An Ensemble Classifier with Random Projection for Predicting Multi-label Protein Subcellular Localization

Shibiao Wan*, Man-Wai Mak*, Bai Zhang†, Yue Wang‡, Sun-Yuan Kung§
*Dept. of Electronic and Information Engineering, The Hong Kong Polytechnic University, Hung Hom, Hong Kong SAR, China (email: shibiao.wan@connect.polyu.hk and enmwmak@polyu.edu.hk)
†Dept. of Pathology, Johns Hopkins Medical Institutions, Baltimore, Maryland, USA (email: baizhang@jhu.edu)
‡Bradley Dept. of Electrical and Computer Engineering, Virginia Polytechnic Institute and State University, Arlington, Virginia, USA (email: yuewang@vt.edu)
§Dept. of Electrical Engineering, Princeton University, Princeton, New Jersey, USA (email: kung@princeton.edu)

***Abstract*—In protein subcellular localization prediction, a predominant scenario is that the number of available features is much larger than the number of data samples. Among the large number of features, many of them may contain redundant or irrelevant information, causing the prediction systems suffer from overfitting. To address this problem, this paper proposes a dimensionality-reduction method that applies random projection (RP) to construct an ensemble multi-label classifier for predicting protein subcellular localization. Specifically, the frequencies of occurrences of gene-ontology terms are used as feature vectors, which are projected onto lower-dimensional spaces by random projection matrices whose elements conform to a distribution with zero mean and unit variance. The transformed low-dimensional vectors are classified by an ensemble of one-vs-rest multi-label support vector machine (SVM) classifiers, each corresponding to one of the RP matrices. The scores obtained from the ensemble are then fused for making the final decision. Experimental results on two recent datasets suggest that the proposed method can reduce the dimensions by six folds and remarkably improve the classification performance.**

***Index Terms*—Random projection; Dimension reduction; Multi-label classification; Protein subcellular localization; Support vector machines.**

I. Introduction

Protein subcellular localization is to predict in which part(s) of a cell a protein resides. In recent years, protein subcellular localization has received tremendous attention due to its vitally important roles in elucidating protein functions, identifying drug targets, and so on [1]. Understanding the relationship between aberrantly localized proteins and human diseases is also important for developing therapeutic intervention [2]. Computational methods are required to replace time-consuming and laborious wet-lab methods for predicting the subcellular locations of proteins.

Conventional methods for subcellular-localization prediction can be roughly divided into sequence-based methods [3]–[5] and annotation-based methods [6]–[9]. No matter using the sequence-based features or annotation-based features, a predominant scenario is that the dimension of available features is much larger than the number of training samples. For example, Lee et. al. [10] used amino-acid sequence-based features, whose dimension (11,992-dim) is remarkably larger than the number of proteins (3017); Xiao et. al. [11] used the Gene Ontology (GO)[1] information as the features and the dimension of features was 11,118 while the number of proteins was only 207. It is highly expected that the high-dimensional features contain redundant or irrelevant information, causing overfitting and worsening the prediction performance.

Among the existing methods mentioned above, it has been demonstrated that methods based on GO are superior [12]. In particular, all of the state-of-the-art methods, such as Virus-mPLoc [13], iLoc-Virus [11], KNN-SVM ensemble classifier [14], and mGOASVM [15], use GO information as features.

The GO comprises three orthogonal taxonomies whose terms are used to describe the attributes of cellular components, biological processes, and molecular functions for a gene product. The GO terms in each taxonomy are organized within a directed acyclic graph (DAG). Many efforts have been made to discover the semantic similarity (SS) between GO terms [16]–[19]. The hierarchical structure of the GO terms as well as the semantic similarity over GO explicitly demonstrates that there are correlations among the GO features, which in turn suggests the presence of the redundant, irrelevant or even detrimental information when the GO information is used for classification.

This paper proposes an ensemble classifier based on random projection (RP) for predicting subcellular localization of multi-label proteins. In the past decades, RP has emerged as a powerful method for dimension reduction in various applications, such as preprocessing text data [20], indexing audio documents [21], processing images [22], learning high-dimensional Gaussian mixture models [23], etc. By using RP, the high dimensional feature vectors are transformed into a much lower-dimensional vectors, which contain less redundant, irrelevant or even detrimental information that might deteriorate classification performance. To make the classifiers more robust, it is necessary to perform random projection of the feature vectors several times. The resulting projected vectors are then presented to an ensemble of one-vs-rest

[1]http://www.geneontology.org

multi-label SVM classifiers. Results on two recent benchmark datasets in protein subcellular localization demonstrate that the proposed ensemble classifier substantially outperforms the state-of-the-art predictors and that RP is significantly better than the conventional dimension-reduction and feature-selection methods (such as PCA and RFE-SVM [24]) for subcellular localization. This paper also demonstrates that only 3 to 4 applications of RP will be sufficient to construct an ensemble classifier with input dimension that is one-sixth of that of the full-feature classifiers, while at the same time improves the classification performance.

II. Feature Extraction

The subcellular localization predictors use GO information as the features, which has been demonstrated to be superior over other features [12], [15].

A. Legitimacy of Using GO Information as Features

Despite their good performance, GO-based methods have received some criticisms from the research community. The main argument of these criticisms is that the cellular component GO terms already have the cellular component categories, i.e., if the GO terms are known, the subcelluar locations will also be known. The prediction problem can therefore be easily solved by creating a lookup table using the cellular component GO terms as the keys and the cellular component categories as the hashed values. Such a naive solution, however, will lead to very poor prediction performance, as demonstrated and explained in our previous studies [15], [25]. A number of studies [26]–[28] by other groups also strongly support the legitimacy of using GO information for subcellular localization. For example, as suggested by [28], the good performance of GO-based methods is due to the high representation power of the GO space as compared to the Euclidean feature spaces used by the conventional sequence-based methods.

B. Retrieval of GO Terms

As shown in Fig. 1, the predictor described in this paper can use either protein accession numbers (AC) or protein sequences as input. For proteins with known ACs, their respective GO terms are retrieved from the Gene Ontology Annotation (GOA) database[2] using the ACs as the searching keys. For a protein without an AC, its amino acid sequence is presented to BLAST [29] to find its homologs, whose ACs are then used as keys to search against the GOA database.

