Journal of Theoretical Biology 310 (2012) 80–87

Contents lists available at SciVerse ScienceDirect

Journal of Theoretical Biology

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

Predicting plant protein subcellular multi-localization by Chou's PseAAC formulation based multi-label homolog knowledge transfer learning

Suyu Mei *

Software College, Shenyang Normal University, Shenyang, China

HIGHLIGHTS

- We define a multi-label confusion matrix.
- We use probabilistic outputs to adapt *MK-TLM* to multi-label learning scenario.
- We propose two multi-labeling performance measures: *LHRNT-LHR*.
- We conduct multi-labeling estimation in *optimistic*, *moderate* and *pessimistic* cases.

ARTICLE INFO

Article history:
Received 25 January 2012
Received in revised form
12 May 2012
Accepted 18 June 2012
Available online 27 June 2012

Keywords:
Multi-label learning
Transfer learning
Nonparametric multiple kernel learning
Protein subcellular localization
Performance overestimation

ABSTRACT

Recent years have witnessed much progress in computational modeling for protein subcellular localization. However, there are far few computational models for predicting plant protein subcellular multi-localization. In this paper, we propose a multi-label multi-kernel transfer learning model for predicting multiple subcellular locations of plant proteins (*MLMK-TLM*). The method proposes a multi-label confusion matrix and adapts one-against-all multi-class probabilistic outputs to multi-label learning scenario, based on which we further extend our published work *MK-TLM* (multi-kernel transfer learning based on Chou's PseAAC formulation for protein submitochondria localization) for plant protein subcellular multi-localization. By proper homolog knowledge transfer, *MLMK-TLM* is applicable to novel plant protein subcellular localization in multi-label learning scenario. The experiments on plant protein benchmark dataset show that *MLMK-TLM* outperforms the baseline model. Unlike the existing models, *MLMK-TLM* also reports its misleading tendency, which is important for comprehensive survey of model's multi-labeling performance.

1. Introduction

Recent years have witnessed much progress in computational modeling for protein subcellular localization (Nakashima and Nishikawa, 1994; Chou and Elrod, 1999; Chou and Cai, 2002; Chou, 2001,2011; Zhou and Doctor, 2003; Nakai, 2000; Chou and Shen, 2007,2009; Wu et al., 2012,2011; Chou et al., 2012). Some researches reported the model performance on several specific species, such as *plant*, *animal*, *fungi*, etc. (Pierleoni et al., 2006; Hoglund et al., 2006; Blum et al., 2009; Mei et al., 2011), while other researches reported the model performance on only one specific species (Shen and Chou, 2010,2007; Xiao et al., 2011a,2011b; Chou and Shen, 2007,2010), or specific cellular compartment, such as *nucleus* (Mei and Wang, 2010; Huang et al., 2009), *mitochondria* (Du and Li, 2006; Mei, 2012), *chloroplast* (Du et al., 2009), etc. No matter whether the computational models have been evaluated on various species of protein data, we had to better train the model on specific species of protein data from the viewpoints of practical use. Obviously, it is not reliable to predict animal protein subcellular localization by the computational model that is trained on plant protein data, in that animal varies widely from plant in terms of genomics. For instance, plants, animals and apicomplexa do not share the same structural transit and/or signal peptide due to the kind of membranes, as it is the product of the organism's history, endosymbiosis and so on. For these reasons, the computational model trained for plant protein subcellular localization would be more targeted to botanist.

To the best of our knowledge, there are several computational methods that reported the model performance for plant protein subcellular localization. *BaCelLo* (Pierleoni et al., 2006) achieved relatively low model performance (68.2% overall accuracy) on plant dataset with 5 subcellular locations (*Nucleus*, *Cytoplasm*, *Extracellular*, *Mitochondria* and *Chloroplast*). *MultiLoc* (Hoglund

* Tel.: +86 24 86578320.
E-mail address: 061021053@fudan.edu.cn

0022-5193/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.jtbi.2012.06.028

et al., 2006) achieved moderate model performance (74.60% overall accuracy) on 5856 plant dataset that covers 10 subcellular locations with 40% sequence similarity. *MultiLoc2* (Blum et al., 2009) remarkably improved the model performance to 89.60% overall accuracy on the same plant dataset by incorporating the gene ontology (*GO*) information to *MultiLoc* (Hoglund et al., 2006). Gene ontology (*GO*) is a controlled vocabulary that describes biomolecules and gene products in terms of biological process, molecular function and cellular component, and the *GO* annotations of proteins are organized in *GOA* database (Ashburner et al., 2000). *Plant-PLoc* (Chou and Shen, 2007) is also developed for plant protein subcellular localization, but runs the risk of performance overestimation for novel proteins as the work (Blum et al., 2009; Lee et al., 2008), because the method used the target protein's own *GO* information to train model. Generally, the *GO* information for novel proteins is unavailable. To solve the problem, many recent *GO*-based methods generally exploit the homolog *GO* information for novel protein subcellular localization (Mei et al., 2011; Shen and Chou, 2010,2007; Xiao et al., 2011a,2011b; Chou and Shen, 2010; Huang et al., 2009,2008; Mei, 2012; Tung and Lee, 2009). From the viewpoints of evolutionary biology, homolog generally refers to the proteins that share the same ancestor. We can infer homolog by the evolutionary convergence/divergence of protein sequences (sequence identity). Here we assume sequence identity $\geq 60\%$ as significant homolog, and sequence identity $\geq 20\%$ as remote homolog. *Plant-mPLoc* (Chou and Shen, 2010) exploited the *GO* information from the homologs with sequence identity $\geq 60\%$ to represent the target protein. However, the method of setting threshold for homolog incorporation has the following disadvantages: (1) homolog with high sequence identity may potentially be divergent from the target protein in terms of protein subcellular localization, *for instance*, target protein *Q9BZZ2* resides in subcellular locations: *Cell membrane*, *Membrane*, *Secreted*, while its significant homolog *O88406* (sequence identity: 97.89%; *PSI-Blast E*-value: 3e-174, obtained by *Blast* default options) resides in subcellular locations: *Cytoplasm*, *Nucleus*. High threshold of sequence identity, e.g. 60%, cannot guarantee that no noise would be introduced to the target protein; (2) some novel proteins may have no significant homolog but remote homolog, thus the *GO* feature vector would be null vector and homolog knowledge transfer cannot work in this case. Although remote homologs are generally divergent from the target protein in terms of protein subcellular localization, yet many remote homologs are subjected to identical protein subcellular localization patterns as the target protein in some cases, *for instance*, target protein *Q9Y4G6* and its remote homolog *Q54K81* (sequence identity: 22.58%; *PSI-Blast E*-value: 1e-054, obtained by *Blast* default options) resides in the same subcellular location: *Cytoplasm* and *Cytoskeleton*. We can see that it is still instrumental to transfer remote homolog knowledge to the target proteins that have no significant homologs. As compared to the method of setting sequence identity threshold, the simplest way of homolog knowledge transfer is to select the top *N* homologs with relatively large identity and low *PSI-Blast E*-value. The top *N* homologs may contain divergent homologs and the divergent homolog is likely to be significant homolog or remote homolog. Proper algorithm design can depress the noise from divergent homologs. *MK-TLM* (Mei, 2012) explicitly computed homolog kernel weight to leverage the contribution of homolog knowledge to the model performance.

