Bioinformatics, 32(14), 2016, 2184–2192
doi: 10.1093/bioinformatics/btw219
Advance Access Publication Date: 23 April 2016
Original Paper

OXFORD

Bioimage informatics

Incorporating organelle correlations into semi-supervised learning for protein subcellular localization prediction

Ying-Ying Xu, Fan Yang and Hong-Bin Shen*

Institute of Image Processing and Pattern Recognition, Key Laboratory of System Control and Information Processing, Ministry of Education of China, Shanghai Jiao Tong University, Shanghai, 200240, China

*To whom correspondence should be addressed.

Associate Editor: Alfonso Valencia

Received on August 2, 2015; revised on March 28, 2016; accepted on April 18, 2016

Abstract

Motivation: Bioimages of subcellular protein distribution as a new data source have attracted much attention in the field of automated prediction of proteins subcellular localization. Performance of existing systems is significantly limited by the small number of high-quality images with explicit annotations, resulting in the small sample size learning problem. This limitation is more serious for the multi-location proteins that co-exist at two or more organelles, because it is difficult to accurately annotate those proteins by biological experiments or automated systems.
Results: In this study, we designed a new protein subcellular localization prediction pipeline aiming to deal with the small sample size learning and multi-location proteins annotation problems. Five semi-supervised algorithms that can make use of lower-quality data were integrated, and a new multi-label classification approach by incorporating the correlations among different organelles in cells was proposed. The organelle correlations were modeled by the Bayesian network, and the topology of the correlation graph was used to guide the order of binary classifiers training in the multi-label classification to reflect the label dependence relationship. The proposed protocol was applied on both immunohistochemistry and immunofluorescence images, and our experimental results demonstrated its efficiency.
Availability and implementation: The datasets and code are available at: www.csbio.sjtu.edu.cn/bioinf/CorrASemiB.
Contact: hbshen@sjtu.edu.cn
Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

A cell contains ~10^9 proteins functioning in different organelles (Chou and Shen, 2006), and knowledge of the subcellular localization of those proteins can provide important clues for understanding their functions (Glory and Murphy, 2007; Komor *et al.*, 2012). For example, mitochondria are responsible for cellular aerobic respiration and producing energy while Golgi apparatus act to process and package the biological macromolecules. Annotating proteins subcellular patterns using biological experiments is in general too expensive in both time and costs, so various automated systems have been developed in these years. The data source to the automated predictors can be either protein amino acid sequence or microscopy bioimage. The sequence-based annotation methods usually utilize the homology transfer or targeting signal to predict the location motif (Chou and Shen, 2008; Pierleoni *et al.*, 2011; Yu *et al.*, 2012). Different from the 1D amino acid sequence, the 2D bioimage-based pattern analysis can get more intuitive and quantitative spatial distribution information. The bioimage analysis has gained popularity with the rapid advance of microscopic imaging and image processing over the past two decades (Boland and Murphy, 2001; Newberg and Murphy, 2008; Xu *et al.*, 2013).

Many protein bioimage repositories with annotations of subcellular location have been constructed in recent years. For instance, the human protein atlas (HPA, http://proteinatlas.org) stores millions of microscopic images of immunohistochemistry (IHC) and immunofluorescence (IF) showing the spatial distribution of proteins in cells (Uhlén *et al.*, 2015). However, the proportion of reliable data in the HPA is still small, where only ~13% protein samples in HPA have high-quality IHC images (Xu *et al.*, 2015), and less than half of the IF images have high-reliability subcellular annotations. Under this situation, previous efforts that trained statistical supervised classifiers in the feature space extracted from just high-quality and high-reliability images will not only reduce the generalization ability, but also cause a waste of a large ratio of lower-quality and unlabeled data, using which can be a solution to the current difficulty of small sample size of high-quality samples.

Semi-supervised learning is a branch of statistical machine learning that is able to incorporate the unlabeled samples into model construction. According to the way of how to use unlabeled samples, semi-supervised approaches can be grouped in three different types. The first is the heuristic-based approach that uses unlabeled data within a supervised learning framework. For example, one of the earliest semi-supervised methods is a heuristic method, which is also known as self-training (McLachlan, 1975). It trains classifier and predicts unlabeled data to generate more labeled samples that can be used as additional training data for updating the classifier iteratively. Later on, other variants of heuristic-based approaches have been developed, such as co-training (Liu and Yuen, 2011) and CoForest method (Li and Zhou, 2007). Recently, our group also developed a knowledge-guided semi-supervised protocol following this direction, named AsemiB (Xu *et al.*, 2015). It uses cross-class transferred knowledge as a guide to add the unlabeled samples, which has achieved encouraging performance. The second group is the graph-based methods, which usually use a graph where nodes stand for the samples and edges represent the pairwise relationship between samples. Subsequently, they propagate labels from labeled nodes to unlabeled nodes (Kobayashi *et al.*, 2012; Macskassy and Provost, 2007). The third type is known as the transductive learning, which is usually built on the cluster assumption that the data tends to form discrete groups, thus they attempt to place decision boundaries in low-density regions (Collobert *et al.*, 2006).

Obviously, the mathematical hypotheses of the above semi-supervised learning algorithms are different. To test the detailed performance of different types of semi-supervised protocols on tackling the small size of high-quality samples problem in bioimaging-based protein subcellular location prediction, five widely used semi-supervised algorithms from the three different types are systematically investigated in this study. Our results showed that they performed very differently on the benchmark datasets, and also demonstrated that an ensemble of them would generate better accuracy than any single algorithm.

Another challenge in protein subcellular location prediction is handling the multi-location proteins, which simultaneously localize at two or more organelles. A recent study of applying fluorescent-protein tagging techniques on mammalian cells showed that multi-label proteins account for ~60% (Stadler *et al.*, 2013). This is a very high ratio posing the importance of a classification system that is capable of effectively learning the multiplex features. The core difference of a multi-label classification system from a single-label one is that it should be able to assign a set of labels to the query rather than one single label. The binary relevance (BR) idea has been widely used to construct the multi-label classifier (Boutell *et al.*, 2004; Xu *et al.*, 2013). It transforms one multi-label problem into several independent binary classification problems ignoring the dependence relationship among classes (Boutell *et al.*, 2004).