While the GOA database allows us to associate the AC of a protein with a set of GO terms, for some novel proteins, neither their ACs nor the ACs of their top homologs have any entries in the GOA database; in other words, no GO terms can be retrieved by their ACs or the ACs of their top homologs. In such case, the ACs of the homologous proteins, as returned from BLAST search, will be successively used to search against the GOA database until a match is found. In case where no GO terms can be retrieved by the ACs or even by the ACs of all the homologs, back-up methods that rely on other features, such as pseudo-amino-acid composition [5] should be used. Fortunately, with the rapid progress of the GOA database [30], it is reasonable to assume that the homologs of the query proteins can retrieve at least one GO term [8]. Thus, it is rarely necessary to use back-up methods to handle the situation where no GO terms can be found.

[2]http://www.ebi.ac.uk/GOA

C. Construction of GO Vectors

Given a dataset, the GO terms of all of its proteins are retrieved by using the procedure described in Section II-B. Then, the number of distinct GO terms corresponding to the dataset is determined. Suppose T distinct GO terms are found; these GO terms form a GO Euclidean space with T dimensions. For each sequence in the dataset, a GO vector is constructed by matching its GO terms to all of the T GO terms. Unlike the conventional 1-0 value [7], in this work, term-frequency [25] is used to construct the GO vectors. Similar to the 1-0 value approach, a protein is represented by a point in a Euclidean space. However, unlike the 1-0 approach, the term-frequency approach uses the number of occurrences of individual GO terms as the coordinates. Specifically, the GO vector $\mathbf{p}_i$ of the i-th protein is defined as:

$$\mathbf{p}_i = [b_{i,1}, \cdots, b_{i,j}, \cdots, b_{i,T}]^{\mathsf{T}}, b_{i,j} = \begin{cases} f_{i,j} & \text{, GO hit} \\ 0 & \text{, otherwise} \end{cases} \tag{1}$$

where $f_{i,j}$ is the number of occurrences of the j-th GO term (term-frequency) in the i-th protein sequence. The rationale is that the term-frequencies may also contain important information for classification and therefore should not be quantized to either 0 or 1. Note that $b_{i,j}$'s are analogous to the term-frequencies commonly used in document retrieval.

III. Random Projection

The key idea of RP arises from Johnson-Lindenstrauss lemma [31]:

Lemma 1. (Johnson and Lindenstrauss [31]). *Given $\epsilon > 0$, a set $\mathcal{X}$ of N points in $\mathcal{R}^T$, and a positive integer d such that $d \geq d_0 = \mathcal{O}(\log N/\epsilon^2)$, there exists $f : \mathcal{R}^T \to \mathcal{R}^d$ such that*

$$(1-\epsilon)\|u-v\|^2 \leq \|f(u)-f(v)\|^2 \leq (1+\epsilon)\|u-v\|^2$$

for all $u, v \in \mathcal{X}$.

The lemma suggests that if points in a high-dimensional space are projected onto a randomly selected subspace of suitable dimension, the distances between the points are approximately preserved. A proof can be found in [32].

Specifically, the original T-dimensional data is projected onto a d-dimensional ($d \ll T$) subspace, using a $d \times T$ random matrix $\mathbf{R}$ whose columns are unit lengths. In our case, for the i-th protein, the GO vector $\mathbf{p}_i$ can be projected as:

$$\mathbf{p}_i^{RP} = \frac{1}{\sqrt{d}}\mathbf{R}\mathbf{p}_i, \tag{2}$$

where $1/\sqrt{d}$ is a scaling factor, $\mathbf{p}_i^{RP}$ is the projected vector after RP, and $\mathbf{R}$ is a random $d \times T$ matrix.

The choice of the random matrix $\mathbf{R}$ has been studied extensively. Practically, as long as the elements $r_{h,j}$ of $\mathbf{R}$

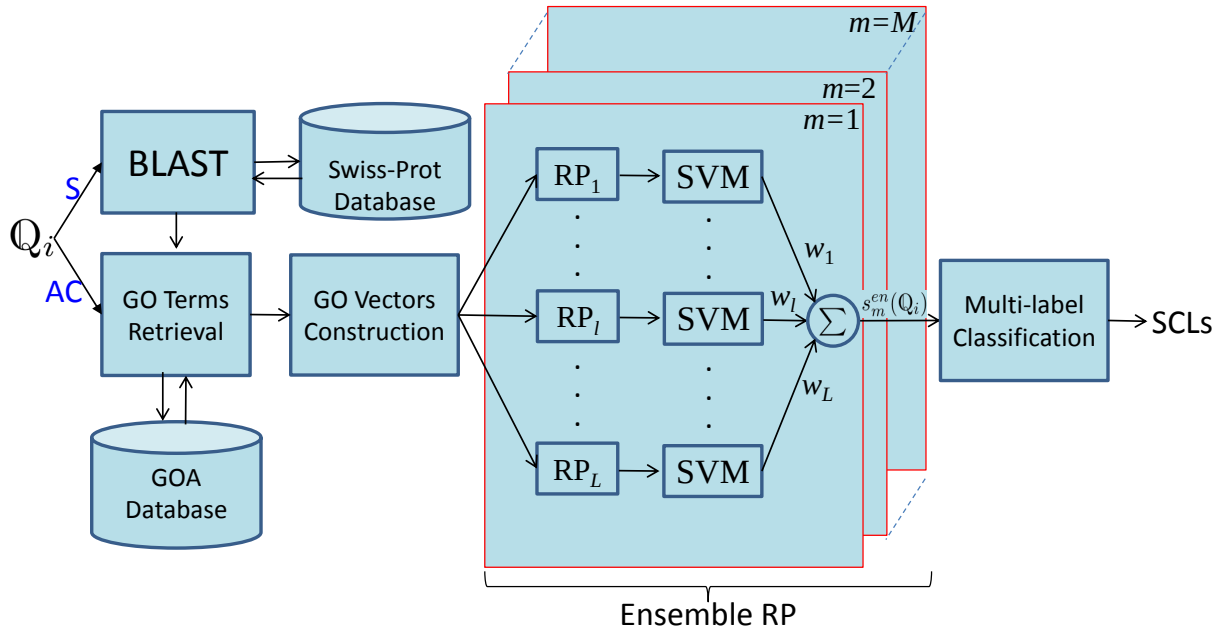


Fig. 1. Flowchart of RP-SVM/RP-AD-SVM. $\mathbb{Q}_i$: the i-th query protein; *S*: protein sequence; *AC*: protein accession number; *RP*: random projection; *SVM*: SVM scoring (Eq. 4); *Ensemble RP*: ensemble random projection; w_1, w_l *and* w_L: the 1-st, l-th and L-th weights in Eq. 5; $s_m^{en}(\mathbb{Q}_i)$: the ensemble score in Eq. 5; *SCLs*: subcellular location(s).

conforms to any distribution with zero mean and unit variance, $\mathbf{R}$ will give a mapping that satisfies the Johnson-Lindenstrauss lemma [22]. For computational simplicity, we adopted a simple distribution proposed by Achlioptas [33] for the elements $r_{h,j}$ as follows:

$$r_{h,j} = \sqrt{3} \times \begin{cases} +1 & \text{with probability 1/6,} \\ 0 & \text{with probability 2/3,} \\ -1 & \text{with probability 1/6.} \end{cases} \quad (3)$$

As can be seen, Eq. 3 conforms to a distribution with zero mean and unit variance.