Recently, the phenomenon that a protein may exist or move across more than one cellular compartments has aroused more attentions from the computational biologists (Shen and Chou, 2010,2007; Xiao et al., 2011a,2011b; Chou and Shen, 2010). *Plant-mPLoc* (Chou and Shen, 2010), together with the work (Xiao et al., 2011a,2011b), conducted multi-labeling by assigning the cosine-distance based nearest protein's multiple labels to the query protein. Similar to single-label learning model, *Plant-mPLoc* (Chou and Shen, 2010) only reported the percentage that the prediction hit/covered the true labels. In fact, the multi-labeling performance estimation is more complicated than single-label learning. Besides correct subcellular locations (correct *HIT*), a query protein may also be assigned one or more than one wrong subcellular locations (wrong *HIT*) at the same time. Wrong *HIT* rate, also called *Non-target Label Hit Rate* (*NT-LHR*), measures the model's misleading tendency. *Plant-mPLoc* (Chou and Shen, 2010) together with the work (Xiao et al., 2011a,2011b) did not report *NT-LHR* or the performance for novel proteins.

In this paper, we propose a multi-label multi-kernel transfer learning model for plant protein subcellular multi-localization (*MLMK-TLM*). The method further extends our published work *MK-TLM* (multi-kernel transfer learning based on Chou's PseAAC formulation for protein submitochondria localization) (Mei, 2012) to multi-label learning scenario. *MK-TLM* (Mei, 2012) cannot be applied to the scenario when the training protein or test protein has more than one subcellular location. *MLMK-TLM*, with the merits of proper homolog knowledge transfer and easy noise control for novel protein prediction that *MK-TLM* possesses, further proposes a multi-label confusion matrix and adapts one-against-all multi-class probabilistic outputs to the prediction of plant protein subcellular multi-localization. Based on the above improvements, we conduct a comprehensive multi-labeling performance evaluation on plant protein dataset *Plant-mPLoc* (Mei, 2012) to validate *MLMK-TLM*'s effectiveness for novel and multiplex plant protein subcellular localization.

2. Methods

2.1. Transfer learning

As a research field of machine learning community, transfer learning has attracted more and more attentions in recent years (Pan and Yang, 2010). Pan and Yang (2010) reviewed the recent progress of transfer learning modeling and classified transfer learning into three categories based on the way of knowledge transfer: instance-based knowledge transfer, feature-based knowledge transfer and parameter-based knowledge transfer. Dai et al. (2007) proposed an instance-based knowledge transfer learning method that drew in auxiliary data to augment the target training set. The transfer learning model is implemented using *AdaBoost* to lower the weights of the unfavorable auxiliary data that are subjected to different distribution. Dai et al. (2008) proposed a feature-based translated transfer learning method, where a translator was constructed between the text feature space and the image feature space for knowledge transfer from text data to image data. Yang et al. (2009) proposed a parameter-based knowledge transfer learning method, where the knowledge contained in the annotated image of heterogeneous social web was transferred for the target image clustering. Transfer learning can be viewed as a good candidate to bridge two related domains, especially for the learning scenario of heterogeneous feature representation and different distributions.

MK-TLM (Mei, 2012) explicitly computed homolog kernel weight to leverage the contribution of homolog knowledge to the model performance, based on the two sides of biological evolution: evolutionary conservation keeps the subcellular localization patterns of significant/remote homologs, while evolutionary divergence diverges significant/remote homologs to different subcellular locations. In this work, we also adopt non-parametric multiple kernel learning to depress the noise from divergent homologs as *MK-TLM* (Mei, 2012) conducted. The

difference is that we redefine confusion matrix, so that the *GO* kernel weights can be derived by cross validation in multi-label learning scenario.

2.2. GO feature construction

In this work, we adopt the same method of *GO* feature construction as *MK-TLM* (Mei, 2012). We describe the process of *GO* feature construction as below, for self-contained description and integrity. The *GO* terms are extracted from *GOA* database (Ashburner et al., 2000) (77 Release, as of 30 November, 2009), and the homologs are extracted from *SwissProt* 57.3 database (Boeckmann et al., 2003) using *PSI-Blast* (Altschul et al., 1997). Assume there are l *GO* terms of biological process (denoted with *P*), m *GO* terms of molecular function (denoted with *F*) and n *GO* terms of cellular compartments (denoted with *C*). A protein X can be represented with the following six binary feature vectors: $X_P^T, X_F^T, X_C^T, X_P^H, X_F^H, X_C^H$:

$$X_P^T=(x_{P,1},x_{P,2},\ldots,x_{P,l});\ X_F^T=(x_{F,1},x_{F,2},\ldots,x_{F,m});$$
$$X_C^T=(x_{C,1},x_{C,2},\ldots,x_{C,n}) \quad (1)$$

$$X_P^H(x)=(x_{P,1},x_{P,2},\ldots,x_{P,l});\ X_F^H(x)=(x_{F,1},x_{F,2},\ldots,x_{F,m});$$
$$X_C^H(x)=(x_{C,1},x_{C,2},\ldots,x_{C,n}) \quad (2)$$

where the superscript T denotes the target protein and H denotes homolog; $x_{P,i}(i=1,2,\ldots,l)$ denotes 1 if protein X is assigned the l-th *GO* terms of biological process; otherwise, 0. $x_{F,i}(i=1,2,\ldots,m)$ together with $x_{C,i}(i=1,2,\ldots,n)$ is interpreted the same way. The *GO* terms from homologs are aggregated into one homolog feature vector as formula (2) states.