Such independence assumption has problems on the topic studied in this paper. Actually, there are underlying correlations among subcellular locations in cells (Wang and Li, 2013). These correlations can be spatial proximity, functional correlations, or other such reasons that can make some organelles have a higher probability of appearing simultaneously in annotations. For example, the proteins colocalizing in plasma membrane and endosomes are very common than other cases because endosomes are derived from the plasma membrane (Foster *et al.*, 2006). Modeling and incorporating such underlying correlations in classification would have important effects on prediction results (Wang and Li, 2013; Simha and Shatkay, 2014). In this article, we modeled the label correlations among subcellular locations by a Bayesian graph model, and subsequently used it to guide both the order of binary classifiers training and as additional features fed in each classifier. In this way, the label dependence can be properly modeled, which was implemented in the proposed ensemble semi-supervised framework for further performance enhancement.

2 Methods

2.1 Datasets and features

To demonstrate the general application abilities of the proposed methods, we used two types of protein images, IHC and IF, as datasets. As shown in Table 1, each type of images has three datasets, i.e. A dataset (ADN), B dataset (BDN) and the independent dataset (IDN). The ADN set contains IHC images with high-level expression or IF images with reliable annotations, which are preferred in existing studies. Samples in ADN set are regarded as labeled data in semi-supervised learning. Then those protein images with medium-level expression or uncertain annotations are collected in BDN pool, and they are taken as the candidate selective unlabeled data in semi-supervised learning. To test different protocols objectively, the IDN sets were also derived from HPA. It should be noted that IDN has

Table 1. All the datasets used in this study

Image type	Datasets	Expression quality/reliability	Number of proteins	Number of images
IHC	A dataset (ADN)	High	337	4224
	B dataset (BDN)	Medium	345	4098
	Independent dataset (IDN)	High or medium	34	400
IF	A dataset (ADN)	Supportive	1751	3375
	B dataset (BDN)	Uncertain	1716	3336
	Independent dataset (IDN)	Supportive	206	400

no overlap of proteins with training datasets, i.e. ADN and BDN. In the experiments, we evaluated different supervised and semi-supervised algorithms on the IDN, which is not contained in the training set for all the training stages. Among all the proteins we used in this study, ∼30% are multi-label proteins.

2.1.1 The IHC datasets

Protein expression quality of IHC images in the HPA database is scored as one of four different levels, i.e. high, medium, low, or very low (Uhlén *et al.*, 2015). The IHC datasets just used the best two levels (Fig. 1A). These images involve six major cellular organelles: cytoplasm, endoplasmic reticulum, Golgi apparatus, mitochondria, nucleus and vesicles. Linear spectral separation was used to unmix each IHC image into DNA and protein channels (Newberg and Murphy, 2008). Then Haralick texture features (836 dimensions) (Haralick *et al.*, 1973), DNA distribution features (four dimensions) (Newberg and Murphy, 2008) and local binary patterns features (256 dimensions) (Nanni and Lumini, 2008) were calculated to describe the subcellular pattern in each image. Daubechies 1–10 filters were used and generated ten sets of Haralick texture features referred as db1 through db10. So in each Daubechies space, 1096 numerical features were used to represent one IHC image (Xu *et al.*, 2013).

2.1.2 The IF datasets

In HPA, the location annotation of each protein is assigned a reliability score, classified as either supportive, uncertain or non-supportive based on concordance with available experimental protein characterization data (Uhlén *et al.*, 2015). There is no existing semi-supervised benchmark dataset of IF images, so we collected IF images of A-431 cell line with supportive and uncertain annotations from HPA version 13 as datasets (Fig. 1B). The IF datasets involve nine major cellular organelles: cytoplasm, cytoskeleton, endoplasmic reticulum, Golgi apparatus, mitochondria, nucleoli, nucleus, nucleus but not nucleoli and vesicles. The images were then segmented into cells, and total 714 subcellular location features (SLFs) were calculated for each cell (Hu and Murphy, 2004). We then calculated the average of features of all cells in one IF image as the feature of this image.

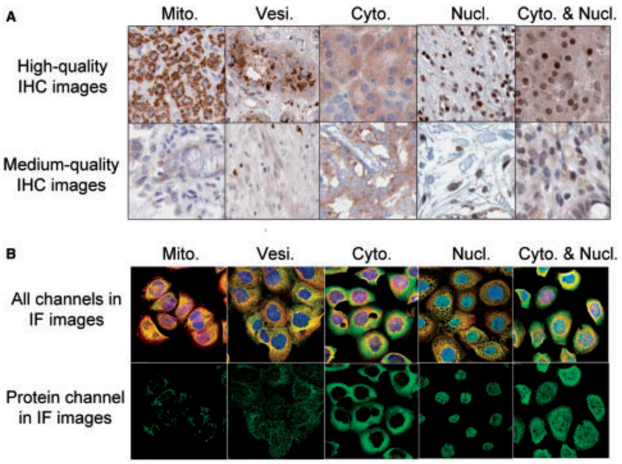

Fig. 1. Examples of IHC and IF images. (**A**) IHC images with high and medium expression levels in different subcellular locations. (**B**) IF images in different subcellular locations. The first row shows the original IF images containing four channels, i.e. protein (green), nucleus (blue), microtubules (red) and ER (yellow). The second row shows only the protein channel (Color version of this figure is available at *Bioinformatics* online.)

As shown in previous studies and our local tests, not all the features that can be calculated are useful for the classification (Newberg and Murphy, 2008; Xu *et al.*, 2015). Due to the redundancy among the features, learning model will be more likely over-fitting in a high-dimensional space, resulting in low generalization ability. Considering of this, we performed the feature selection step to achieve the feature reduction. The stepwise discriminant analysis was utilized in this study to select the most informational features, and the dimension of selected features was about 80 varying with different training sets (Newberg and Murphy, 2008; Xu *et al.*, 2015).

2.2 Semi-supervised methods

To solve the small sample size problem of the high-quality protein images, we employed five popular semi-supervised learning methods, i.e. AsemiB, logistic label propagation (LLP), low density separation (LDS), cost-sensitive semi-supervised support vector machine (CS4VM) and transductive multi-label classification (TRAM), which belong to the three types of semi-supervised learning algorithms. BR classification idea was firstly used to enable the classification system to deal with multi-location proteins, and later we will show that the performance will be improved when incorporating the label correlations. The BR model here is a one-versus-all method by training one binary classifier per label. The details of five semi-supervised methods are described below.