IV. Ensemble Multi-label Classifier

The projected GO vectors obtained from Eq. 2 are used for training multi-label one-vs-rest SVMs. Specifically, for an M-class problem (here M is the number of subcellular locations), M independent binary SVMs are trained, one for each class. Denote the GO vector created by using the true AC of the i-th query protein as $\mathbf{q}_{i,0}$ and the GO vector created by using the accession number of the k-th homolog as $\mathbf{q}_{i,k}$, $k \in \{1, \ldots, k_{\max}\}$, where $k_{\max}$ is the number of homologs retrieved by BLAST with the default parameter setting. By Eq. 2, we obtained the corresponding projected vectors $\mathbf{q}_{i,0}^{RP}$ and $\mathbf{q}_{i,k}^{RP}$, respectively. Then, given the i-th query protein $\mathbb{Q}_i$, the score of the m-th SVM is:

$$s_m(\mathbb{Q}_i) = \sum_{r \in \mathcal{S}_m} \alpha_{m,r} y_{m,r} K(\mathbf{p}_r^{RP}, \mathbf{q}_{i,k}^{RP}) + b_m \quad (4)$$

where $k \in \{0, 1, \ldots, k_{\max}\}$, $\mathcal{S}_m$ is the set of support vector indexes corresponding to the m-th SVM, $y_{m,r} \in \{-1, +1\}$ are the class labels, $\alpha_{m,r}$ are the Lagrange multipliers, $K(\cdot, \cdot)$ is a kernel function; here, the linear kernel is used. Note that $\mathbf{p}_r^{RP}$'s in Eq. 4 represents the projected GO training vectors, which may include the projected GO vectors created by using the true AC of the training sequences or their homologous ACs.

Since $\mathbf{R}$ is a random matrix, the scores in Eq. 4 for each application of RP will be different. To construct a robust classifier, we fused the scores for several applications of RP and obtained an ensemble classifier, whose ensemble score of the m-th SVM for the i-th query protein is given as follows:

$$s_m^{en}(\mathbb{Q}_i) = \sum_{l=1}^{L} w_l \cdot s_m^{(l)}(\mathbb{Q}_i), \quad (5)$$

where $\sum_{l=1}^{L} w_l = 1$, $s_m^{(l)}(\mathbb{Q}_i)$ represents the score of the m-th SVM for the i-th protein via the l-th application of RP, L is the total number of applications of RP, and $\{w_l\}_{l=1}^{L}$ are the weights. For simplicity, here we set $w_l = 1/L, l = 1, \ldots, L$. We refer L as 'ensemble size' in the sequel. Unless stated otherwise, the ensemble size was set to 10 in our experiments, i.e., $L = 10$. Note that instead of mapping the original data into an Ld-dim vector, the ensemble RP projects it into L d-dim vectors.

V. Multi-label Classification

To predict the subcellular locations of datasets containing both single-label and multi-label proteins, a decision scheme for multi-label SVM classifiers should be used. Unlike the single-label problem where each protein has one predicted label only, a multi-label protein should have more than one predicted labels. In this paper, we evaluated two decision schemes. The first decision scheme is the same as that used in mGOASVM [15]. In this scheme, the predicted subcellular location(s) of the i-th query protein are given by:

$$\mathcal{M}^*(\mathbb{Q}_i) = \begin{cases} \bigcup_{m=1}^{M} \{m : s_m^{en}(\mathbb{Q}_i) > 0\}, \text{ where } \exists\ s_m^{en}(\mathbb{Q}_i) > 0\ ; \\ \arg\max_{m=1}^{M} s_m^{en}(\mathbb{Q}_i), \text{ otherwise.} \end{cases} \tag{6}$$

The second decision scheme is an adaptive decision improved upon the first one. This decision scheme is the same as that used in AD-SVM [34]. In this scheme, the predicted subcellular location(s) of the i-th query protein are given by: If $\exists\ s_m^{en}(\mathbb{Q}_i) > 0$,

$$\mathcal{M}(\mathbb{Q}_i) = \bigcup_{m=1}^{M} \{\{m : s_m^{en}(\mathbb{Q}_i) > 1.0\} \cup \{m : s_m^{en}(\mathbb{Q}_i) \geq f(s_{\max}(\mathbb{Q}_i))\}\} \tag{7}$$

otherwise,

$$\mathcal{M}(\mathbb{Q}_i) = \arg\max_{m=1}^{M} s_m^{en}(\mathbb{Q}_i). \tag{8}$$

In Eq. 7, $f(s_{\max}(\mathbb{Q}_i))$ is a function of $s_{\max}(\mathbb{Q}_i)$, where $s_{\max}(\mathbb{Q}_i) = \max_{m=1}^{M} s_m^{en}(\mathbb{Q}_i)$. In this work, we used a linear function as follows:

$$f(s_{\max}(\mathbb{Q}_i)) = \theta s_{\max}(\mathbb{Q}_i), \tag{9}$$

where $\theta \in [0.0, 1.0]$ is a parameter that was optimized to achieve the best performance.

For ease of comparison, we refer to the proposed ensemble classifier with the first and the second decision scheme as RP-SVM and RP-AD-SVM, respectively. Fig. 1 illustrates the whole prediction process for RP-SVM and RP-AD-SVM. If we use the first decision scheme for *multi-label classification*, the diagram represents RP-SVM; if we use the second decision scheme, it represents RP-AD-SVM.

VI. Datasets and Performance Metrics

In this paper, a virus dataset [11], [13] and a plant dataset [35] were used to evaluate the performance of the proposed predictors. The virus and the plant datasets were created from Swiss-Prot 57.9 and 55.3, respectively. The virus dataset contains 207 viral proteins distributed in 6 locations. Of the 207 viral proteins, 165 belong to one subcellular locations, 39 to two locations, 3 to three locations and none to four or more locations. This means that about 20% of the proteins in the dataset are located in more than one subcellular location. The plant dataset contains 978 plant proteins distributed in 12 locations. Of the 978 plant proteins, 904 belong to one subcellular locations, 71 to two locations, 3 to three locations and none to four or more locations. The sequence identity of both datasets was cut off at 25%.