2.3. Non-parametric multiple kernel learning

In this work, we also use non-parametric multiple kernel learning to derive the target kernel weights and the homolog kernel weights as *MK-TLM* (Mei, 2012) conducted. The difference is that we redefine confusion matrix, so that the *GO* kernel weights can be derived by cross validation in multi-label learning scenario. For self-contained description and integrity, the complete kernel weight estimation method is given below. The final kernel is defined as the following linear combination of sub-kernels:

$$K_{MK-TLM}=\sum_{s\in T,H}\sum_{t\in P,F,C} w_t^s\times K_t^S \quad (3)$$

The kernel weights can be derived by simple non-parametric cross validation as *MK-TLM* (Mei, 2012) and *GO-TLM* (Mei et al., 2011) conducted:

$$w_u^v=SE_u^v\times MCC_u^v/\sum_{t\in(T,H)}\sum_{s\in(P,F,C)} SE_s^t\times MCC_s^t,\ u\in\{P,F,C\},\ v\in\{T,H\} \quad (4)$$

where SE_u^v denotes recall rate or sensitivity, and MCC_u^v denotes Matthew's correlation coefficient. Given a training dataset, we divide the training set into k-fold disjoint parts. For each fold cross validation, one part is used as validation set and the other parts are merged as training set to train the combined-kernel SVM. Thus, we can derive a confusion matrix M by evaluating the trained SVM against the validation set. From the confusion matrix M, we can derive the intermediate variables:

$$p_l=M_{l,l}, q_l=\sum_{i=1,i\neq l}^{L}\sum_{j=1,j\neq l}^{L} M_{i,j},\ r_l=\sum_{i=1,i\neq l}^{L} M_{i,l}, s_l=\sum_{j=1,j\neq l}^{L} M_{l,j}$$
$$p=\sum_{l=1}^{L} p_l,\ q=\sum_{l=1}^{L} q_l, r=\sum_{l=1}^{L} r_l,\ s=\sum_{l=1}^{L} s_l \quad (5)$$

where $M_{i,j}$ records the counts that class i are classified to class j; superscript L denotes subcellular locations. Based on the intermediate variables, we can further derive the kernel's *SE* and *MCC* measures:

$$SE=\sum_{l=1}^{L} M_{l,l}/\sum_{i=1}^{L}\sum_{j=1}^{L} M_{i,j}$$
$$MCC=(pq-rs)/\sqrt{(p+r)(p+s)(q+r)(q+s)} \quad (6)$$

In single-label learning scenario, $M_{i,j}(i\neq j)$ records the counts that class i are misclassified to class j, which is not applicable in multi-label learning scenario anymore. Let us borrow the notion of *locative* protein proposed by Shen and Chou (2010) to describe the multi-label confusion matrix. Assume that a protein p is located at two subcellular locations $\{C_1,C_2\}$, i.e., $p\in S_{C1}\wedge p\in S_{C2}$ (S_{C1},S_{C2} denote the protein set that belongs to C_1,C_2, respectively), the notion of *locative* protein means that protein p can be viewed as two different proteins p_1, $p_2(p_1\in S_{C1}\wedge p_2\in S_{C2})$. Now, take p_1 protein as test protein and the trained SVM labels p_1 as follows:

$$C=\arg_l\max(f(p_1,l)|l=1,\ldots,L) \quad (7)$$

where $f(p_1,l)$ denotes the probability that protein p_1 is assigned the label l (see Section 2.4 for how to derive probability outputs). Thus, the *multi-label confusion matrix* can be defined as follows:

$$\begin{cases} M(C_1,C_1)=M(C_1,C_1)+1 C\in C_1,C_2 \\ M(C_1,C)=M(C_1,C)+1 C\notin C_1,C_2 \end{cases} \quad (8)$$

We can see from formula (8) that only if the predicted label of locative protein p_1 hits its true labels $\{C_1,C_2\}$, the prediction is deemed as *correct*; otherwise, the prediction would be deemed as *incorrect*.

As regarding kernels $K_t^s s\in T,H,\ t\in P,F,C$, Gaussian kernel is used here:

$$K_t^s(x,y)=\exp(\gamma|x-y|^2) s\in T,P,\ t\in P,F,C. \quad (9)$$

2.4. Multi-label learning

In this work, we extend *MK-TLM* (Mei, 2012) to multi-learning scenario based on one-against-all multi-class learning and binary SVM probability outputs (Wu et al., 2004; Platt, 1999). Probability outputs tell us the confidence level that a query protein belong to each subcellular location, thus more intuitive and reasonable than ensemble voting (Shen and Chou, 2010,2007) and label transfer of *kNN* nearest neighbor protein (Xiao et al., 2011a,2011b; Chou and Shen, 2010). The posterior class probability for binary *SVM* (Wu et al., 2004; Platt, 1999) is defined below:

$$h_y(x)=p(y|x)=1/(1+e^{Af(x)+B}) \quad (10)$$

where the coefficients A and B can be derived from data by cross validation, and $f(x)$ is the uncalibrated decision value of binary *SVM*.

Actually, the one-against-all multi-class SVM with probability output has been implemented into the LIBSVM tool (http://www.csie.ntu.edu.tw/~cjlin/libsvm/), which can be easily used for multi-label learning. Only if we set the prediction option "-b 1", we can obtain the probability vector that a query protein is predicted to each subcellular location. By setting optimal probability threshold, we can determine the optimal multiple labeling for the query protein.

2.5. Model evaluation and model selection

MK-TLM (Mei, 2012) attempted to conduct a comprehensive survey of the model performance in *optimistic*, *moderate* and *pessimistic* cases, and demonstrated good performance for novel

proteins and those proteins that belong to the protein family we know little about. The *Optimistic* case means the training set and the test set both abound in *GO* information; the *Moderate* case means that the test set contains no *GO* information at all, which can be simulated by removing the test kernels K_P^T,K_F^T,K_C^T; and the *Pessimistic* case means that both the training set and the test set contains no *GO* information at all where the target *GO* information is removed from both the training set and the test set, which can be simulated by removing the training kernels K_P^T,K_F^T,K_C^T and test kernels K_P^T,K_F^T,K_C^T. The *Moderate* case and the *Pessimistic* case borrow the homolog *GO* information for novel protein prediction, while the *Optimistic* case has its own *GO* information (especially the *GO* terms of cellular components) involved in the prediction. In some sense, the cellular component is synonymous to subcellular location in *GOA* database (Ashburner et al., 2000). It seems puzzling for the *Optimistic* case together with the work (Blum et al., 2009; Mei et al., 2011; Shen and Chou, 2010,2007; Xiao et al., 2011a,2011b; Chou and Shen, 2007,2010,2006; Huang et al., 2009,2008; Mei, 2012; Lee et al., 2008; Tung and Lee, 2009) to predict the subcellular location using *GO* terms of cellular components. To clear out the doubts, we enumerate the following application scenarios: (1) the annotated proteins may have their subcellular locations contaminated, for instance, by direct homolog-based annotation transfer, so we can train an outlier- or noise-resistant model to re-estimate the contaminated subcellular locations (Chou and Shen, 2006,2008,2010; Chou and Zhang, 1995); (2) many proteins have more than one subcellular location, and one subcellular location (*GO* term of cellular components) can be used to infer other subcellular locations. Two cases are discussed here: (I) one of the cellular component *GO* terms coincides with the target subcellular location for a test protein, which is the main source of doubts. In such a case, the target subcellular location will be correctly predicted at a higher confidence level, and the doubt that uses what to predict would be cast on the prediction. Actually, the prediction would yield more than one label, and the cellular component *GO* term helps reveal other subcellular locations; (II) none of the cellular component *GO* terms coincide with the target subcellular location. In such a case, the doubt would be relieved, and the cellular component *GO* terms can greatly improve the prediction; (3) if the test proteins have been annotated in terms of molecular function and biological process except cellular component, the *GO* information is also useful for the target subcellular localization prediction. In a word, for a well or partly *GO* annotated protein, the *GO* information helps comprehensively reveal the protein's other subcellular locations, and for novel proteins, we should survey the effectiveness of homolog knowledge transfer in *Moderate* case and *Pessimistic* case.

