Pattern Recognition 70 (2017) 89–103

Contents lists available at ScienceDirect

Pattern Recognition

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

Hierarchical Multi-label Classification using Fully Associative Ensemble Learning

L. Zhang, S.K. Shah, I.A. Kakadiaris*

Computational Biomedicine Lab, 4849 Calhoun Rd, Rm 373, Houston, TX 77204, United States

ARTICLE INFO

Article history:
Received 26 October 2016
Revised 4 April 2017
Accepted 7 May 2017
Available online 8 May 2017

Keywords:
Hierarchical multi-label classification
Ensemble learning
Ridge regression

ABSTRACT

Traditional flat classification methods (*e.g.*, binary or multi-class classification) neglect the structural information between different classes. In contrast, Hierarchical Multi-label Classification (HMC) considers the structural information embedded in the class hierarchy, and uses it to improve classification performance. In this paper, we propose a local hierarchical ensemble framework for HMC, *Fully Associative Ensemble Learning* (FAEL). We model the relationship between each class node's global prediction and the local predictions of all the class nodes as a multi-variable regression problem with Frobenius norm or l_1 norm regularization. It can be extended using the kernel trick, which explores the complex correlation between global and local prediction. In addition, we introduce a binary constraint model to restrict the optimal weight matrix learning. The proposed models have been applied to image annotation and gene function prediction datasets with tree structured class hierarchy and large scale visual recognition dataset with Direct Acyclic Graph (DAG) structured class hierarchy. The experimental results indicate that our models achieve better performance when compared with other baseline methods.

Published by Elsevier Ltd.

1. Introduction

Hierarchical Multi-label Classification (HMC) is a variant of classification where each sample has more than one label and all these labels are organized hierarchically in a tree or Direct Acyclic Graph (DAG). In reality, HMC can be applied to many domains [1–3]. In web page classification, one website with the label "football" could be labeled with a high level label "sport". In image annotation, an image tagged as "outdoor" might have other low level concept labels, like "beach" or "garden". In gene function prediction, a gene can be simultaneously labeled as "metabolism" and "catalytic or binding activities" by the biological process hierarchy and the molecular function hierarchy, respectively.

A rich source of hierarchical information in tree and DAG structured class hierarchies is helpful to improve classification performance [4]. Based on how this information is used, previous HMC approaches can be divided into global (big-bang) or local [5]. Global approaches learn a single model for the whole class hierarchy. Global approaches enjoy smaller model size because they build one model for the whole hierarchy. However, they ignore the local modularity, which is an essential advantage of HMC. Local approaches first build multiple local classifiers on the class hierarchy. Then, hierarchical information is aggregated across the local prediction results of all the local classifiers to obtain the global prediction results for all the nodes. We refer to "local prediction result" and "global prediction result" as "local prediction" and "global prediction", respectively. Previous local approaches suffer from three drawbacks. First, most of them focus only on the parent-child relationship. Other relationships in the hierarchy (e.g., ancestor-descendant, siblings) are ignored. Second, their models are sensitive to local prediction. The global prediction of each node is only decided by the local predictions of several closely related nodes. The error of local predictions is more likely to propagate to global predictions. Third, most local methods assume that the local structural constraint between two nodes will be reflected in their local predictions. However, this assumption might be shaken by different choices of features, local classification models, and positive-negative sample selection rules [6,7]. In such situations, previous methods would fail to integrate valid structural information into local prediction.

In this paper, we propose a novel local HMC framework, Fully Associative Ensemble Learning (FAEL). We call it "fully associative ensemble" because in our model the global prediction of each node considers the relationships between the current node and all the other nodes. Specifically, a multi-variable regression model is built to minimize the empirical loss between the global predictions of all the training samples and their corresponding true label observations.

* Corresponding author.
E-mail addresses: lzhang34@uh.edu (L. Zhang), sshah@central.uh.edu (S.K. Shah), ioannisk@uh.edu (I.A. Kakadiaris).

http://dx.doi.org/10.1016/j.patcog.2017.05.007
0031-3203/Published by Elsevier Ltd.

Our contributions are: we (i) developed a novel local hierarchical ensemble framework, in which all the structural relationships in the class hierarchy are used to obtain global prediction; (ii) introduced empirical loss minimization into HMC, so that the learned model can capture the most useful information from historical data; and (iii) proposed sparse, kernel, and binary constraint HMC models.