Compared to traditional single-label classification, multi-label classification requires more complicated performance metrics to better reflect the multi-label capabilities of classifiers. These measures include *Accuracy, Precision, Recall, F1-score (F1)* and *Hamming Loss (HL)*. Specifically, denote $\mathcal{L}(\mathbb{Q}_i)$ and $\mathcal{M}(\mathbb{Q}_i)$ as the true label set and the predicted label set for the i-th protein $\mathbb{Q}_i$ ($i = 1, \ldots, N$), respectively.[3] Then the five measurements are defined as follows:

[3]Here, $N = 207$ for the virus dataset and $N = 978$ for the plant dataset.

$$\textit{Accuracy} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{|\mathcal{M}(\mathbb{Q}_i) \cap \mathcal{L}(\mathbb{Q}_i)|}{|\mathcal{M}(\mathbb{Q}_i) \cup \mathcal{L}(\mathbb{Q}_i)|}\right) \tag{10}$$

$$\textit{Precision} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{|\mathcal{M}(\mathbb{Q}_i) \cap \mathcal{L}(\mathbb{Q}_i)|}{|\mathcal{M}(\mathbb{Q}_i)|}\right) \tag{11}$$

$$\textit{Recall} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{|\mathcal{M}(\mathbb{Q}_i) \cap \mathcal{L}(\mathbb{Q}_i)|}{|\mathcal{L}(\mathbb{Q}_i)|}\right) \tag{12}$$

$$\textit{F1} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{2|\mathcal{M}(\mathbb{Q}_i) \cap \mathcal{L}(\mathbb{Q}_i)|}{|\mathcal{M}(\mathbb{Q}_i)|+|\mathcal{L}(\mathbb{Q}_i)|}\right) \tag{13}$$

$$\textit{HL} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{|\mathcal{M}(\mathbb{Q}_i) \cup \mathcal{L}(\mathbb{Q}_i)| - |\mathcal{M}(\mathbb{Q}_i) \cap \mathcal{L}(\mathbb{Q}_i)|}{M}\right) \tag{14}$$

where $|\cdot|$ means counting the number of elements in the set therein and $\cap$ represents the intersection of sets.

Two additional measurements [11], [15] are often used in multi-label subcellular localization prediction. They are overall locative accuracy (*OLA*) and overall actual accuracy (*OAA*). The former is given by:

$$\textit{OLA} = \frac{1}{\sum_{i=1}^{N}|\mathcal{L}(\mathbb{Q}_i)|}\sum_{i=1}^{N}|\mathcal{M}(\mathbb{Q}_i) \cap \mathcal{L}(\mathbb{Q}_i)|, \tag{15}$$

and the overall actual accuracy (*OLA*) is:

$$\textit{OAA} = \frac{1}{N}\sum_{i=1}^{N}\Delta[\mathcal{M}(\mathbb{Q}_i), \mathcal{L}(\mathbb{Q}_i)] \tag{16}$$

where

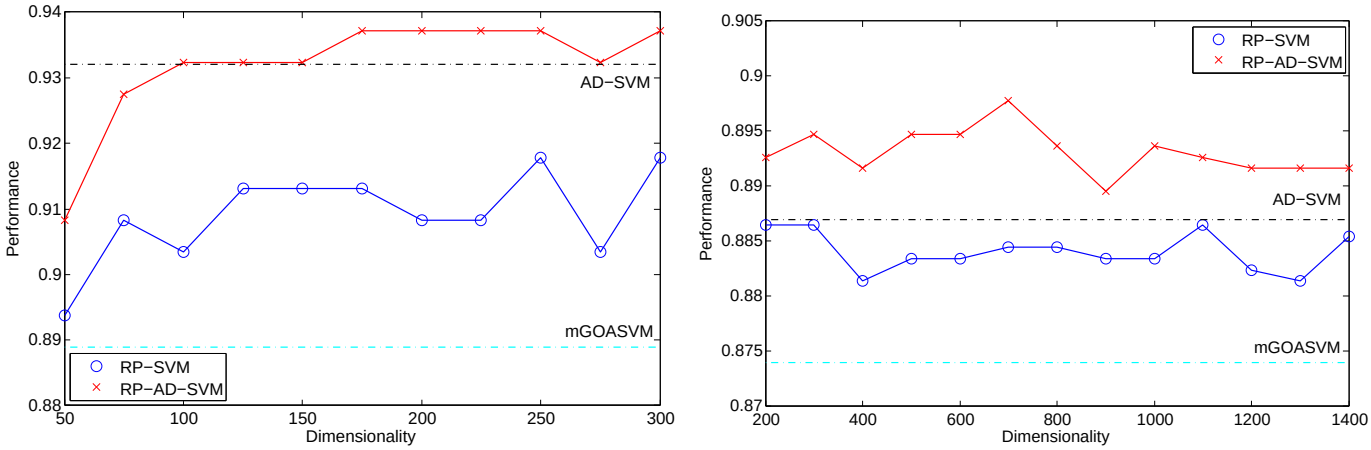
$$\Delta[\mathcal{M}(\mathbb{Q}_i), \mathcal{L}(\mathbb{Q}_i)] = \begin{cases} 1 & \text{, if } \mathcal{M}(\mathbb{Q}_i) = \mathcal{L}(\mathbb{Q}_i) \\ 0 & \text{, otherwise.} \end{cases} \tag{17}$$

Among all the metrics mentioned above, *OAA* is the most stringent and objective. This is because if only some (but not all) of the subcellular locations of a query protein are correctly predict, the numerators of the other 4 measures (Eqs. 10 to 15) are non-zero, whereas the numerator of *OAA* in Eq. 16 is 0 (thus contribute nothing to the frequency count). Therefore, we will focus on *OAA*, and unless stated otherwise, the term 'performance' in the following section refers to *OAA*.