In this work, the proposed *MLMK-TLM* inherits all *MK-TLM*'s advantages. We should evaluate *MLMK-TLM*'s multi-labeling performance in *Optimistic*, *Moderate* and *Pessimistic* cases. However, the performance evaluation under multi-label learning scenario seems more complicated, because the model performance estimation involves both *singlex* protein (only one subcellular location) and *multiplex* protein (multiple subcellular locations). We should conduct two performance estimation experiments: one experiment is *overall* performance estimation on *locative* dataset, where *multiplex* protein is viewed as multiple *singlex* proteins as *Virus-mPLoc* (Shen and Chou, 2010), *iLoc-Virus* (Xiao et al., 2011a), *Hum-mPLoc* 2.0 (Shen and Chou, 2007) and *Plant-mPLoc* (Chou and Shen, 2010); the other experiment is multi-labeling estimation for *multiplex* proteins. The first experiment is similar to traditional supervised learning estimation except that multi-label confusion matrix is adopted instead (see formulas (7) and (8)). We adopt conventional performance measures for *locative* proteins: Sensitivity (*SE*), Specificity (*SP*), Matthew's correlation coefficient (*MCC*), *Overall MCC*, and *Overall Accuracy*. In the second experiment, cross validation is conducted on *multiplex* proteins only and the *singlex* proteins are always treated as training data. Thus, the whole training set is composed of two parts: *fixed part* from the *singlex* proteins and the *variable part* from the *multiplex* proteins. We adopt two performance measures for *multiplex* proteins: *Label Hit Rate* (*LHR*) and *Non-target Label Hit Rate* (*NT-LHR*). *LHR* measures the model's ability that the prediction covers the true labels, while *NT-LHR* measures the mode's misleading tendency. Assume there are *N* subcellular locations in total, for a test protein with *M* subcellular locations (*target labels*), the prediction would hit 0, 1,…, *M* target labels and 1, 2,…, *N-M* *non-target labels*. We should report *LHR* and *NT-LHR* for each case.

Independent test, *k*-fold cross validation and *Jackknife* test (leave-one-out cross validation) are used for model performance evaluation (Chou and Zhang, 1995). As elucidated in the review (Chou, 2011), *Jackknife* test is deemed the least arbitrary among the three cross-validation methods (Qiu et al., 2009; Esmaeili et al., 2010; Georgiou et al., 2009; Mohabatkar, 2010; Sahu and Panda, 2010; Lin et al., 2011; Wang et al., 2011; Hu et al., 2011; Huang et al., 2011). *MLMK-TLM* is a relatively complex model that requires time-consuming computation for model comparison and model selection, so we adopt 5-fold cross validation instead of leave-one-out cross validation (*LOOCV*) (*Jackknife*) for model performance evaluation. Within each cross validation, we further conduct inner 3-fold cross validation on the training set to derive kernel weights. To reduce computational complexity, we select at most top 15 homologs with lowest *PSI-Blast E*-value using default *Blast* options. The number of homologs is denoted as hyper-parameter $H(1 \le H \le 15)$. The other *MLMK-TLM* model parameters are selected from the set: $\gamma \in \{2^{-3},2^{-2},2^{-1}\}$; $C \in \{2^3,2^4,2^5,2^6,2^7,2^8,2^9,2^{10},2^{11}\}$, where γ is the deviation of Gaussian kernel and C is the regularization parameter that trades off between training error and generalization error.

3. Results

3.1. Dataset

Plant-mPLoc (Chou and Shen, 2010) constructed a quality plant protein dataset. The dataset covers 12 subcellular locations and contains 978 distinct viral proteins, where 904 proteins belong to one subcellular location, 71 to two subcellular locations, and 3 to three subcellular locations. To endow the predictive model with multi-labeling ability, the protein with multiple subcellular locations should be treated as one training example for each subcellular location it belongs to, thus the same protein should be viewed as different proteins within different subcellular locations, referred to as *locative* protein in the literatures (Mei et al., 2011; Shen and Chou, 2010,2007; Xiao et al., 2011a,2011b; Chou and Shen, 2010). There are 1055 *locative* proteins in the dataset (Chou and Shen, 2010). The dataset is a good benchmark for model performance comparison, because none of the proteins have $\ge 25\%$ sequence identity to any other proteins in the same subcellular location. Accordingly, we choose *Plant-mPLoc* (Chou and Shen, 2010) as the baseline model for performance comparison.

3.2. Model performance evaluation

3.2.1. Optimistic case: both training set and test set abound in target GO information

As *MK-TLM* (Mei, 2012) conducted, we also comprehensively survey *MLMK-TLM*'s performance in multi-label learning scenario. The *optimistic* case assumes that both the training set and the test

set abound in target *GO* information. We call this case *MLMK-TLM-I*. As shown in *MLMK-TLM-I* section of Table 1, *MLMK-TLM* achieves 93.74% accuracy and 0.9318 *MCC* on *Plant-mPLoc* (Mei, 2012) protein data, significantly outperforming the baseline *Plant-mPLoc* 63.70% accuracy and the baseline model *iLoc-plant* 71% (Wu et al., 2011). The optimal hyper-parameter setting is ($H=1;\gamma=2^{-3};C=2^{8}$) where $H=1$ means that only one homolog's *GO* information is transferred to the target protein. The high *MCC* value (0.9318) implies that *MLMK-TLM* achieves good predictive balance among the 12 plant subcellular locations. We can see from *MLMK-TLM-I* section of Table 1 that *MLMK-TLM* achieves excellent prediction balance among all the subcellular locations. Especially, *MLMK-TLM* achieves good performance on the smallest *Golgi* ($SP=0.9500$; $SE=0.9048$; $MCC=0.9257$) and *Peroxisome* ($SP=0.8571$; $SE=0.8571$; $MCC=0.8543$).