2.2.1 AsemiB

This method is based on the self-training framework. It builds an incremental process that incorporates candidate samples into training stage iteratively (Xu *et al.*, 2015). The initial training set is the entire high quality samples in ADN set, and the candidate samples are from BDN set. An initial BR predictor is built composing of multiple support vector machine (SVM) models, each of which corresponds to one subcellular location class. A flow chart of the iterative process of AsemiB is shown in the Supplementary Figure S1. In each iterative round, the BR classifier model is updated and used to predict the rest candidate samples, then these candidate samples whose predicted label sets match pre-defined rules are selected and added into training set for next round of updating the model. The iteration stops when the effect of adding new training samples to the classifier is small enough, where the effect was measured by the number of newly added samples and the change of the predicted scores for same images. The BR predictor trained at the last round is regarded as the final AsemiB predictor (Xu *et al.*, 2015).

2.2.2 Logistic label propagation

LLP method employs the logistic function to classify input pattern vectors, and the classifier is optimized by label propagation using the graph representation to cope with both labeled and unlabeled samples (Kobayashi *et al.*, 2012). Label propagation can propagate the labels from labeled data to unlabeled data based on a graph. In LLP, the methods of label propagation and logistic regression are integrated in terms of posterior probabilities. The learnt logistic function is used to estimate the labels of input samples, whereas the method of label propagation has to optimize the whole labels whenever an input sample comes. For each label, we built one LLP model to determine whether the samples belong to it.

2.2.3 Low density separation

This method derives a special graph kernel using the LDS criterion and employs gradient descent to solve the SVM optimization problem (Collobert *et al.*, 2006). LDS method is based on the transductive support vector machine (TSVM), which implements the cluster assumption by trying to find a hyperplane which is far away from the unlabeled points. The main drawback of TSVM is that its objective function is non-convex and difficult to minimize and different ways for optimizing it can lead to very different results. LDS addresses this issue by using gradient descent approach that directly optimizes the objective function and gets better results than other optimization strategies. One binary LDS model is built for each class under the BR framework, but all the binary models are based on the same graph, for the graph construction is based on Euclidean distances among all samples.

2.2.4 Cost-sensitive semi-supervised support vector machine

CS4VM first estimates the label means of the unlabeled instances and then trains the CS4VM with the plug-in label means by an efficient sequential minimal optimization solver (Li *et al.*, 2010). Cost information is introduced to measure the importance of different samples in different classes, and different costs reflect different amounts of losses. The aim of cost-sensitive learning is to minimize the total cost rather than the total error. The large margin principle is employed to estimate the label means, i.e. maximizing the margin between the means, which is also interpretable by Hilbert space embedding of distributions. The class with the larger misclassification cost is given a larger weight in margin computation. CS4VM is also a binary model, so one BR predictor composes of multiple CS4VM models.

2.2.5 Transductive multi-label classification

This method can effectively assign multiple labels to each sample using both labeled and unlabeled data (Kong *et al.*, 2013). TRAM first formulates the transductive multi-label classification as an optimization problem of estimating label concept compositions, then derives a closed-form solution to this optimization problem and adopts an effective algorithm to assign label sets to the unlabeled instances. This algorithm is designed for the TRAM problem, so there is no need to use BR strategy. The TRAM can as well output the scores representing whether the input sample belongs to the labels.

2.3 Label set criteria

To determine the set of labels for a testing sample from its score vector outputted from the classifier, specific criteria should be applied. For the *i*th testing image, the semi-supervised predictor would output a score vector $\boldsymbol{s}_i = [s_{i1}, s_{i2}, \cdots, s_{iN}]$, where N is the total number of classes. Each element of the score vector corresponds to the confidence of belonging to one of the N classes, i.e. $l_1, l_2, \ldots, l_N$. Suppose Y_i is the ground truth and $\widehat{Y}_i = [\widehat{y}_{i1}, \widehat{y}_{i2}, \cdots, \widehat{y}_{iN}]$ is the predicted label set vector for the *i*th image, where $\widehat{y}_{ij}$ $(j = 1, \cdots, N)$ is 1 if the image is predicted having the *j*-th label and is 0 if not. In the single-label case, only the label of argmax $\{s_{i1}, s_{i2}, \cdots, s_{iN}\}$ is 1 in $\widehat{Y}_i$. In the multi-label classification, more than one element in $\widehat{Y}_i$ might be 1, it is a vector and we have to further determine the length of it.

We used three criteria for determining the label sets in this study. Top criterion (*T-criterion*) is a simple and widely used approach. It simply uses 0 as the threshold to decide whether to assign a label (Boutell *et al.*, 2004). A dynamic threshold criterion (*D-criterion*) that calculates one specific threshold for each score vector considering the specificity of each image sample was proposed in (Xu et al., 2015), which is demonstrated capable of enhancing the classification accuracy. In this article, we unified the *D-criterion* into a new *U-criterion* to make it applicable in a more general case, where all the scores for a test sample are negative.

2.3.1 T-criterion

Top criterion (*T-criterion*) is the most widely used approach for determining label sets in the multi-label learning (Boutell *et al.*, 2004; Xu *et al.*, 2013). Its basic hypothesis is that the score value is positive if the corresponding binary classifier predicts the image belongs to the corresponding class, and negative if not. It can be written as:

$$\widehat{y}_{ij} = \begin{cases} 1, & \mathit{if}\ s_{ij} \ge 0\ \mathit{or}\ s_{ij} = s_{i,\max} \\ 0, & \mathit{otherwise} \end{cases} \tag{1}$$

It considers the label set consisting of labels with positive scores, and if all scores in the vector are negative, the label with the maximum score is considered as the unique label. This criterion does not have parameters and is easy to use, but it just takes 0 as the decision threshold without considering the specificity of images. The scales of score vectors for different images can differ a lot, so a static unified threshold may not fit all the images.