VII. Results and Discussions

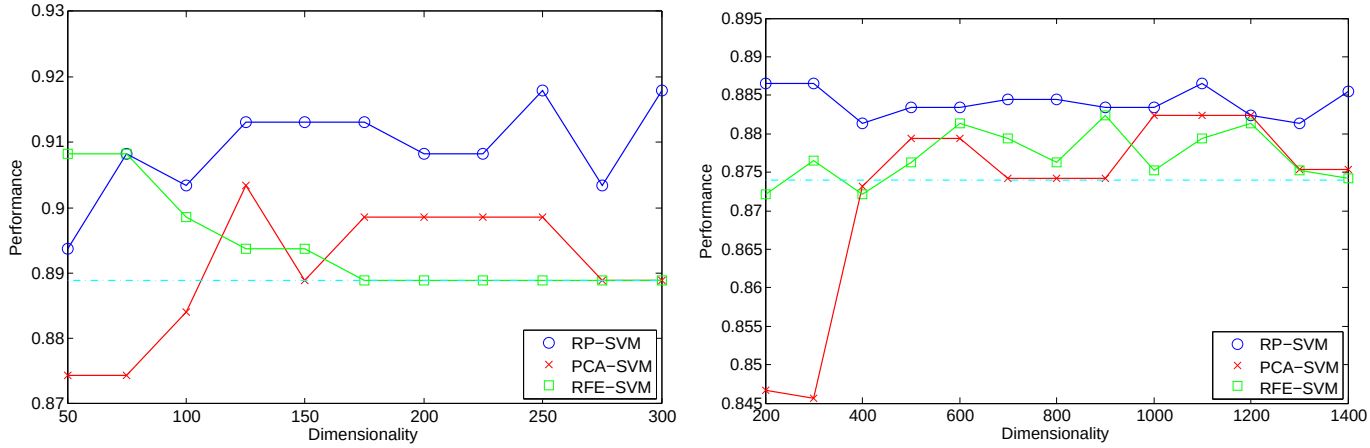
A. Performance of Ensemble Random Projection

Fig. 2(a) shows the performances of RP-SVM and RP-AD-SVM for different feature dimensions based on leave-one-out cross-validation on the virus dataset. The cyan dotted lines and black dotted lines represent the performance of mGOASVM [15] and AD-SVM [34], respectively. In other words, these two horizontal lines represent the original performance without dimension reduction for the two decision schemes. The dimensionality of the original feature vectors is 331. As can be seen, for dimensions between 50 and 300, the performance of RP-SVM is better than that of mGOASVM, which demonstrates that RP can boost the classification performance even the dimension is only one-sixth (50/331) of that of the original one. This suggests that the original feature vectors really have irrelevant or redundant information. Fig. 2(a) also shows



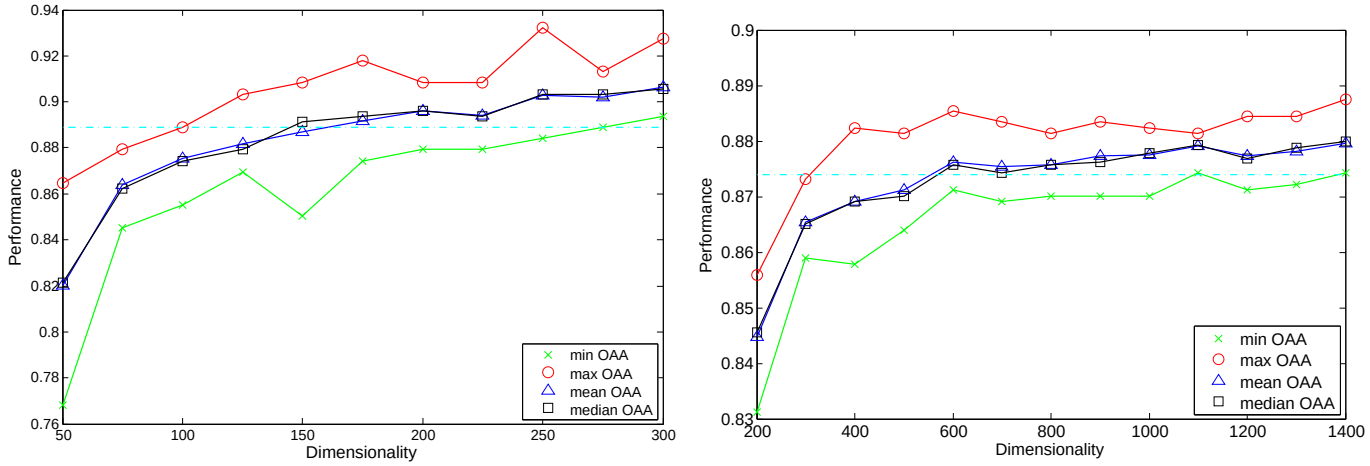
(a) Performance on the virus dataset (b) Performance on the plant dataset

Fig. 2. Performance of RP-SVM and RP-AD-SVM at different feature dimensions based on leave-one-out cross-validation (LOOCV) on (a) the virus dataset and (b) the plant dataset. The cyan dotted lines and black dotted lines in both figures represent the performance of mGOASVM [15] and AD-SVM [34] on the two datasets, respectively.



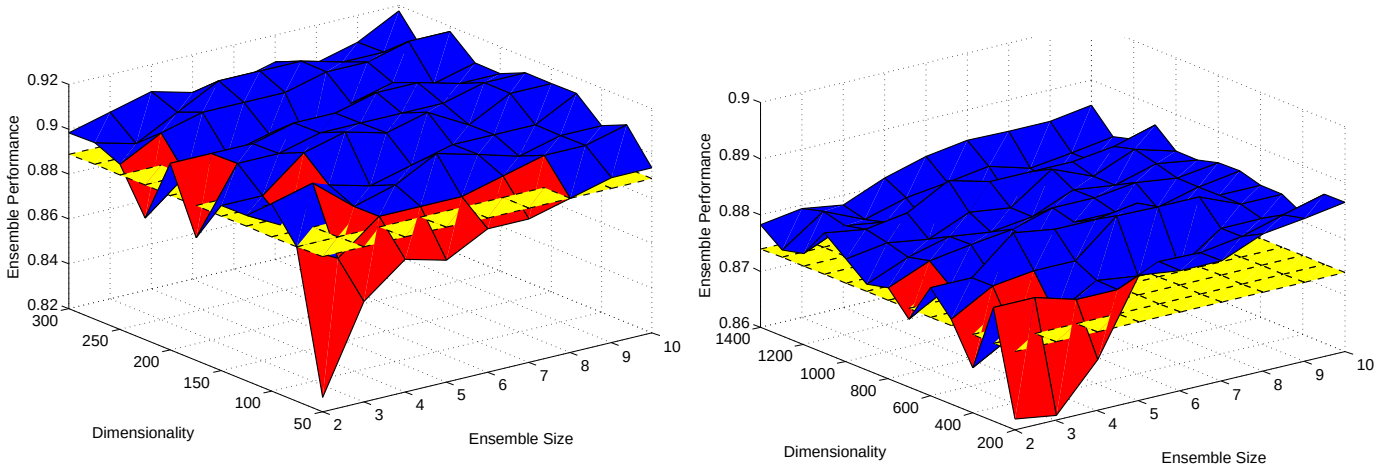
(a) Performance on the virus dataset (b) Performance on the plant dataset

Fig. 3. Comparing ensemble random projection with other dimension-reduction methods at different feature dimensions based on leave-one-out cross-validation (LOOCV) on (a) the virus dataset and (b) the plant dataset. The cyan dotted lines in both figures represent the performance of mGOASVM for the two datasets.