3.2.2. Moderate case: training set abounds in target GO information while test set contains no target GO information

The most common scenario we encounter may be that we have a plenty of well-annotated training proteins and we need to label some novel proteins at hand. We call the scenario as *moderate* case, referred to as *MLMK-TLM-II*. Novel proteins generally have no *GO* information at all. Once the proposed models work in such a scenario, the performance may not be as *optimistic* as reported. Therefore, experiments should be expressly designed for the *moderate* case to test *MLMK-TLM*'s applicability to novel proteins.

The test procedure for *moderate* case seems more complicated than that for the *optimistic* case, because the proteins in the test set have no target *GO* information. Thus, the three target test kernels $K^{Test}_{T_c}=\exp(\gamma|F^{Test}_{T_c}(x)-F^{Train}_{T_c}|^2), c\in P,F,C$ cannot be derived, because $F^{Test}_{T_c}(x)$ is null (superscript *Test* denotes test set and *Train* denotes training set). For this reason, we substitute the homolog's *GO* feature vector of test protein for its target *GO* feature vector to calculate the test kernel as follows:

$$K^{Test}_{T_c}=\exp(\gamma|F^{Test}_{H_c}(x)-F^{Train}_{T_c}|^2); K^{Test}_{H_c}=\exp(\gamma|F^{Test}_{H_c}(x)-F^{Train}_{H_c}|^2), c\in P,F,C \quad (11)$$

As shown in *MLMK-TLM-II* section of Table 1, *MLMK-TLM* achieves 71.94% accuracy and 0.6952 *MCC* on the benchmark data, substantially underperforming the *optimistic* case with 21.9% accuracy drop, but still outperforming the baseline *Plant-mPLoc* 63.70% accuracy with 8.24% accuracy increase, and equivalent to the baseline model *iLoc-plant* 71% (Wu et al., 2011). As compared to the *optimistic* case, the large performance drop implies that the plant homolog proteins introduce much noise to the target protein. From the result, we can see that the *optimistic* case tends to overestimate the model performance for novel plant protein subcellular localization, and thus it is highly necessary to report the model performance for *moderate* case. The optimal hyper-parameter setting is ($H=2;\gamma=2^{-3};C=2^{7}$), and *MLMK-TLM* still achieves moderate predictive balance among the 12 subcellular locations ($MCC=0.6952$), acceptable on the smallest subcellular locations: *Peroxisome* ($SP=0.7647$; $SE=0.6190$; $MCC=0.6825$) and *Golgi* ($SP=0.7647$; $SE=0.6190$; $MCC=0.6825$), but rather poor on *Plastid* ($SP=0.5000$; $SE=0.1026$; $MCC=0.2149$) and *Vacuole* ($SP=0.6842$; $SE=0.5000$; $MCC=0.5676$). As compared to the baseline *Plant-mPLoc*, *MLMK-TLM* still demonstrates acceptable performance for novel plant protein prediction.

3.2.3. Pessimistic case: both training set and test set contain no target GO information

In this section, we study an extreme case, called *pessimistic* case, where a protein subfamily or species is not *GO*-annotated at all, that is, we know nothing about the protein subfamily or species but only the protein sequence information. Here, we assume that at least one *GO*-annotated homolog can be queried for the target protein, which is not restrictive with the rapid progress of *GOA* database (Ashburner et al., 2000). What interests us is whether the homolog's *GO* information is informative enough to train an effective prediction model for the protein subfamily or species we know little about. If experimental results support the idea, *MLMK-TLM* will gain much wider application. Different from the *optimistic* case and the *moderate* case, the test procedure contains only three homologs's *GO* kernels without target *GO* kernels.

As shown in *MLMK-TLM-III* section of Table 1, *MLMK-TLM* achieves 67.68% accuracy and 0.6490 *MCC* on the benchmark data, a little better than the baseline *Plant-mPLoc* 63.70% accuracy (Chou and Shen, 2010), and a little lower than the baseline model *iLoc-plant* 71% (Wu et al., 2011). The optimal hyper-parameter setting still is ($H=5;\gamma=2^{-3};C=2^{11}$). We can see that *MLMK-TLM* demonstrates relatively poor performance in the *pessimistic* case (see the *underscored* numbers). Nevertheless, the results imply that *MLMK-TLM* is still applicable to newly-discovered plants or plants we know little about.

3.3. Optimal number of homologs

Homolog is a good bridging for knowledge transfer across evolutionarily-related protein subfamilies, super-families or

Table 1
Optimal performance on 1055 plant *locative* protein dataset.

Subcellular location	***Size***	***MLMK-TLM-I* (*optimistic*)** ($H=1;\gamma=2^{-3};C=2^{8}$)			***MLMK-TLM-II* (*moderate*)** ($H=1;\gamma=2^{-3};C=2^{7}$)			***MLMK-TLM-III* (*pessimistic*)** ($H=1;\gamma=2^{-3};C=2^{11}$)		
		SP	***SE***	***MCC***	***SP***	***SE***	***MCC***	***SP***	***SE***	***MCC***
Cell wall	32	0.9063	0.9063	0.9033	0.8000	0.6250	0.6993	0.5366	0.6875	0.5940
Chloroplast	286	0.9549	0.9615	0.9430	0.7492	0.7727	0.6840	0.6677	0.7378	0.6063
Cytoplasm	182	0.9227	0.9176	0.9040	0.5691	0.7692	0.5912	0.6915	0.7143	0.6467
Endoplasmic reticulum	42	0.9487	0.8810	0.9108	0.8065	0.5952	0.6825	0.7368	0.6667	0.6894
Extracell	22	1.0000	0.9545	0.9765	0.6667	0.6364	0.6442	0.6250	0.4545	0.5248
Golgi	21	0.9500	0.9048	0.9257	0.7647	0.6190	0.6825	0.5556	0.4762	0.5054
Mitochondrion	150	0.9539	0.9667	0.9537	0.8163	0.8000	0.7786	0.7606	0.7200	0.7015
Nucleus	152	0.9367	0.9737	0.9474	0.8552	0.8158	0.8097	0.7848	0.8158	0.7680
Peroxisome	21	0.8571	0.8571	0.8543	0.7647	0.6190	0.6825	0.6875	0.5238	0.5932
Cell membrane	56	0.9074	0.8750	0.8851	0.6000	0.6964	0.6260	0.5893	0.5893	0.5673
Plastid	39	0.9444	0.8718	0.9040	0.5000	0.1026	0.2149	0.1905	0.1026	0.1171
Vacuole	52	0.8868	0.9038	0.8899	0.6842	0.5000	0.5676	0.5111	0.4423	0.4515
Overall accuracy/MCC			93.74%/0.9318			71.94%/0.6952			67.68%/0.6490	

species. However, the incorporation of divergent homolog may do harm to the model performance. As *MK-TLM* (Mei, 2012) conducted, we also quantitatively study how much homolog's *GO* information should be transferred to the target protein in multi-label learning scenario. Here we define the homolog search space as 15 homologs with the most significant *E*-value to reduce computational complexity.

