] E. Formstecher, S. Aresta, V. Collura et al., "Protein interaction mapping: a Drosophila case study," *Genome Research*, vol. 15, no. 3, pp. 376–384, 2005.

[32] T. Bouwmeester, A. Bauch, H. Ruffner et al., "A physical and functional map of the human TNF-α/NF-κB signal transduction pathway," *Nature Cell Biology*, vol. 6, no. 2, pp. 97–105, 2004.

[33] Y. Qi, J. Klein-Seetharaman, and Z. Bar-Joseph, "Random forest similarity for protein-protein interaction prediction from multiple sources," in *Proceedings of the Pacific Symposium on Biocomputing*, pp. 531–542, 2005.

[34] H. Yu, P. Braun, M. A. Yildirim et al., "High-quality binary protein interaction map of the yeast interactome network," *Science*, vol. 322, no. 5898, pp. 104–110, 2008.

[35] X. Wu, L. Zhu, J. Guo, D. Y. Zhang, and K. Lin, "Prediction of yeast protein-protein interaction network: insights from the Gene Ontology and annotations," *Nucleic Acids Research*, vol. 34, no. 7, pp. 2137–2150, 2006.

[36] H. Wang, F. Azuaje, O. Bodenreider, and J. Dopazo, “Gene expression correlation and gene ontology-based similarity: an assessment of quantitative relationships,” in *Proceedings of the IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB ’04)*, pp. 25–31, 2004.

[37] G. O. Consortium, “The gene ontology (GO) database and informatics resource,” *Nucleic Acids Research*, vol. 32, pp. D258–D261, 2004.

[38] J. Yu, M. Guo, C. J. Needham, Y. Huang, L. Cai, and D. R. Westhead, “Simple sequence-based kernels do not predict protein-protein interactions,” *Bioinformatics*, vol. 26, no. 20, Article ID btq483, pp. 2610–2614, 2010.

[39] H. Peng, F. Long, and C. Ding, “Feature selection based on mutual information: criteria of Max-Dependency, Max-Relevance, and Min-Redundancy,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1226–1238, 2005.

[40] P. Block, J. Paern, E. Hüllermeier, P. Sanschagrin, C. A. Sotriffer, and G. Klebe, “Physicochemical descriptors to discriminate protein-protein interactions in permanent and transient complexes selected by means of machine learning algorithms,” *Proteins: Structure, Function and Genetics*, vol. 65, no. 3, pp. 607–622, 2006.

[41] M. J. Mizianty and L. Kurgan, “Modular prediction of protein structural classes from sequences of twilight-zone identity with predicting sequences,” *BMC Bioinformatics*, vol. 10, article 414, 2009.

[42] E. Camon, M. Magrane, D. Barrell et al., “The gene ontology annotation (GOA) database: sharing knowledge in uniprot with gene oncology,” *Nucleic Acids Research*, vol. 32, pp. D262–D266, 2004.

[43] U. Güldener, M. Münsterkötter, G. Kastenmüller et al., “CYGD: the comprehensive yeast genome database,” *Nucleic Acids Research*, vol. 33, pp. D364–D368, 2005.

[44] A. Stein, R. B. Russell, and P. Aloy, “3did: interacting protein domains of known three-dimensional structure,” *Nucleic Acids Research*, vol. 33, pp. D413–D417, 2005.

[45] B. Boeckmann, A. Bairoch, R. Apweiler et al., “The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003,” *Nucleic Acids Research*, vol. 31, no. 1, pp. 365–370, 2003.

[46] R. Roslan, R. M. Othman, Z. A. Shah et al., “Utilizing shared interacting domain patterns and Gene Ontology information to improve protein-protein interaction prediction,” *Computers in Biology and Medicine*, vol. 40, no. 6, pp. 555–564, 2010.

[47] H. M. Berman, J. Westbrook, Z. Feng et al., “The protein data bank,” *Nucleic Acids Research*, vol. 28, no. 1, pp. 235–242, 2000.

[48] T. U. Consortium, “The universal protein resource (UniProt),” *Nucleic Acids Research*, vol. 35, pp. D193–D197, 2007.

[49] R. D. Finn, J. Tate, J. Mistry et al., “The Pfam protein families database,” *Nucleic Acids Research*, vol. 36, no. 1, pp. D281–D288, 2008.

[50] P. A. Estévez, M. Tesmer, C. A. Perez, and J. M. Zurada, “Normalized mutual information feature selection,” *IEEE Transactions on Neural Networks*, vol. 20, no. 2, pp. 189–201, 2009.