2.3.2 D-criterion

Dynamic threshold criterion (*D-criterion*) calculates a specific threshold for each sample according to the scale and distribution of its score vector (Xu *et al.*, 2015). Compared with the *T-criterion*, the *D-criterion* is adaptable to each query image, hence is able to generate better score segmentation. The assumption of *D-criterion* is that the score values corresponding to the real labels are the largest, and its real labels will have high and similar scores in the case of a multi-label sample. *D-criterion* defines a novel way to measure whether the top score values are close enough. The *D-criterion* can be presented as:

$$\widehat{y}_{ij} = \begin{cases} 1, & \mathit{if}\ s_{ij} \ge \theta\ \mathit{or}\ \dfrac{s_{i,\max} - s_{ij}}{s_{i,\max}} \le t,\ s_{i,\max} \ge 0 \\ 1, & \mathit{if}\ s_{ij} = s_{i,\max},\ s_{i,\max} < 0 \\ 0, & \mathit{otherwise} \end{cases} \tag{2}$$

where $s_{i,\max}$ is the maximum value in the score vector, t and θ are two constant parameters for controlling the decision boundary, which are estimated through the maximum a posteriori principle on training set by cross validation test (Xu *et al.*, 2015).

2.3.3 U-criterion

As can be seen from Equation (2) that, when all the scores are negative, the *T-criterion* and *D-criterion* will be degenerated to a single-label decision. To solve this problem, we extend the *D-criterion* to the uniform dynamic threshold criterion (*U-criterion*), which is formulated as:

$$\widehat{y}_{ij} = \begin{cases} 1, & \mathit{if}\, s_{ij} \ge \theta\ \mathit{or}\ \dfrac{s_{i,\max} - s_{ij}}{|s_{i,\max}|} \le t \\ 0, & \mathit{otherwise} \end{cases} \tag{3}$$

It only differs from *D-criterion* in terms of cases when all the scores are negative, otherwise they are the same. This is more interpretable and generally applicable because those samples with all-negative scores may also be multi-label subcellular location proteins, which we also observed in our experiments.

2.4 Modeling and incorporating cell organelle correlations

As stated before, the BR framework that separates a multi-label problem into N binary classifiers ignores the relationship among labels. Because there are correlations among organelles virtually

(Dell'Angelica *et al.*, 2000; Foster *et al.*, 2006), we are interested in finding out whether incorporating these correlations is helpful in protein subcellular pattern analysis. In this study, a three-step method as following was designed to reveal this problem (Fig. 2).

2.4.1 Construct correlation graph

The first problem is modeling the correlations among the subcellular locations. In this study, the correlation graph was modeled as a Bayesian network where nodes represent subcellular locations and edges represent dependent relations. The correlations among labels were represented by a directed acyclic graph (DAG) structure, where for example, the edge from l_1 to l_2 can be interpreted as that the label l_1 implies the label l_2 with high probability (Fig. 2A). To learn the network structure, we chose the Bayesian DAG learning (BDAGL) package to calculate the underlying links among the nodes of labels (http://www.cs.ubc.ca/~murphyk/Software/BDAGL/). The BDAGL implements the dynamic programming-based algorithm for computing the marginal posterior probability of every edge in a Bayesian network (Eaton and Murphy, 2007). We used labels of the training set as input in the learning process, and each node of label was represented by a 0–1 vector, where 1 represents that a training sample belongs to this label and 0 represents not.

2.4.2 Build independent classifiers in the first stage

A BR predictor containing N independent SVM classifiers was built based on original features. For the i-th testing sample, the predictor can predict its score vector, and then get rough label $\widehat{Y}_i$ by label set criterion (Fig. 2B). The $\widehat{Y}_i$ will be updated and used as features in the next iterative stage.

2.4.3 Build chain classifiers by incorporating label correlations in the second stage

In this stage, N new binary SVM classifiers were trained, and the feature space of each binary classifiers and the order of training these classifiers are determined by the correlation graph. For each label, the predicted labels of its parent nodes were used as additional features to build its corresponding classifier. This is because the states of parental labels are generally considered important on their children label. The order of training the N classifiers was generated by topological sorting to the DAG correlation graph (Fig. 2C and D). Topological sorting works by repeatedly choosing nodes without incoming edges and removing these nodes and their edges until all the nodes are chosen, which would generate a linear order to the nodes (Cormen, 2009). Training chain classifiers according to this order can ensure that the additional features are updated outputs, which are more accurate than rough labels from the last step. The final output of this chain algorithm is still an N-dimensional score vector, which will be used to determine the label set by applying the criterion as defined in Section 2.3.

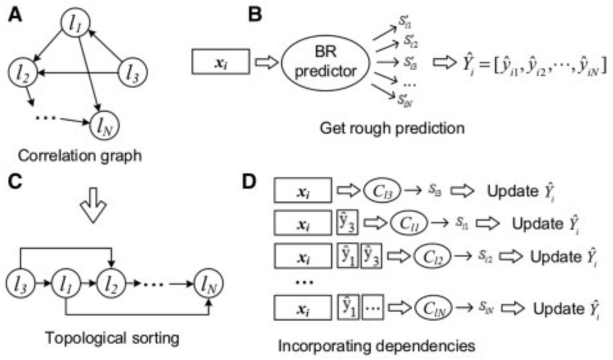

Fig. 2. Process of incorporating label correlations into classification. (**A**) An example of constructed correlation graphs. (**B**) Using normal BR model to predict initial scores and label set. The x_i is the original feature vector of the i-th sample, and $\widehat{Y}_i$ is the vector of labels of the i-th sample, where $\hat{y}_{ij}$ is 1 or 0 representing whether the sample belongs to the j-th class. (**C**) Topological sorting of the graph. All the edges point to the right. (**D**) Incorporating results of parental labels (derived from A) as additional features. The order of training classifiers (C_{l1}, C_{l2}, …, C_{lN}) is determined by topological sorting, and $\widehat{Y}_i$ is updated by every classifier's results

2.5 Evaluation metrics

In this article, five popular metrics specially designed for multi-label learning were used, i.e. subset accuracy, accuracy, recall, precision and average label accuracy. Among them, we mainly used the subset accuracy, which is the most stringent one since it requires the predicted label set to be exactly the same as the true label set, which is defined as:

$$subacc = \frac{1}{q}\sum_{i=1}^{q}\Phi(\widehat{Y}_i = Y_i) \tag{4}$$

where $\Phi(\cdot) = \begin{cases} 1, \; if \;\; \cdot \; is\, ture \\ 0, \; otherwise \end{cases}$ and q is the number of the testing samples. Details of other metrics can be found in Supplementary text.