(a) Performance on the virus dataset (b) Performance on the plant dataset

Fig. 4. Performance of 1-RP-SVM at different feature dimensions based on leave-one-out cross-validation (LOOCV) on (a) the virus dataset and (b) the plant dataset. The cyan dotted lines in both figures represent the performance of mGOASVM on the two datasets. *1-RP-SVM*: RP-SVM with an ensemble size of 1.



(a) Performance on the virus dataset

(b) Performance on the plant dataset

Fig. 5. Performance of RP-SVM at different feature dimensions and different ensemble sizes of random projection based on leave-one-out cross-validation (LOOCV) on (a) the virus dataset and (b) the plant dataset. The blue (red) areas represent the conditions for which RP-SVM outperforms (is inferior to) mGOASVM. The yellow dotted planes in both figures represent the performance of mGOASVM on the two datasets. *Ensemble Size*: Number of applications of random projections for constructing the ensemble classifier.

that the performance of RP-AD-SVM is equivalent to, or better than that of AD-SVM when the dimensionality is larger than 100. This result demonstrates that random projection is complementary to the adaptive decision scheme. Similar conclusions can be drawn from the plant dataset shown in Fig. 2(b). Comparing Fig. 2(a) and Fig. 2(b) reveals that for the plant dataset, RP-AD-SVM outperforms AD-SVM for a wild range of feature dimensions (200 to 1541)[4], whereas for the virus dataset, the former outperforms the latter at a much narrower range (100 to 300). This suggests that RP-AD-SVM is more robust in classifying plant proteins than in classifying virus proteins.

B. Comparing with Other Dimension-Reduction Methods

Fig. 3(a) and Fig. 3(b) compare RP-SVM with other dimension-reduction methods on the virus dataset and the plant dataset, respectively. Here, PCA-SVM and RFE-SVM mean replacing RP with principal component analysis (PCA) and recursive feature elimination (RFE) [24]. As can be seen, for the virus dataset, both RP-SVM and RFE-SVM perform better than mGOASVM when the dimensionality is larger than 50, while PCA-SVM performs better than mGOASVM only when the dimensionality is larger than 100. This suggests that the former two methods are more robust than PCA-SVM. When the dimension is higher than 75, RP-SVM outperforms both RFE-SVM and PCA-SVM, although RFE-SVM performs the best when the dimension is 50. For the plant dataset, only RP-SVM performs the best for a wide range of dimensionality, while RFE-SVM and PCA-SVM perform poorly when the dimension is reduced to 200 (out of 1541).

C. Performance of Single Random-Projection

Fig. 4(a) and Fig. 4(b) show the performance statistics of RP-SVM on the virus and the plant datasets, respectively, when the ensemble size (L in Eq. 5) is fixed to 1, which we refer to as 1-RP-SVM for simplicity. We created ten 1-RP-SVM classifiers, each with a different RP matrix. The *min OAA, max OAA, mean OAA* and *median OAA* represent the minimum, maximum, mean and median *OAA* of these 10 classifiers. As can be seen, for both datasets, even the *max OAA* is not always higher than that of mGOASVM, let alone the *minimum, mean or median OAA*. This demonstrates that a single RP cannot guarantee that the original performance can be kept when the dimension is reduced. On the contrary, combining the effect of several RPs, as evidenced by Fig. 2, can boost the performance to a level higher than any of the individual RPs.

D. Effect of Dimensions and Ensemble Size

As individual RP cannot guarantee good performance, it is reasonable to ask: at least how many applications of RP can guarantee that the performance of the ensemble classifier is equivalent to, or even better than that of the one without RP (i.e., mGOASVM)? Fig. 5(a) and Fig. 5(b) show the performance of RP-SVM for different dimensions and different ensemble sizes of RPs on the virus and plant datasets, respectively. The blue/red areas represent the condition under which RP-SVM performs better/worse than mGOASVM. The yellow dotted planes in both figures represent the performance of mGOASVM on the two datasets. As can be seen, in the virus dataset, for dimensionality between 75 and 300, the performance of RP-SVM with at least 3 times of RP is better than that of mGOASVM; for dimensionality 50, we need at least 8 applications of RP to guarantee that the performance will not deteriorate. In the plant dataset, for dimensionality from 300 to 1400, RP-SVM with at least 4 applications of RP can outperform mGOASVM; for dimensionality 200, we need at least 5 applications of RP to obtain a performance better than mGOASVM. These results suggest that the proposed RP-SVM is very robust because only 3 or 4 applications of RP

[4]The dimensionality of the original feature vectors for the plant dataset is 1541.

will be sufficient to achieve good performance.

E. Comparing with State-of-the-Art Predictors

Table I and Table II compare the performance of RP-SVM against several state-of-the-art multi-label predictors on the virus and plant dataset.[5] All of the predictors use the information of GO terms as features. From the classification perspective, Virus-mPLoc [13] uses an ensemble OET-KNN (optimized evidence-theoretic K-nearest neighbors) classifier; iLoc-Virus [11] uses a multi-label KNN classifier; KNN-SVM [14] uses an ensemble of classifiers combining KNN and SVM; mGOASVM [15] uses a multi-label SVM classifier; and the proposed RP-SVM uses ensemble RP to perform dimension reduction.

As shown in Table I, RP-SVM performs significantly better than Virus-mPLoc and iLoc-Virus. Both the *OLA* and *OAA* of RP-SVM are more than 16% (absolute) higher than iLoc-Virus. They also perform significantly better than KNN-SVM in terms of *OLA*. When comparing with mGOASVM, the *OAA* of RP-SVM is more than 2% (absolute) higher than that of mGOASVM, although with the same *OLA*. In terms of *Accuracy, Precision, Recall, F1* and *HL*, RP-SVM perform better than mGOASVM. The results suggest that the proposed RP-SVM performs better than the state-of-the-art classifiers. The individual locative accuracies of RP-SVM are remarkably higher than that of Virus-mPLoc, iLoc-Virus and KNN-SVM, and are comparable to mGOASVM.

Similar conclusions can be drawn for the plant dataset. As can be seen from Table II, RP-SVM performs better than other predictors in terms of all metrics.