Parts of this work have been published in [8]. In this paper, we extend that work by providing: (i) the sparse basic model with l_1 norm; (ii) a new application of DAG structured class hierarchy in a visual recognition dataset based on deep learning features; (ii) the sensitivity analysis of all the parameters; (iii) the performance of two more kernel functions (Laplace kernel and Polynomial kernel) in the kernel model; and (iv) statistical analysis of all the experimental results.

The rest of this paper is organized as follows: in Section 2 we discuss related work. Section 3 describes the proposed FAEL models. The experimental design, results and analysis are presented in Section 4. Section 5 concludes the paper.

2. Related work

In this section, we review the most recent works in HMC and flat multi-label classification, especially those that are related to our work. Also, we illustrate how our framework is different from previous ones.

In HMC, Both global and local approaches have been developed. Most global approaches are extended from classic single label machine learning algorithms. Wang et al. [9] used association rules for hierarchical document categorization. Hierarchical relationships between different classes are defined based on the similarity of the documents belonging to them. Vens et al. [10] introduced a modified version of decision tree for HMC. One tree is learned to predict all the classes at once. Bi et al. [11] formulated the HMC as a graph problem of finding the best subgraph in a tree or DAG. Kernel dependency estimation is used to reduce the original hierarchy to a manageable number of single label learning problems. A generalized condensing sort and select algorithm is applied to preserve the parent-child relationships in the label hierarchy. Based on a predictive clustering tree, Dimitrovski et al. [2] proposed the cluster-HMC algorithm for medical image annotation. In another work [12], Dimitrovski et al. introduced ensembles of predictive clustering trees for hierarchical classification of diatom images. Bagging and random forests are used to combine the predictions of different trees. Cerri et al. [13] introduced genetic algorithm to HMC. Genetic algorithm is used to evolve the antecedents of classification rules. A set of optimized antecedents is selected to make a prediction for the corresponding classes. Barros et al. [14] introduced the probabilistic clustering HMC framework for protein function problem. The assumption is that training instances can fit to several probability distributions, where instances from the same distribution also share similar class vectors. The major drawback of previous global models is that they ignore the local modularity in the label hierarchy, such as parent-child, ancestor-descendent, and sibling relationships between different labels.

Local approaches also draw heavy attention. Dumais and Chen [15] applied a multiplicative threshold to update local prediction. The posterior probability is computed based on the parent-child relationship. Barutcuoglu and DeCoro [16] proposed a Bayesian aggregation model for image shape classification. The main idea is to obtain the most probable consistent set of global predictions. Cesa-Bianchi et al. [17] developed a top down HMC method using hierarchical Support Vector Machine (SVM), where SVM learning is applied to a node only if its parent has been labeled as positive. Alaydie et al. [18] introduced hierarchical multi-label boosting with label dependency. The pre-defined label hierarchy is used to decide the training set for each classifier. The dependencies of the children are analyzed using Bayesian method and instance based similarity. Ren et al. [19] proposed to address the HMC problem for documents in social text streams with Structural SVM (S-SVM). Multiple structural classifiers are built for each chunk of classes to overcome the unbalanced sample problem. Cerri et al. [20] proposed to build multi-layer perceptron for each level of labels in the label hierarchy. The predictions made by a given level are used as inputs to the next level. Vateekul et al. [21] introduced a hierarchical R-SVM system for gene function prediction. The threshold adjustment from R-SVM is used to mitigate the problem of false negatives in HMC. Valentini [22,23] presented the True Path Rule (TPR) ensembles. In this method, positive local predictions of child nodes affect their parent nodes and negative local predictions of non-leaf nodes affect their descendant nodes.

Our work is inspired by both top-down and bottom-up local models. The top-down models propagate predictions from high level nodes to the bottom [15,24]. In contrast, the bottom-up models propagate predictions from the bottom to the whole hierarchy [25,26]. As a state-of-the-art method, the TPR ensemble integrates both top-down and bottom-up rules [22]. The global prediction of each parent node is updated by the positive local predictions of its child nodes. Then, a top-down rule is applied to synchronize the obtained global predictions. The method is also extended to handle DAG structured class hierarchy [4,23]. In contrast to TPR, our model incorporates all pairs of hierarchical relationships and attempts to learn a fully associative weight matrix from training data. Take the “human” sub-hierarchy from the extended IAPR TC-12 image dataset [27] for example. Fig. 1 depicts the merits of our model and shows the contribution of hierarchical and sibling nodes on each local prediction. The weight matrix computed shows that each local node influences its own decision positively, while nodes not directly connected in the hierarchy provide a negative influence. Since the weight matrix of our model is learned based on all the training samples, we can minimize the influence of outlier examples of each node. The learning model also helps to avoid the error propagation problem, because all the global predictions are obtained simultaneously.