3 Results and discussions

3.1 Results of the baseline supervised method

As a baseline, we first applied supervised learning method on the IHC and IF datasets. All the images in the ADN and BDN have subcellular location annotations from the HPA. Thus, ADN, BDN, ADN + BDN (the entire ADN and the entire BDN) were taken as training set, respectively, and the BR framework with SVM models (http://www.csie.ntu.edu.tw/~cjlin/libsvm/) was used for classification. Subsequently, the three label set criteria, i.e. *T-criterion*, *D-criterion* and *U-criterion* were utilized to determine label sets based on samples' score vectors. The main difference among the three criteria is the decision threshold they choose. That is to say, *T-criterion* just chooses 0 or $s_{i,\max}$ as threshold whereas *D-criterion* and *U-criterion* calculate specific dynamic thresholds for each sample (the thresholds vary on different samples). To investigate their differences on our data, we conducted t-test with multi-hypothesis comparison on decision thresholds of the three label set criteria.

The performances tested on IDN set are shown in Figure 3. From Figure 3A and B, we can see that results of using ADN as training set are better than that of using BDN because of the image quality and annotation reliability, and are not as good as that of using the ADN + BDN due to the less number of samples and generality. Figure 3C and D compare the results of three label set criteria. The average subset accuracies of db1–db10 using the *T-criterion*, *D-criterion* and *U-criterion* are 44.9%, 46.5% and 45.6%, respectively. *D-criterion* and its variant of *U-criterion* outperform the simplest *T-criterion*. It's worth pointing out that the input to the three different criteria methods is the same, which is the score vector outputted from the classifier. For some single db cases, the advantages are more obvious. For example, based on the same score vectors outputted from the classifier trained by db2 features, *D-criterion* and *U-criterion* got subset accuracies of 48.25% and 48%, respectively, which are over 3% higher than the *T-criterion* (45%). Results of IF datasets also illustrate that *D-criterion* and its variant of *U-criterion* have higher accuracies than *T-criterion*.

Besides, as we mentioned before, the difference between *U-criterion* and *D-criterion* is only on the samples that all the scores are negative. In our experiments, there are some examples that can demonstrate the advantage of *U-criterion*. For example, a testing IHC image of protein SLC3A2, whose ground truth is nucleus and cytoplasm, got the score vector of [-0.8754, -1.4154, -1.0021, -1.3529, -0.8286, -1.0016], where all the scores are negative. For this sample, the *D-criterion* just predicted it belonging to the fifth class (nucleus), whereas the *U-criterion* predicted it belonging to both the fifth and the first class (nucleus and cytoplasm). All of these results highlight the importance of label decision criterion selection in multi-label classification. Figure 3E and F further show the means and standard deviations of decision thresholds chosen in the three criteria, and it can be seen that differences among those three criteria are all statistically significant different with *P*-value <0.001 (Bonferroni-corrected).

3.2 Results of different semi-supervised methods

We tested five semi-supervised methods, i.e. AsemiB, LLP, LDS, CS4VM and TRAM on the benchmark datasets. Since the ADN dataset has the highest expression level and are generally chosen as the training set, we took ADN as the labeled data. Afterwards, BDN was used as the candidate unlabeled data, and IDN was independent testing set. We also used the three criteria for label set determination.

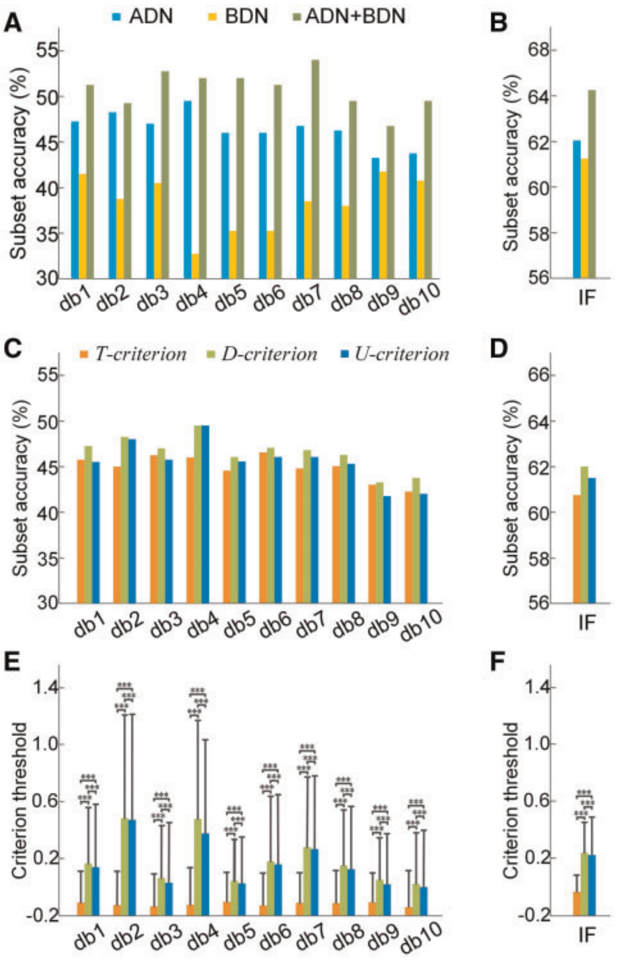

Fig. 3. Results of supervised methods using *T-criterion*, *D-criterion* and *U-criterion*. (**A**, **B**) Subset accuracy results of supervised methods on different datasets. *D-criteria* were used here. (**C**, **D**) Subset accuracy results of three label set criteria on ADN datasets. (**E**, **F**) Means and standard deviations of thresholds chosen in three label set criteria. Group differences are marked by *, ** and ***, corresponding to the significance level 0.05, 0.01 and 0.001, respectively (Color version of this figure is available at *Bioinformatics* online.)

Figure 4A shows the results of using semi-supervised method AsemiB. The average subset accuracies of db1–db10 by using *T-criterion*, *D-criterion* and *U-criterion* are 49, 50.98 and 51.67%, respectively. Although *U*-criterion is a little bit better than the *D*-criterion here, but overall they are comparable to each other due to *U*-criterion is an extension from the *D*-criterion (refer to Equations (2) and (3)). In the following experiments, we will mainly report results from the *D-criterion*.