VIII. Conclusions

This paper proposes a random-projection based ensemble classifier for predicting multi-label protein subcellular localization. By exploiting the information in the gene ontology annotation database, a GO-based feature vector is constructed for each query protein. Subsequently, random matrices, whose elements conform to the distribution suggested by Achlioptas, were used to project the GO-vector onto a much lower-dimensional space, which are then classified by an ensemble of multi-label SVM classifiers through several applications of individual RP. Experimental results show that an ensemble size of 3 or 4 will be sufficient to achieve good performance and reduce the feature dimensions by as many as six folds.

Acknowledgment

This work was in part supported by The Hong Kong Research Grant Council, Grant No. PolyU5264/09E and HKPolyU Grant No. G-YJ86.

References

[1] G. Lubec, L Afjehi-Sadat, J. W. Yang, and J. P. John, "Searching for hypothetical proteins: theory and practice based upon original data and literature," *Prog. Neurobiol*, vol. 77, pp. 90–127, 2005.

[2] M. C. Hung and W. Link, "Protein localization in disease and therapy," *J. of Cell Sci.*, vol. 124, no. Pt 20, pp. 3381–3392, 2011.

[3] H. Nakashima and K. Nishikawa, "Discrimination of intracellular and extracellular proteins using amino acid composition and residue-pair frequencies," *J. Mol. Biol.*, vol. 238, pp. 54–61, 1994.

[4] M. W. Mak, J. Guo, and S. Y. Kung, "PairProSVM: Protein subcellular localization based on local pairwise profile alignment and SVM," *IEEE/ACM Trans. on Computational Biology and Bioinformatics*, vol. 5, no. 3, pp. 416 – 422, 2008.

[5] K. C. Chou, "Prediction of protein cellular attributes using pseudo amino acid composition," *Proteins: Structure, Function, and Genetics*, vol. 43, pp. 246–255, 2001.

[6] S. Wan, M. W. Mak, and S. Y. Kung, "Protein subcellular localization prediction based on profile alignment and Gene Ontology," in *2011 IEEE International Workshop on Machine Learning for Signal Processing (MLSP'11)*, Sept 2011, pp. 1–6.

[7] K. C. Chou and H. B. Shen, "Plant-mPLoc: A top-down strategy to augment the power for predicting plant protein subcellular localization," *PLoS ONE*, vol. 5, pp. e11335, 2010.

[8] S. Mei, "Multi-label multi-kernel transfer learning for human protein subcellular localization," *PLoS ONE*, vol. 7, no. 6, pp. e37716, 2012.

[9] S. Wan, M. W. Mak, and S. Y. Kung, "GOASVM: Protein subcellular localization prediction based on gene ontology annotation and SVM," in *2012 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'12)*, 2012, pp. 2229–2232.

[10] K. Y. Lee, D. W. Kim, D. K. Na, K. H. Lee, and D. H. Lee, "PLPD: Reliable protein localization prediction from imbalanced and overlapped datasets," *Nucleic Acids Research*, vol. 34, no. 17, pp. 4655–4666, 2006.

[11] X. Xiao, Z. C. Wu, and K. C. Chou, "iLoc-Virus: A multi-label learning classifier for identifying the subcellular localization of virus proteins with both single and multiple sites," *Journal of Theoretical Biology*, vol. 284, pp. 42–51, 2011.

[12] K. C. Chou and H. B. Shen, "Predicting eukaryotic protein subcellular location by fusing optimized evidence-theoretic K-nearest neighbor classifiers," *J. of Proteome Research*, vol. 5, pp. 1888–1897, 2006.

[13] H. B. Shen and K. C. Chou, "Virus-mPLoc: A fusion classifier for viral protein subcellular location prediction by incorporating multiple sites," *J. Biomol. Struct. Dyn.*, vol. 26, pp. 175–186, 2010.

[14] L. Q. Li, Y. Zhang, L. Y. Zou, Y. Zhou, and X. Q. Zheng, "Prediction of protein subcellular multi-localization based on the general form of Chou's pseudo amino acid composition," *Protein & Peptide Letters*, vol. 19, pp. 375–387, 2012.

[15] S. Wan, M. W. Mak, and S. Y. Kung, "mGOASVM: Multi-label protein subcellular localization based on gene ontology and support vector machines," *BMC Bioinformatics*, vol. 13, pp. 290, 2012.

[16] P. Resnik, "Semantic similarity in a taxonomy: an information-based measure and its application to problems of ambiguity in natural language," *Journal of Artificial Intelligence Research*, vol. 11, pp. 95–130, 1999.

[17] P. W. Lord, R. D. Stevens, A. Brass, and C. A. Goble, "Investigating semantic similarity measures across the Gene Ontology: the relationship between sequence and annotation ," *Bioinformatics*, vol. 19, no. 10, pp. 1275–1283, 2003.

[18] C. Pesquita, D. Faria, A. O. Falcao, P. Lord, and F. M. Counto, "Semantic similarity in biomedical ontologies," *PLoS Computational Biology*, vol. 5, no. 7, pp. e1000443, 2009.

[19] S. Wan, M. W. Mak, and S. Y. Kung, "Semantic similarity over gene ontology for multi-label protein subcellular localization," *Engineering*, vol. 5, pp. 68–72, 2013.

[20] C. H. Papadimitriou, P. Raghavan, H. Tamaki, and S. Vempala, "Latent semantic indexing: A probabilistic analysis," in *Proceedings of the 17th ACM Symposium on the Principles of Database Systems*, 1998, pp. 159–168.

[21] M. Kurimo, "Indexing audio documents by using latent semantic analysis and som," in *Kohonen Maps*. 1999, pp. 363–374, Elsevier.

[22] E. Bingham and H. Mannila, "Random projection in dimension reduction: Applications to image and text data," in *the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'01)*, 2001, pp. 245–250.

[23] S. Dasgupta, "Learning mixtures of Gaussians," in *40th Annual IEEE Symposium on Foundations of Computer Science,*, 1999, pp. 634–644.

[24] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene selection for cancer classification using support vector machines," *Machine Learning*, vol. 46, pp. 389–422, 2002.

[25] S. Wan, M. W. Mak, and S. Y. Kung, "GOASVM: A subcellular location predictor by incorporating term-frequency gene ontology into the general

[5] Note that we used the best performance of RP-SVM here.

TABLE I

COMPARING RP-SVM WITH STATE-OF-THE-ART MULTI-LABEL PREDICTORS BASED ON LEAVE-ONE-OUT CROSS VALIDATION (LOOCV) USING THE VIRUS DATASET. "–" MEANS THE CORRESPONDING REFERENCES DO NOT PROVIDE THE RELATED METRICS. *Host ER*: HOST ENDOPLASMIC RETICULUM.