[51] G. John, R. Kohavi, and K. Pfleger, “Irrelevant features and the subset selection problem,” in *Proceedings of the International Conference on Machine Learning*, pp. 121–126, 1994.

[52] J. Bins and B. A. Draper, “Feature selection from huge feature sets,” in *Proceedings of the 8th International Conference on Computer Vision*, vol. 2, pp. 159–165, July 2001.

[53] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, John Wiley & Sons, Hoboken, NJ, USA, Second edition, 2006.

[54] S. Kullback, *Information Theory and Statistics*, Dover Publications, Mineola, NY, USA, 1997.

[55] C. Cortes and V. Vapnik, “Support-vector networks,” *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.

[56] L. J. Herrera, H. Pomares, I. Rojas, A. Guillén, A. Prieto, and O. Valenzuela, “Recursive prediction for long term time series forecasting using advanced models,” *Neurocomputing*, vol. 70, no. 16–18, pp. 2870–2880, 2007.

[57] A. Statnikov, L. Wang, and C. F. Aliferis, “A comprehensive comparison of random forests and support vector machines for microarray-based cancer classification,” *BMC Bioinformatics*, vol. 9, article 319, 2008.

[58] T. Wu, C. Lin, and R. C. Weng, “Probability estimates for multi-class classification by pairwise coupling,” *Journal of Machine Learning Research*, vol. 5, pp. 975–1005, 2004.

[59] J. A. K. Suykens, T. V. Gestel, J. D. Brabanter, B. D. Moor, and J. Vandewalle, *Least Squares Support Vector Machines*, World Scientific Publishing Company, 2003.

[60] I. Rojas, H. Pomares, J. Gonzáles et al., “Analysis of the functional block involved in the design of radial basis function networks,” *Neural Processing Letters*, vol. 12, no. 1, pp. 1–17, 2000.

[61] T. Zhang, "Statistical behavior and consistency of classification methods based on convex risk minimization," *The Annals of Statistics*, vol. 32, no. 1, pp. 56–85, 2004.

[62] C. Chang and C. Lin, LIBSVM: a Library for Support Vector Machines, 2001, http://www.csie.ntu.edu.tw/main.php.

[63] J. Macqueen, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability*, pp. 281–297, University of California Press, Berkeley, Calif, USA, 1967.

[64] A. Guillén, H. Pomares, J. González, I. Rojas, O. Valenzuela, and B. Prieto, "Parallel multiobjective memetic RBFNNs design and feature selection for function approximation problems," *Neurocomputing*, vol. 72, no. 16–18, pp. 3541–3555, 2009.

[65] A. Guillen, D. Sovilj, A. Lendasse, F. Mateo, and I. Rojas, "Minimising the delta test for variable selection in regression problems," *International Journal of High Performance Systems Architecture*, vol. 1, no. 4, pp. 269–281, 2008.

[66] A. Kumar, S. Agarwal, J. A. Heyman et al., "Subcellular localization of the yeast proteome," *Genes and Development*, vol. 16, no. 6, pp. 707–719, 2002.

[67] F. Browne, H. Wang, H. Zheng, and F. Azuaje, "Supervised statistical and machine learning approaches to inferring pairwise and module-based protein interaction networks," in *Proceedings of the 7th IEEE International Conference on Bioinformatics and Bioengineering*, pp. 1365–1369, 2007.

[68] H. Zheng, H. Wang, and D. H. Glass, "Integration of genomic data for inferring protein complexes from global protein-protein interaction networks," *IEEE Transactions on Systems, Man, and Cybernetics Part B*, vol. 38, no. 1, pp. 5–16, 2008.

[69] F. Browne, H. Wang, H. Zheng, and F. Azuaje, "A knowledge-driven probabilistic framework for the prediction of protein-protein interaction networks," *Computers in Biology and Medicine*, vol. 40, no. 3, pp. 306–317, 2010.

[70] J. Fogarty, R. S. Baker, and S. E. Hudson, "Case studies in the use of ROC curve analysis for sensor-based estimates in human computer interaction," in *Proceedings of the Graphics Interface (GI '05)*, pp. 129–136, Canadian Human-Computer Communications Society, Victoria, Canada, 2005.

[71] J. A. Hanley and B. J. McNeil, "A method of comparing the areas under receiver operating characteristic curves derived from the same cases," *Radiology*, vol. 148, no. 3, pp. 839–843, 1983.

[72] T. Jiang and A. E. Keating, "AVID: an integrative framework for discovering functional relationship among proteins," *BMC Bioinformatics*, vol. 6, article 136, 2005.