The comparison of semi-supervised methods and the baseline supervised method shows that AsemiB method outperforms other tested methods (Fig. 4B and C). It should be noted that the final predictor of AsemiB was trained by the entire ADN set and some automatically selected samples from BDN. Its results are higher than that of the baseline supervised model trained on ADN and also higher than the supervised model trained on ADN + BDN in 7 of the 10 db cases. These results indicate the interesting phenomenon that more training samples do not always result in better performance because some lower-quality data in the BDN may degenerate the system. The LLP method as a graph-based method gets the second best performance on our datasets. Other semi-supervised approaches are

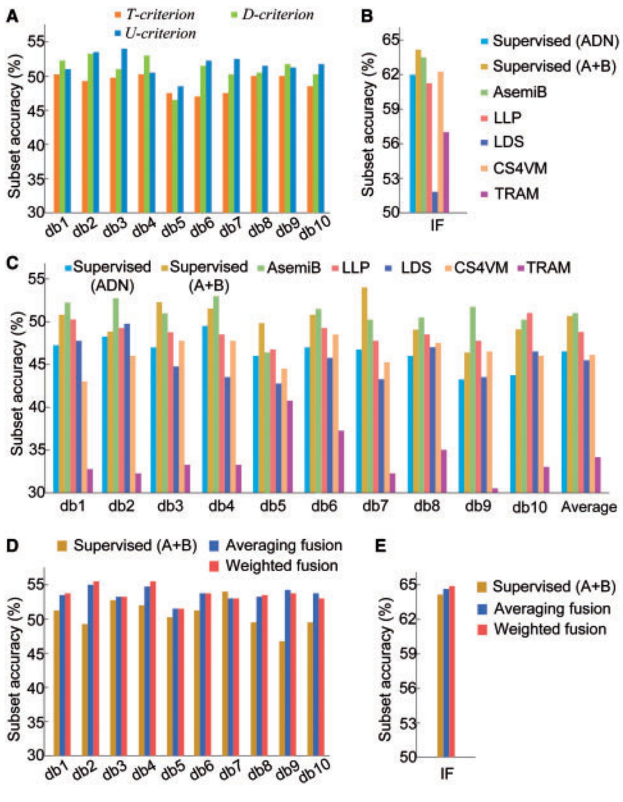

Fig. 4. Results of semi-supervised methods. (**A**) Results of three criteria using AsemiB method on the IHC datasets. (**B**) Comparison of supervised methods and five semi-supervised methods on IF datasets using *D-criterion*. (**C**) Comparison of supervised methods and five semi-supervised methods on IHC datasets using *D-criterion*. The last batch of bars give the average results of db1–db10. (**D**) Comparison of supervised classifiers trained using ADN + BDN and ensemble semi-supervised classifiers on IHC datasets. (**E**) Comparison of supervised classifiers trained using ADN + BDN and ensemble semi-supervised classifiers on IF datasets (Color version of this figure is available at *Bioinformatics* online.)

three transductive methods, and their results are not as good as LLP and the heuristic-based AsemiB method. The reason might be that transductive methods are based on cluster assumption but the multi-label samples are near the boundaries, making it difficult to find low density regions. These results also tell us that all of these methods have their own applicable scopes according to different mechanisms.

3.3 Performance of the ensemble classifiers

Different mathematical mechanisms make these semi-supervised approaches diverse from each other, resulting in their complementarities. Hence, to achieve better classification performance, we fused above five semi-supervised algorithms. Two fusion methods at the decision level were used: averaging fusion and weighted fusion. The averaging fusion just averages the score vectors outputted from different semi-supervised classifiers. However, simply averaging has the problem that those methods of bad performance may drag down the final ensemble. Therefore, we used a weighted fusion by giving different weights to different semi-supervised methods according to their performance. The calculation formula of weight of the *k*th approach is:

$$\omega_k = \frac{subacc_k}{\sum_{i=1}^{n} subacc_i} \quad (5)$$

where $n = 5$ is the number of approaches, and $subacc_k$ is the subset accuracy achieved by the *k*th approach tested on IDN. Then the score vectors outputted from different semi-supervised approaches were weighted and summed.

Figure 4D and E compare the results between ensemble semi-supervised classifiers of two fusing methods and the baseline supervised classifiers. More results of ensemble can be seen in Supplementary Table S1. Overall, ensemble classifiers using either averaging fusion or weighted fusion can get higher accuracies than supervised learning. For example, the ensemble semi-supervised classifier using weighted fusion based on db2 features of IHC data can achieve 55.5% of subset accuracy and 87.04% of averaged label accuracy (shown in Supplementary Table S1), which are 6.25% and 2. 33% higher than the baseline predictor. On IF datasets, the subset accuracy of the ensemble classifier using weighted fusion gets 65%, better than that of averaged fusion and supervised classifier. In addition, the weighted fusion (IHC average 53.65%, IF 65%) slightly outperforms the equally weighted averaging fusion as illustrated in Figure 4D and E.

3.4 Results of label correlation-driven chain prediction

For testing the effects of organelle correlations, we replaced the BR classifiers in the framework of AsemiB with the proposed two-stage chain classifier protocol that incorporates the label correlation information (Fig. 2). Figure 5A and B present the dynamic correlation graphs and classifier performance obtained by the AsemiB on independent testing set IDN. The correlation graphs represent the dependencies among different subcellular locations, and the dependencies are co-occurrence (Read *et al.*, 2015). Since the training set