Label	Subcellular Location	LOOCV Locative Accuracy (LA)				
		Virus-mPLoc [13]	KNN-SVM [14]	iLoc-Virus [11]	mGOASVM [15]	RP-SVM
1	Viral capsid	8/8 = 1.000	8/8 = 1.000	8/8 = 1.000	8/8 = 1.000	8/8 = 1.000
2	Host cell membrane	19/33 = 0.576	27/33 = 0.818	25/33 = 0.758	32/33 = 0.970	31/33 = 0.939
3	Host ER	13/20 = 0.650	15/20 = 0.750	15/20 = 0.750	17/20 = 0.850	17/20 = 0.850
4	Host cytoplasm	52/87 = 0.598	86/87 = 0.988	64/87 = 0.736	85/87 = 0.977	86/87 = 0.989
5	Host nucleus	51/84 = 0.607	54/84 = 0.651	70/84 = 0.833	82/84 = 0.976	82/84 = 0.976
6	Secreted	9/20 = 0.450	13/20 = 0.650	15/20 = 0.750	20/20 = 1.000	20/20 = 1.000
Overall Locative Accuracy (*OLA*)		152/252 = 0.603	203/252 = 0.807	197/252 = 0.782	244/252 = **0.968**	244/252 = **0.968**
Overall Actual Accuracy (*OAA*)		–	–	155/207 =0.748	184/207 = 0.889	190/207 = **0.918**
Accuracy		–	–	–	0.935	**0.950**
Precision		–	–	–	0.939	**0.957**
Recall		–	–	–	0.973	**0.976**
F1		–	–	–	0.950	**0.961**
HL		–	–	–	0.026	**0.020**

TABLE II

COMPARING RP-SVM WITH STATE-OF-THE-ART MULTI-LABEL PREDICTORS BASED ON LEAVE-ONE-OUT CROSS VALIDATION (LOOCV) USING THE PLANT DATASET. "–" MEANS THE CORRESPONDING REFERENCES DO NOT PROVIDE THE RELATED METRICS.

Label	Subcellular Location	LOOCV Locative Accuracy (LA)			
		Plant-mPLoc [7]	iLoc-Plant [35]	mGOASVM [15]	RP-SVM
1	Cell membrane	24/56 = 0.429	39/56 = 0.696	53/56 = 0.946	54/56 = 0.964
2	Cell wall	8/32 = 0.250	19/32 = 0.594	27/32 = 0.844	29/32 = 0.906
3	Chloroplast	248/286 = 0.867	252/286 = 0.881	272/286 = 0.951	284/286 = 0.993
4	Cytoplasm	72/182 = 0.396	114/182 = 0.626	174/182 = 0.956	172/182 = 0.945
5	Endoplasmic reticulum	17/42 = 0.405	21/42 = 0.500	38/42 = 0.905	39/42 = 0.929
6	Extracellular	3/22 = 0.136	2/22 = 0.091	22/22 = 1.000	21/22 = 0.955
7	Golgi apparatus	6/21 = 0.286	16/21 = 0.762	19/21 = 0.905	19/21 = 0.905
8	Mitochondrion	114/150 = 0.760	112/150 = 0.747	150/150 = 1.000	150/150 = 1.000
9	Nucleus	136/152 = 0.895	140/152 = 0.921	151/152 = 0.993	148/152 = 0.974
10	Peroxisome	14/21 = 0.667	6/21 = 0.286	21/21 = 1.000	21/21 = 1.000
11	Plastid	4/39 = 0.103	7/39 = 0.179	39/39 = 1.000	37/39 = 0.949
12	Vacuole	26/52 = 0.500	28/52 = 0.538	49/52 = 0.942	50/52 = 0.962
Overall Locative Accuracy (*OLA*)		672/1055 = 0.637	756/1055 = 0.717	1015/1055 = 0.962	1024/1055 = **0.971**
Overall Actual Accuracy (*OAA*)		–	666/978 = 0.681	855/978 = 0.874	867/978 = **0.887**
Accuracy		–	–	0.926	**0.938**
Precision		–	–	0.933	**0.946**
Recall		–	–	0.968	**0.979**
F1		–	–	0.942	**0.954**
HL		–	–	0.013	**0.011**

form of Chou's pseudo-amino acid composition," *Journal of Theoretical Biology*, vol. 323, pp. 40–48, 2013.

[26] X. Wang and G. Z. Li, "A multi-label predictor for identifying the subcellular locations of singleplex and multiplex eukaryotic proteins," *PLoS ONE*, vol. 7, no. 5, pp. e36317, 2012.

[27] K. C. Chou, "Some remarks on predicting multi-label attributes in molecular biosystems," *Molecular BioSystems*, vol. 9, pp. 1092–1100, 2013.

[28] K. C. Chou and H. B. Shen, "Cell-PLoc: A package of web-servers for predicting subcellular localization of proteins in various organisms," *nature Protocols*, vol. 3, pp. 153–162, 2008.

[29] S. F. Altschul, T. L. Madden, A. A. Schaffer, J. Zhang, Z. Zhang, W. Miller, and D. J. Lipman, "Gapped BLAST and PSI-BLAST: A new generation of protein database search programs," *Nucleic Acids Res.*, vol. 25, pp. 3389–3402, 1997.

[30] D. Barrel, E. Dimmer, R. P. Huntley, D. Binns, C O'Donovan, and R. Apweiler, "The GOA database in 2009—an integrated Gene Ontology Annotation resource," *Nucl. Acids Res.*, vol. 37, pp. D396–D403, 2009.

[31] W. B. Johnson and J. Lindenstrauss, "Extensions of Lipschitz mappings into a Hilbert space," in *Conference in Modern Analysis and Probability*, 1984, pp. 599–608.

[32] P. Frankl and H. Maehara., "The Johnson-Lindenstrauss lemma and the sphericity of some graphs," *Journal of Combinatorial Theory, Series B*, vol. 44, pp. 355–362, 1988.

[33] D. Achlioptas, "Database-friendly random projections: Johnson-Lindenstrauss with binary coins," *Journal of Computer and Systems Sciences*, vol. 66, pp. 671–687, 2003.

[34] S. Wan and M. W. Mak and S. Y. Kung, "Adaptive thresholding for multi-label SVM classification with application to protein subcellular localization prediction," in *2013 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'13)*, 2013, pp. 3547–3551.

[35] Z. C. Wu, X. Xiao, and K. C. Chou, "iLoc-Plant: A multi-label classifier for predicting the subcellular localization of plant proteins with both single and multiple sites," *Molecular BioSystems*, vol. 7, pp. 3287–3297, 2011.