As shown in Fig. 1, the optimal number of homologs is 1 for *optimistic* case (*MLMK-TLM-I*), 2 for *moderate* case (*MLMK-TLM-II*), and 5 for *pessimistic* case (*MLMK-TLM-III*). With the incorporation of more homologs, the model performance remains stable for the *optimistic* case, but obviously drops at 8 for the *moderate* case, and slightly decreases at 5 for the *pessimistic* case. For practical prediction of novel plant protein, we can adopt the optimal setting as illustrated in Table 1.

3.4. Kernel weight distribution

The *GO* kernel weights are evaluated using 3-fold cross validation according to formulas (3)–(6). The *optimistic* case and the *moderate* case contain six *GO* kernels, while the *pessimistic* case contains only three homolog *GO* kernels because the target *GO* information is missing for novel proteins.

As shown in Fig. 2, the *x* axis denotes the six *GO* kernels, where *T* denotes target, *H* denotes homolog, *F*, *C* and *P* denote the three aspects of gene ontology (molecular function, cellular component and biological process, respectively). We can see that both the *optimistic* case and the *moderate* case have similar kernel weight distributions on the benchmark dataset, while the *pessimistic* case is similar to the homolog *GO* kernel weight distribution of the *optimistic* case and the *moderate* case (see the latter part of curve in Fig. 2). No matter the target *GO* kernels or the homolog *GO* kernels, *C* (cellular component) demonstrates much higher kernel weight. For the *optimistic* case and the *moderate* case, the target *GO* kernels make a much more contribution to the model performance than the homolog *GO* kernels (compare the former half part and the latter half part of the curve in Fig. 2). The results imply that the homolog *GO* information carries much noise than the *GO* information, which partly explains why the *moderate* and the *pessimistic* cases demonstrate a large performance drop as compared to the *optimistic* case (see Table 1).

3.5. Multi-labeling estimation

In this section, we expressly survey *MLMK-TLM*'s multi-labeling performance on multiplex plant proteins. Different from the performance measures *locative* plant proteins as stated in Section 3.3, we propose two criteria for multi-labeling performance estimation: *Label Hit Rate* (*LHR*) and *Non-target Label Hit Rate* (*NT-LHR*). *LHR* measures *MLMK-TLM*'s multi-labeling performance, while *NT-LHR* measures *MLMK-TLM*'s misleading tendency. For proteins with 2 subcellular locations, we evaluate 0-*LHR*, 1-*LHR* and 2-*LHR*; for proteins with 3 subcellular locations, evaluate 0-*LHR*, 1-*LHR*, 2-*LHR* and 3-*LHR*, accordingly. Intuitively, if we set low probability threshold, the prediction would correctly cover more true labels (high *LHR*) but yield more misleading labels (high *NT-LHR*) simultaneously. Optimal probability threshold can achieve good balance between *LHR* and *NT-LHR*. Here, we select the optimal probability threshold from {0.06,0.07,0.08,0.09,0.10,0.15,0.2} and the optimal probability threshold is 0.08. *Plant-mPLoc* dataset (Chou and Shen, 2010) has 12 subcellular locations, thus the prediction would yield 0–10 non-target labels (wrong label) for those proteins with 2 subcellular locations, and 0–9 non-target labels for those proteins with 3 subcellular locations. The experiments show that most of the probability masses are assigned to 3–4 labels, and the probability mass of other labels approximate to zero, thus the 4–10- *NT-LHR* is nearly to *zero* and only 1–3-*NT-LHR* is reported here.

Table 2.1
Multi-labeling evaluation for the *optimistic* case.

Multiplex locations	***Size***	***Label Hit Rate* (*LHR*)**				***Non-target Label Hit Rate* (*NT-LHR*)**		
		0	***1***	***2***	***3***	***1***	***2***	***3***
2	71	*1.41%*	*21.13%*	**77.46%**	0.0000	0.0000	16.90%	1.41%
3	3	*0.0000*	*0.0000*	*66.67%*	**33.33%**	0.0000	33.33%	0.0000

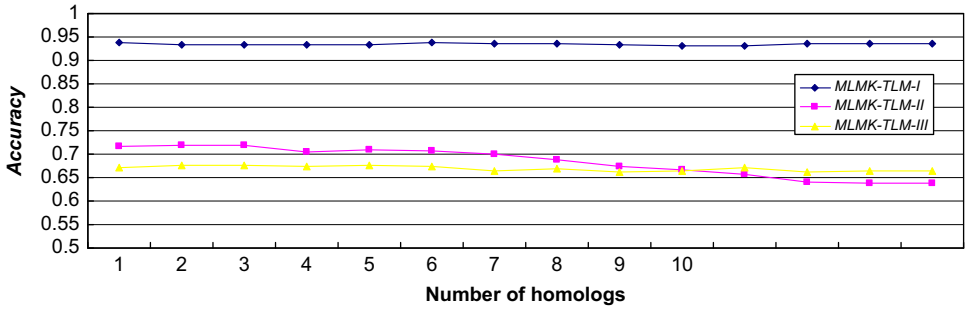

Fig. 1. Performance on 1055 plant *locative* protein dataset with varying homologs.

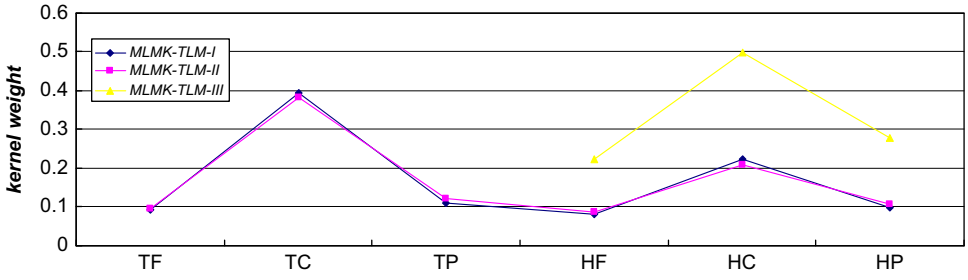

Fig. 2. Kernel weight estimation on 1055 plant *locative* protein dataset.