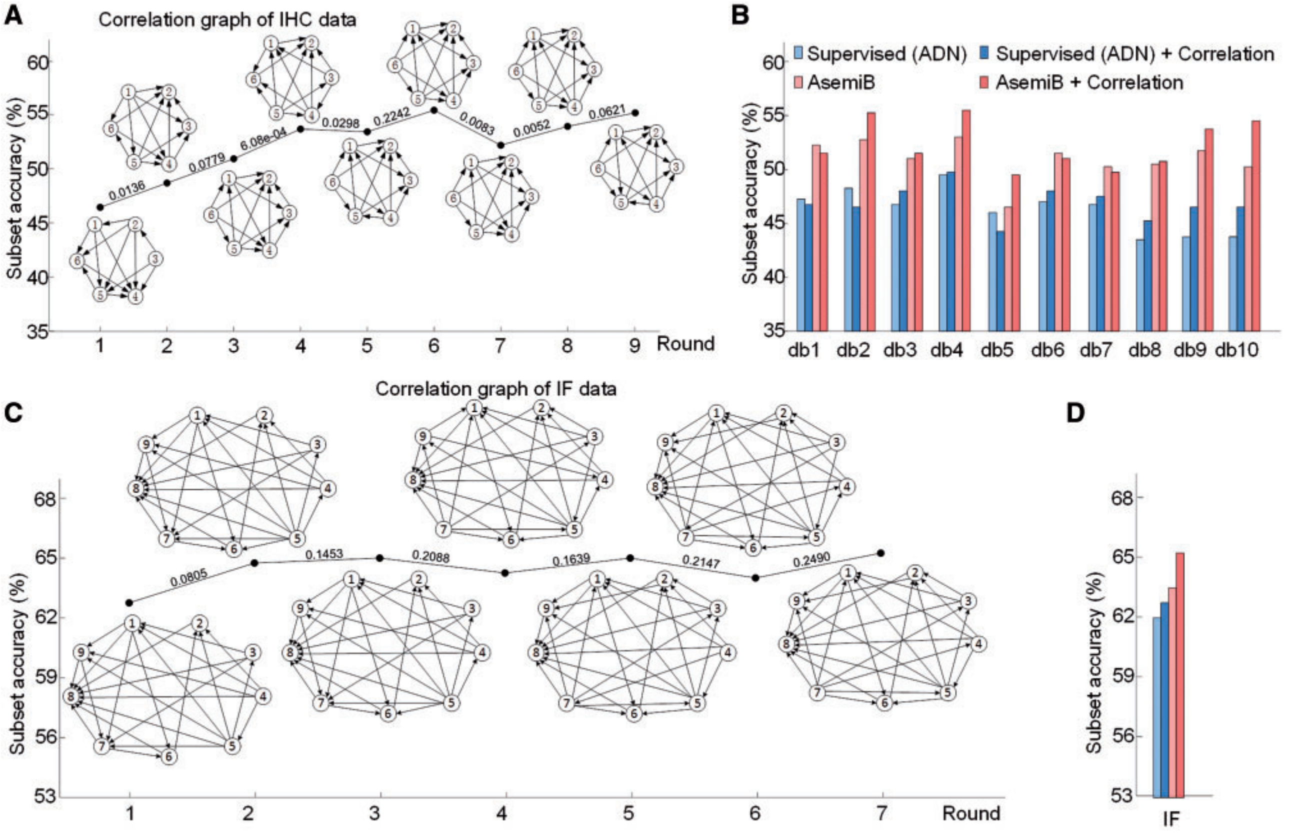

Fig. 5. Results of the AsemiB method incorporating label correlations. (**A**) Results of iterative process of AsemiB on IHC db2 data. The numbers 1–6 in the nodes represent cytoplasm, ER, Golgi apparatus, mitochondria, nucleus and vesicles, respectively. The *P*-values are shown above the lines between adjacent rounds. (**B**) Results of iterative process of AsemiB on IF data. The numbers 1–9 in the nodes represent cytoplasm, cytoskeleton, ER, Golgi apparatus, mitochondria, nucleoli, nucleus, nucleus but not nucleoli and vesicles, respectively. (**C**) Comparison of IHC data results before and after incorporating correlation graph. (**D**) Comparison of IF data results before and after incorporating correlation graph (Color version of this figure is available at *Bioinformatics* online.)

grows as the semi-supervised iterations proceed (Supplementary Fig. S1), the correlation graph is updated accordingly in each round, which gives a dynamic label correlation chain.

These graphs change in edges and directions of edges in different iterative rounds. For example, the graph of IHC at the second round has an additional edge from node 1 to node 3 than the graph at the first round, whereas the graph of IF at fifth round cuts the edge from node 9 to node 4 than that at fourth round. These DAG graphs are learned from training data, so their stability is getting better as increasing of the training set by adding the samples from BDN.

In terms of the classification performance, we can see that in the iteration process of AsemiB, the subset accuracies show a rising trend in spite of some slight fluctuations and reached stable at the end of iterations. This demonstrates that incorporating correlations can achieve reasonable and enhanced results. Additionally, the *t*-test was used to measure whether incorporating the label correlation will make the updated classifier different. We used their output scores of IDN to calculate the *P*-values in different classes. The averaged *P*-values are shown in Figure 5A and B. Three observations can be found from Figure 5:

1. Most of the differences between adjacent rounds are significant (*P*-value <0.05) on IHC data, whereas the adjacent differences on IF data are not statistically significant (*P*-value >0.05). The reason could be that the IF has over five times more proteins than the IHC, and the relatively bigger data makes its trained classifier and graph more stable. By carefully checking the *P*-values along the IF iterations, we can see that the *P*-values have an increasing trend. This also indicates the classifiers are increasingly stable along the iterations.
2. Besides the *t*-tests between the adjacent iterations, we also conducted *t*-test on the classifiers between the first and the last round to evaluate the overall effects by incorporating the label correlations. The *P*-value is 0.0017 for IHC dataset, indicating their prediction scores are indeed changed statistically significant after incorporating correlations of labels. The *P*-value is 0.0134 for IF dataset, which also demonstrates the overall significance by taking the label correlations.
3. The comparisons of subset accuracy results shown in Figure 5C and D also demonstrate that incorporating organelle correlations has better classification performance than that of using independent BR method.

To get a final classifier for predicting IHC image-based proteins subcellular patterns, we combined the db1–db10 classifiers trained by AsemiB as well as incorporating correlation chain method. We averaged the 10 classification outputs and got a final classification for the testing set IDN. The subset accuracy achieved 56%, which is 0.75–6.25% higher than the single classifiers.

4 Conclusions and future directions

In this article, we construct multi-label semi-supervised systems for predicting human protein subcellular localization of IHC and IF images from the HPA database. The systems are featured with an ensemble fused by five different semi-supervised approaches, and capable of dealing with both single- and multi-location proteins. Using semi-supervised idea enables us to make use of those images with lower-quality expression or uncertain annotations, which account for a large proportion in HPA. This not only enhanced the generality of our system but also avoided the waste of imaging and annotation work.