Many works have also been proposed for flat multi-label classification, where no specific hierarchical relationships between labels are given. Because multiple labels share the same input space and semantics conveyed by different labels are usually correlated, it is essential to exploit the correlation information contained in different labels by a multi-task learning framework. Ji et al. [28] developed a general multi-task framework for extracting shared structures in multi-label classification. The optimal solution to the proposed formulation is obtained by solving a generalized eigenvalue problem. Zhu et al. [29] proposed a multi-view multi-label framework with block-row regularization. The regularizer concatenates a Frobenius norm regularizer and l_{21} norm regularizer, which are used to select informative views and features. To handle the missing label problem, semi-supervised learning was introduced to multi-label classification. Luo et al. [30] proposed a manifold regularized multi-task learning algorithm. A discriminative subspace shared by multiple classification tasks is learned while manifold regularization ensures that the learned predictive structure is reliable for both labeled data and unlabeled data. In another work, Luo et al. [31] developed a multi-view matrix completion framework for semi-supervised multi-label image classification. A cross-validation strategy is used to learn combination coefficients of different views. Inspired by the great success of deep Convolutional Neural Networks (CNN) in single label image classification in the past few years [32–34], CNN-based multi-label image classification algorithms were also developed. Wei et al. [35] proposed a hypotheses CNN pooling framework. Different object segment hypotheses are taken as inputs of a shared CNN. The CNN output re-

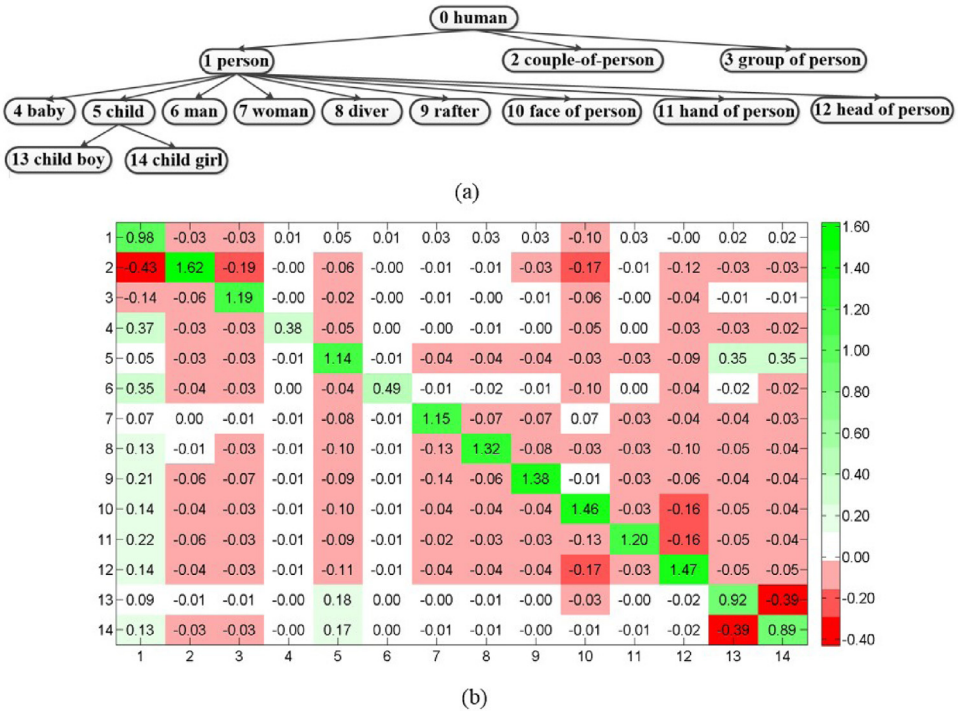

Fig. 1. (a) The "human" sub-hierarchy. (b) The weight matrix W^* learned from B-FAEL. Each element w^*_{ij} represents the weight of the ith label's local prediction to the jth label's global prediction. Using TPR, the global predictions are first computed by their local prediction and the local predictions (those above threshold 0.5) of the child nodes, then a top-down scheme is used to propagate the influence of ancestor nodes. Using our model, they are made by the local predictions of all the fourteen non-root nodes. In (b), we can observe that, for each node, the nodes in the same path give positive weights; the other nodes give negative weights. Take the weights for node 1 in the first column, for example: nodes 2 and 3 give negative weights ($w^*_{21} = -0.43$ and $w^*_{31} = -0.14$). All the remaining nodes give positive weights. This rule works for all the weights except $W^*_{1,10}$, and $W^*_{7,10}$. These observations follow the fact that each image region is annotated by the labels of one continuous path from the root to the bottom, gradually and exclusively.