Table 2.2
Multi-labeling evaluation for the *moderate* case.

Multiplex locations	*Size*	*Label Hit Rate (LHR)*				*Non-target Label Hit Rate (NT-LHR)*		
		0	*1*	*2*	*3*	*1*	*2*	*3*
2	71	*7.04%*	*25.35%*	**67.61%**	0.0000	0.0000	12.68%	16.90%
3	3	*0.0000*	*66.67%*	*33.33%*	**0.0000**	0.0000	33.33%	0.0000

Table 2.3
Multi-labeling evaluation for the *pessimistic* case.

Multiplex locations	*Size*	*Label Hit Rate (LHR)*				*Non-target Label Hit Rate (NT-LHR)*		
		0	*1*	*2*	*3*	*1*	*2*	*3*
2	71	*2.82%*	*21.13%*	**76.06%**	0.0000	0.0000	28.17%	18.31%
3	3	*0.0000*	*0.0000*	*100%*	**0.0000**	0.0000	0.0000	0.0000

As shown in the former part of Table 2.1 through Table 2.3, *MLMK-TLM* correctly covers all the true labels with 77.46% accuracy for the multiplex proteins with 2 subcellular locations and 33.33% accuracy for the multiplex proteins with 3 subcellular locations in the *optimistic* case (number in *bold font* in Table 2.1); 67.61% and 0 in the *moderate* case (number in *bold font* in Table 2.2); 76.06% and 0 in the *pessimistic* case (number in *bold font* in Table 2.3). The results are promising for multiplex proteins with 2 subcellular locations. The other *LHR* measures (numbers *in italic* in Tables 2.1–2.3) are also promising. The 0-*LHR* is acceptably low (2-label 1.41%, 3-label 0 for the *optimistic* case; 2-label 7.04%, 3-label 0 for the *moderate* case; and 2-label 2.82%, 3-label 0 for the *pessimistic* case).

As shown in the latter part of Table 2.1 through Table 2.3, *MLMK-TLM* achieves acceptable 16.90% *2-NT-LHR* for multiplex proteins with 2 subcellular locations and 33.33% 2-*NT-LHR* for multiplex proteins with 3 subcellular locations in the *optimistic* case (*underscored* numbers in Table 2.1); 12.68% 2-*NT-LHR* for multiplex proteins with 2 subcellular locations and 33.33% 2-*NT-LHR* for multiplex proteins with 3 subcellular locations in the *moderate* case (*underscored* numbers in Table 2.2); 28.17% 2-*NT-LHR* for multiplex proteins with 2 subcellular locations and 0 2-*NT-LHR* for multiplex proteins with 3 subcellular locations in the *pessimistic* case (*underscored* numbers in Table 2.3). We can see that *MLMK-TLM* shows acceptable misleading tendency, which is much lower than *LHR* measures. Interestingly, the work *iLoc-plant* (Chou et al., 2012; Wu et al., 2011) investigated the *perfect label match rate* (exact label match) to conduct a more stringent survey of model performance, and achieved 68.1% accuracy. Here, we investigate both the *target label hit rate* (*LHR*) and the *non-target label hit rate* (*NT-LHR*), where high *LHR* and low *NT-LHR* would imply high *perfect label match rate*. As compared to the nearest neighbor based accumulation-label method (Wu et al., 2011; Chou and Shen, 2010), the label-probability based *MLMK-TLM* persuades the biologists accepting the predicted labels with a certain confidence level.

4. Conclusions

In this paper, we propose a multi-label multi-kernel transfer learning model for plant protein subcellular multi-localization (*MLMK-TLM*), an extension to our published *MK-TLM* model (Mei, 2012). By redefinition of confusion matrix and one-against-all multi-class probabilistic outputs, we adapt *MK-TLM* to multi-label learning scenario. With the merits of proper homolog knowledge transfer and easy noise control for novel protein prediction that *MK-TLM* possesses, *MLMK-TLM* proposes two multi-labeling performance measures: *Label Hit Rate* (*LHR*) and *Non-target Label Hit Rate* (*NT-LHR*), based on which the model performance for plant protein subcellular multi-localization in the *optimistic*, *moderate* and *pessimistic* cases can be comprehensively surveyed. The experiments on the benchmark dataset show that *MLMK-TLM* outperforms the baseline model and achieves acceptable model performance for novel and multiplex plant protein subcellular localization.

References

Altschul, S., Madden, T., Schaffer, A., Zhang, J., Zhang, Z., Miller, W., Lipman, D., 1997. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res. 25, 3389–3402.

Ashburner, M., et al., 2000. Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. Nat. Genet. 25, 25–29.

Blum, T., Briesemeister, S., Kohlbacher, O., 2009. MultiLoc2: integrating phylogeny and gene ontology terms improves subcellular protein localization prediction. BMC Bioinform. 10, 274.

Boeckmann, B., et al., 2003. The SWISS-PROT protein knowledgebase and its supplement TrEMBL. Nucleic Acids Res. 31, 365–370.

Chou, K.C., 2001. Prediction of protein cellular attributes using pseudo amino acid composition. Proteins: Struct. Funct. Genet. (Erratum: ibid., 2001, vol. 44, 60) 43, 246–255.

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

Chou, K.C., Cai, Y.D., 2002. Using functional domain composition and support vector machines for prediction of protein subcellular location. J. Biol. Chem. 277, 4575–45769.

Chou, K.C., Elrod, D.W., 1999. Protein subcellular location prediction. Protein Eng. 12, 107–118.

Chou, K.C., Shen, H.B., 2006. Hum-PLoc: a novel ensemble classifier for predicting human protein subcellular localization. Biochem. Biophys. Res. Commun. 347, 150–157.

Chou, K.C., Shen, H.B., 2007. Review: recent progresses in protein subcellular location prediction. Anal. Biochem. 370, 1–16.

Chou, K.C., Shen, H.B., 2007. Large-scale plant protein subcellular location prediction. J. Cell. Biochem. 100, 665–678.

Chou, K.C., Shen, H.B., 2008. Cell-PLoc: a package of Web servers for predicting subcellular localization of proteins in various organisms. Nat. Protocols 3, 153–162.

Chou, K.C., Shen, H.B., 2009. Review: recent advances in developing web-servers for predicting protein attributes. Nat. Sci. 2, 63–92. (Openly accessible from).