The other contribution of this paper is that, to the best of our knowledge, we are the first group to propose a method that incorporates organelle correlations as a DAG structure into the semi-supervised framework for bioimage-based subcellular location classification. Our experimental results have demonstrated its efficacy, which shed new light on the challenging bioimage-based multi-label sample classification problems. Our future work will exploit more in this direction. For instance, the correlations between two organelles also can be regarded as mutual, and the correlation graph would be represented with an undirected graph. We also plan to test the feasibility of deriving this correlation graph from the much larger proteomics and pathway databases to integrate the bioimaging data with the proteomics data sources. Previous study (Li *et al.*, 2012) has shown that part of the current HPA subcellular location annotations need to be updated, and the automated classifier is able to detect the most potential targets from a large image pool of HPA in a fast and reliable mode. Motivated by this, another future effort will apply the developed ensemble semi-supervised chain classification system of this study to a large-scale screening on current annotated images in HPA to find those targets, whose existing subcellular location annotations may be re-evaluated.

Acknowledgement

We are grateful to Wallace Chan for reading through the manuscript.

Funding

This work was supported in part by the National Natural Science Foundation of China (No. 61222306).

Conflict of Interest: none declared.

References

Boland,M.V. and Murphy,R.F. (2001) A neural network classifier capable of recognizing the patterns of all major subcellular structures in fluorescence microscope images of HeLa cells. *Bioinformatics*, **17**, 1213–1223.

Boutell,M.R. *et al.* (2004) Learning multi-label scene classification. *Pattern Recogn.*, **37**, 1757–1771.

Collobert,R. *et al.* (2006) Large scale transductive SVMs. *J. Mach. Learn. Res.*, **7**, 1687–1712.

Chou,K.C. and Shen,H.B. (2006) Predicting protein subcellular location by fusing multiple classifiers. *J. Cell Biochem.*, **99**, 517–527.

Chou,K.C. and Shen,H.B. (2008) Cell-PLoc: a package of web servers for predicting subcellular localization of proteins in various organisms. *Nat. Protoc.*, **3**, 153–162.

Cormen,T.H. (2009) *Introduction to Algorithms*. MIT Press, Cambridge, Massachusetts (United States).

Dell'Angelica,E. *et al.* (2000) Lysosome-related organelles. *Faseb J.*, **14**, 1265–1278.

Eaton,D. and Murphy,K. (2007) Belief net structure learning from uncertain interventions. *J. Mach. Learn. Res*, **1**, 1–48.

Foster,L.J. *et al.* (2006) A mammalian organelle map by protein correlation profiling. *Cell*, **125**, 187–199.

Glory,E. and Murphy,R.F. (2007) Automated subcellular location determination and high-throughput microscopy. *Dev. Cell*, **12**, 7–16.

Haralick,R.M. *et al.* (1973) Textural features for image classification. *IEEE Trans. Syst. Man Cybern.*, **6**, 610–621.

Hu,Y. and Murphy,R.F. (2004) Automated interpretation of subcellular patterns from immunofluorescence microscopy. *J. Immunol. Methods*, **290**, 93–105.

Kobayashi,T. *et al.* (2012) Logistic label propagation. *Pattern Recogn. Lett*, **33**, 580–588.

Komor,A.C. *et al.* (2012) Cell-selective biological activity of rhodium metalloinsertors correlates with subcellular localization. *J. Am. Chem. Soc*, **134**, 19223–19233.

Kong,X. *et al.* (2013) Transductive multilabel learning via label set propagation. *IEEE T. Knowl. Data En*, **25**, 704–719.

Li,J. *et al.* (2012) Automated analysis and reannotation of subcellular locations in confocal images from the human protein atlas. *PLoS One*, **7**, 0050514.

Li,Y.F. *et al.* (2010) Cost-sensitive semi-supervised support vector machine. In: *AAAI Conference on Artificial Intelligence, Atlanta, Georgia*, pp. 500–505.

Li,M. and Zhou,Z.H. (2007) Improve computer-aided diagnosis with machine learning techniques using undiagnosed samples. *IEEE Trans. Syst. Man. Cybern. A*, **37**, 1088–1098.

Liu,C. and Yuen,P.C. (2011) A boosted co-training algorithm for human action recognition. *IEEE Trans. Circuits Syst. Video Technol.*, **21**, 1203–1213.

Macskassy,S.A. and Provost,F. (2007) Classification in networked data: a toolkit and a univariate case study. *J. Mach. Learn. Res.*, **8**, 935–983.

McLachlan,G.J. (1975) Iterative reclassification procedure for constructing an asymptotically optimal rule of allocation in discriminant analysis. *J. Am. Stat. Assoc.*, **70**, 365–369.

Nanni,L. and Lumini,A. (2008) Local binary patterns for a hybrid fingerprint matcher. *Pattern Recogn.*, **41**, 3461–3466.

Newberg,J. and Murphy,R.F. (2008) A framework for the automated analysis of subcellular patterns in human protein atlas images. *J. Proteome Res.*, **7**, 2300–2308.

Pierleoni,A. *et al.* (2011) MemLoci: predicting subcellular localization of membrane proteins in eukaryotes. *Bioinformatics*, **27**, 1224–1230.

Read,J. *et al.* (2015) Scalable multi-output label prediction: From classifier chains to classifier trellises. *Pattern Recogn.*, **48**, 2096–2109.

Simha,R. and Shatkay,H. (2014) Protein (multi-) location prediction: using location inter-dependencies in a probabilistic framework. *Algorithms Mol. Biol*, **9**, 8.

Stadler,C. *et al.* (2013) Immunofluorescence and fluorescent-protein tagging show high correlation for protein localization in mammalian cells. *Nat. Methods*, **10**, 315–323.

Uhlén,M. *et al.* (2015) Tissue-based map of the human proteome. *Science*, **347**, 1260419.

Wang,X. and Li,G.Z. (2013) Multilabel learning via random label selection for protein subcellular multilocations prediction. *IEEE ACM Trans. Comput. Bioinform.*, **10**, 436–446.

Xu,Y.Y. *et al.* (2013) An image-based multi-label human protein subcellular localization predictor (*i*Locator) reveals protein mislocalizations in cancer tissues. *Bioinformatics*, **29**, 2032–2040.

Xu,Y.Y. *et al.* (2015) Bioimaging-based detection of mislocalized proteins in human cancers by semi-supervised learning. *Bioinformatics*, **31**, 1111–1119.

Yu,D. *et al.* (2012) Enhancing membrane protein subcellular localization prediction by parallel fusion of multi-view features. *IEEE Trans. Nanobiosci.*, **11**, 375–385.