sults from different hypotheses are aggregated with max pooling to produce multi-label predictions. Wang et al. [36] introduced recurrent neural networks (RNN) to capture the dependencies of multiple labels in an image. Combined with CNNs, the proposed framework learns a joint image-label embedding to characterize both semantic label dependency and image label relevance. Zhao et al. [37] developed a regional gating neural network framework. Candidate image regions are fed to a shared CNN to produce regional representation. Then, the unites of region level gate and feature level gate are imposed on regional presentations to select useful contextual region features. The whole network is optimized with multi-label loss. Compared with HMC approaches, these methods ignore the hierarchical relationships between different labels.

The proposed framework also inherits features from Multi-Task Learning (MTL) works [38–41]. Our model is close to the MTLs with tree or graph structures, where pre-defined structural information is extracted to fit the learning model [42,43]. Similar to these MTLs, our hierarchical ensemble model can use various loss functions and regularization terms. One major difference lies in the features used in the model. In the MTLs, the features are shared consistently over all the tasks and they must be the same for each task. In our model, local predictions of all the nodes are used as features. Therefore, each local classifier can be built by completely different features.

3. Fully associative ensemble learning

Let $\mathcal{S} = \{s_1, s_2, \ldots, s_n\}$ represent a hierarchical multi-label training set, which comprises n samples. Its hierarchical label set is denoted by $\mathcal{C} = \{c_1, c_2, \ldots, c_l\}$. There are l labels in total, and each label corresponds to one unique node in hierarchy $\mathcal{H}$. The training label matrix is defined as a binary matrix $Y = \{y_{ij}\}$, with size $n \times l$. If the ith sample has the jth label, $y_{ij} = 1$, otherwise $y_{ij} = 0$. As a local approach, local classifiers $\mathcal{F} = \{f_1, f_2, \ldots, f_l\}$ are built on each node. The local predictions of $\mathcal{S}$ are denoted by matrix $Z = \{z_{ij}\}$, where z_{ij} represents the prediction of the ith sample on the jth label. A probabilistic classifier is used as the local learner, so we have $z_{ij} \in [0, 1]$. Similarly, we represent the global prediction matrix by $\widehat{Y} = \{\widehat{y}_{ij}\}$ with size $n \times l$. In our model, global prediction is achieved based on local prediction and hierarchical information. To take all the node-to-node relationships into account, we define $W = \{w_{ij}\}$ as a weight matrix, where w_{ij} represents the weight of the ith label's local prediction to the jth label's global prediction. Thus, each label's global prediction is a weighted sum of the local predictions of all the nodes in $\mathcal{H}$. The global prediction matrix $\widehat{Y}$ is computed as: $\widehat{Y} = ZW$.

3.1. The basic model

The simplest way to estimate the weight matrix W is by minimizing the squared loss between the global prediction matrix $\widehat{Y}$ with the true label matrix Y. To reduce the variance of w_{ij}, we penalize the Frobenius norm of W and obtain this objective function:

$$\min_W \|Y - ZW\|_F^2 + \lambda_1 \|W\|_F^2, \tag{1}$$

where the first term measures the empirical loss of the training set, the second term controls the generalization error, and λ_1 is a regularization parameter. The above function is known as ridge regression. Taking derivatives w.r.t. W and setting to zero, we have:

$$W = \left(Z^T Z + \lambda_1 I_l\right)^{-1} Z^T Y, \tag{2}$$

where I_l represents the $l \times l$ identity matrix. Thus, we obtain an analytical solution for the basic FAEL model.

Inspired the success of low rank constraint [44–46], we could replace the Frobenius norm in (1) with l_1 norm, add obtain the following objective function:

$$\min_{W_s} \|Y - ZW_s\|_F^2 + \lambda_2 \|W_s\|_1^2, \tag{3}$$

where λ_2 is a regularization parameter. This function has both smooth and non-smooth terms. The gradient descent or accelerated gradient method (AGM) [47] can be applied to solve the optimization. We employ the algorithm in SLEP package [48] to obtain a solution. However, the obtained sparse weight matrix conflicts with our goal of learning a fully associative weight matrix, where all the hierarchical relationships are considered, such as ancestor-descendant and sibling relationships. We compared the performance of the two norms on different datasets in Section 4.2. The results confirm our analysis that the Frobenius norm is a better choice for the HMC problem.