Chou, K.C., Shen, H.B., 2010. Cell-PLoc 2.0: an improved package of web-servers for predicting subcellular localization of proteins in various organisms. Nat. Sci. 2, 1090–1103. (Openly accessible from)〈http://www.scirp.org/journal/NS/〉.

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

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

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

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

Dai W., Yang Q., Xue G., Yu Y. Boosting for Transfer Learning. In: Proceedings of the 24th International Conference on Machine Learning, 2007.

Dai, W., Chen, Y., Xue, G., Yang Q., Yu, Y. Translated learning: transfer learning across different feature spaces. In: Proceedings of the NIPS 2008.

Du, P., Li, Y., 2006. Prediction of protein submitochondria locations by hybridizing pseudo-amino acid composition with various physicochemical features of segmented sequence. BMC Bioinform. 7, 518.

Du, P., Cao, S., Li, Y., 2009. SubChlo: Predicting protein subchloroplast locations with pseudo-amino acid composition and the evidence-theoretic K-nearest neighbor (ET-KNN). J. Theor. Biol. 261, 330–335.

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

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

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

Hu, L.L., Huang, T., Cai, Y.D., Chou, K.C., 2011. Prediction of body fluids where proteins are secreted into based on protein interaction network. PLoS ONE 6, e22989.
Huang, T., Chen, L., Cai, Y.D., Chou, K.C., 2011. Classification and analysis of regulatory pathways using graph property, biochemical and physicochemical property, and functional property. PLoS ONE 6, e25297.
Huang, W., Tunq, C., Ho, S., Hwang, S., Ho, S., 2008. ProLoc-GO: utilizing informative gene ontology terms for sequence-based prediction of protein subcellular localization. BMC Bioinform. 9, 80.
Huang, W., Tung, C., Huang, S., Ho, S., 2009. Predicting protein subnuclear localization using GO-amino-acid composition features. BioSystems.
Lee, K., Chuang, H., Beyer, A., Sung, M., Huh, W., Lee, B., Ideker, T., 2008. Protein networks markedly improve prediction of subcellular localization in multiple eukaryotic species. Nucleic Acids Res. 36 (20), e136.
Lin, W.Z., Fang, J.A., Xiao, X., Chou, K.C., 2011. iDNA-Prot: identification of DNA binding proteins using random forest with grey model. PLoS ONE 6, e24756.
Mei, S., 2012. Multi-kernel transfer learning based on Chou's PseAAC formulation for protein submitochondria localization. J. Theor. Biol. 293, 121–130.
Mei, S., Wang, Fei, 2010. Amino acid classification based spectrum kernel fusion for protein subnuclear localization. BMC Bioinform. 11 (Suppl. 1), S17.
Mei, S., Wang, F., Zhou, S., 2011. Gene ontology based transfer learning for protein subcellular localization. BMC Bioinform. 12, 44.
Mohabatkar, H., 2010. Prediction of cyclin proteins using Chou's pseudo amino acid composition. Protein Pept. Lett. 17, 1207–1214.
Nakai, K., 2000. Protein sorting signals and prediction of subcellular localization. Adv. Protein Chem. 54, 277–344.
Nakashima, H., Nishikawa, K., 1994. Discrimination of intracellular and extracellular proteins using amino acid composition and residue-pair frequencies. J. Mol. Biol. 238, 54–61.
Pan, S., Yang, Q.A., 2010. Survey on transfer learning. IEEE Trans. Knowl. Data Eng. 22 (10), 1345–1359.
Pierleoni, A., Luigi, P., Fariselli, P., Casadio, R., 2006. BaCelLo: a balanced subcellular localization predictor. Bioinformatics 22 (14), e408–e416.
Platt, J. Probabilistic outputs for support vector machines and comparison to regularized likelihood methods. In: Advances in Large Margin Classifiers. MIT Press, 1999.
Qiu, J.D., Huang, J.H., Liang, R.P., Lu, X.Q., 2009. Prediction of G-protein-coupled receptor classes based on the concept of Chou's pseudo amino acid composition: an approach from discrete wavelet transform. Anal. Biochem. 390, 68–73.
Sahu, S.S., Panda, G., 2010. A novel feature representation method based on Chou's pseudo amino acid composition for protein structural class prediction. Comput. Biol. Chem. 34, 320–327.
Shen, H.B., Chou, K.C., 2007. Hum-mPLoc: an ensemble classifier for large-scale human protein subcellular location prediction by incorporating samples with multiple sites. Biochem. Biophys. Res. Commun. 355, 1006–1011.
Shen, H.B., Chou, K.C., 2010. Virus-mPLoc: a fusion classifier for viral protein subcellular location prediction by incorporating multiple sites. J. Biomol. Struct. Dyn. 28, 175–186.
Tung, T., Lee, D., 2009. A method to improve protein subcellular localization prediction by integrating various biological data sources. BMC Bioinform. 10 (Suppl. 1), S43.
Wang, P., Xiao, X., Chou, K.C., 2011. NR-2L: a two-level predictor for identifying nuclear receptor subfamilies based on sequence-derived features. PLoS ONE 6, e23505.
Wu, T., Lin, C., Weng, R., 2004. Probability estimates for multi-class classification by pairwise coupling. J. Mach. Learn. Res. 5, 975–1005.
Wu, Z.C., Xiao, X., Chou, K.C., 2011. iLoc-Plant: a multi-label classifier for predicting the subcellular localization of plant proteins with both single and multiple sites. Mol. BioSyst. 7, 3287–3297.
Wu, Z.C., Xiao, X., Chou, K.C., 2012. iLoc-Gpos: a multi-layer classifier for predicting the subcellular localization of singleplex and multiplex gram-positive bacterial proteins. Protein Pept. Lett. 19, 4–14.
Xiao, X., Wu, Z.C., Chou, K.C., 2011a. iLoc-Virus: a multi-label learning classifier for identifying the subcellular localization of virus proteins with both single and multiple sites. J. Theor. Biol. 284, 42–51.
Xiao, X., Wu, Z.C., Chou, K.C., 2011b. A multi-label classifier for predicting the subcellular localization of gram-negative bacterial proteins with both single and multiple sites. PLoS ONE 6, e20592.
Yang, Q., Chen, Y., Xue, G., Dai, W., Yu, Y. Heterogeneous transfer learning for image clustering via the social Web. In: Proceedings of the 47th Annual Meeting of the ACL and the 4th IJCNLP of the AFNLP 2009, pp. 1–9.
Zhou, G.P., Doctor, K., 2003. Subcellular location prediction of apoptosis proteins. Proteins: Struct. Funct. Genet. 50, 44–48.