3.2. The kernel model

To capture the complex correlation between global and local prediction, we can generalize the above basic model using the kernel trick. Let Φ represent the map applied to each example's local prediction vector $\mathbf{z}_i$. A kernel function is induced by $K(\mathbf{z}_i, \mathbf{z}_j) = \Phi(\mathbf{z}_i)^T \Phi(\mathbf{z}_j)$. By replacing the term Z in (1), we obtain:

$$\min_{W_k} \|Y - \Phi W_k\|_F^2 + \lambda_1 \|W_k\|_F^2. \tag{4}$$

After several matrix manipulations [49], the solution of W_k becomes:

$$\begin{aligned} W_k &= \left(\Phi^T \Phi + \lambda_1 I_l\right)^{-1} \Phi^T Y \\ &= \Phi^T \left(\Phi \Phi^T + \lambda_1 I_n\right)^{-1} Y, \end{aligned} \tag{5}$$

where I_n represents the $n \times n$ identity matrix. For a testing example s^t and its local prediction $\mathbf{z}^t$, the global prediction $\widehat{\mathbf{y}}^t$ is obtained by $\widehat{\mathbf{y}}^t = \mathbf{z}^t W$. For a kernel version, we obtain:

$$\begin{aligned} \widehat{\mathbf{y}}_k^t &= \Phi(\mathbf{z}^t) W_k \\ &= \Phi(\mathbf{z}^t) \Phi^T \left(\Phi \Phi^T + \lambda_1 I_n\right)^{-1} Y \\ &= K(\mathbf{z}^t, \mathbf{z}) \left(K(\mathbf{z}, \mathbf{z}) + \lambda_1 I_n\right)^{-1} Y, \end{aligned} \tag{6}$$

where $K(\mathbf{z}^t, \mathbf{z}) = [k(\mathbf{z}^t, \mathbf{z}^1), k(\mathbf{z}^t, \mathbf{z}^2), \ldots, k(\mathbf{z}^t, \mathbf{z}^n)]$ and $K(\mathbf{z}, \mathbf{z}) = \{k(\mathbf{z}_i, \mathbf{z}_j)\}$ are both kernel computations.

One potential drawback of the above kernel model is its scalability. During the training phase, the complexity of computing and storing $K(\mathbf{z}, \mathbf{z})$ is significant even for moderate size problems. Therefore, we adopt a simple random sample-selection technique to reduce the kernel complexity of large-scale datasets. The assumption behind this is to select a small number of samples that could represent the distribution of large scale dataset. We randomly select $n_k (n_k \ll n)$ samples from training set for kernel model, which reduces the kernel complexity from $O(n \times n)$ to $O(n_k \times n_k)$.

3.3. The binary constraint model

Another limitation of the basic model is that the weights between different nodes are considered independently. To make full use of the hierarchical relationships between different nodes, we introduce a regularization term to the optimization function in (1). The motivation is that when we calculate the weight to a third node, the current parent node should play more role than the current child node while the current ancestor node should play a greater role than the current descendent node. In this way, we rely more on the high level nodes than on the low level nodes, rather than treating them equally.

The hierarchical structure can be viewed as a set of "binary constraints" among all the nodes. Here, we only focus on the "parent-child" constraints and the "ancestor-descendent" constraints. Let $\mathcal{R} = \{r_i(c_p, c_q)\}$ denote the binary constraint set of hierarchy $\mathcal{H}$. Each member $r_i(c_p, c_q)$ meets either $c_p =\uparrow c_q$ or $c_p =\Uparrow c_q$, where "$\uparrow$" and "$\Uparrow$" represent the "parent-child" constraint and the "ancestor-descendent" constraint, respectively [5]. The size of $\mathcal{R}$ depends on the structure of $\mathcal{H}$. Its maximum is $l \times (l-1)/2$, which is equal to the number of all the possible constraints. In this case, there is only one path from the root node to the single leaf node in the hierarchy. Now, we introduce a weight restriction to each pair of nodes in $\mathcal{R}$. Define coefficient $m_{pq} \in \mathbb{R}^+$ for the ith pair $r_i(c_p, c_q)$, so that:

$$w_{pk} = m_{pq} * w_{qk}. \tag{7}$$

The intuition behind this definition is that high-level nodes should give weights larger than low-level nodes. For the global prediction of node k, the weight of node p is m_{pq} times the weight of node q. The value of m_{pq} is set by:

$$m_{pq} = \begin{cases} \mu & c_p =\uparrow c_q \\ \mu * (e_{pq} + 1) & c_p =\Uparrow c_q \end{cases}, \tag{8}$$

where μ is a positive constant and e_{pq} represents the number of nodes between c_p and c_q. Thus, the coefficient of an "ancestor-descendent" constraint is larger than that of a "parent-child" constraint. Specifically, it is decided by the depth difference of the two corresponding nodes in the hierarchy. If there are other nodes between node c_p and node c_q, the coefficient m_{pq} is larger. Because they have an ancestor-descendent relationship, we rely more on the high level node c_p. If there are no other nodes between them, they have a parent-child relationship. If the coefficient m_{pq} is smaller, the constraint is looser than that of a ancestor-descendent relationship. In a DAG-structured class hierarchy, if one node has more than one parent node, we create constraint for each parent node separately and add them all to the binary constraint set. The same rule applies to "ancestor-descendent" constraints. All the restrictions over the hierarchy are summarized as:

$$\sum_{r_i(c_p, c_q)}^{|\mathcal{R}|} \sum_{k=1}^{l} \left(w_{pk} - m_{pq} * w_{qk}\right)^2. \tag{9}$$

To convert the above equations into a matrix version, we introduce a sparse matrix $M = [\mathbf{m}_1, \mathbf{m}_2, \ldots, \mathbf{m}_{|R|}]^T$, in which the ith row $\mathbf{m}_i$ corresponds to the ith pair in $\mathcal{R}$. Each row in M has only two non-zero entries. The pth entry is 1 and the qth entry is $-m_{pq}$, and all the other entries are zero. Thus, we obtain the regularization term of the binary constraint model:

$$\sum_{r_i(c_p, c_q)}^{|\mathcal{R}|} \sum_{k=1}^{l} \left(w_{pk} - m_{pq} * w_{qk}\right)^2 = \|MW_b\|_F^2. \tag{10}$$

Adding this term to (1), the optimization function becomes:

$$\min_{W_b} \|Y - ZW_b\|_F^2 + \lambda_1 \|W_b\|_F^2 + \lambda_3 \|MW_b\|_F^2. \tag{11}$$

Taking the derivative w.r.t. W_b, setting to zero, and merging similar terms, we obtain:

$$(Z^T Z + \lambda_1 I_l + \lambda_3 M^T M) W_b = Z^T Y. \tag{12}$$

The analytical solution of the binary constraint model is given by:

$$W_b = \left(Z^T Z + \lambda_1 I_l + \lambda_3 M^T M\right)^{-1} Z^T Y. \tag{13}$$

The analytical solution ensures a low computational complexity for this model. In practice, we can also choose a few rows from M to build the regularization term and focus on a more specific constraint set. It is also interesting to extend the binary constraint model to a kernel version. However, the rule of (9) from

[49,50] does not apply to (13) directly to obtain a closed form solution, because the component $\lambda_1 l_l + \lambda_3 M^T M$ is not an identity matrix any more. An iterative solution will increase computational complexity for the model.

3.4. Hierarchical prediction

After we get the global predictions for all the nodes, the next step is to set thresholds for the global prediction of each node, and assign proper labels for each testing sample. In the original TPR model, the author uses 0.5 as the threshold of all the nodes, which ignores the distribution difference of positive and negative samples. Here, the threshold is learned to separate them averagely. Let $\mathbf{d} = \{d_1, d_2, \ldots, d_l\}$ denote the threshold set of global prediction, where d_i corresponds to node i. Let $\mathcal{S}_i^+$ and $\mathcal{S}_i^-$ represent the positive and negative training sets of node i, respectively. Their global predictions are computed as $\widehat{Y}_i^+$ and $\widehat{Y}_i^-$. We define threshold d_i as the midpoint of the averaged positive and negative global predictions of node i:

$$d_i = 0.5 * \left(\frac{1}{|\mathcal{S}_i^+|} \sum_j \widehat{y}_{ji}^+ + \frac{1}{|\mathcal{S}_i^-|} \sum_j \widehat{y}_{ji}^- \right) \tag{14}$$

where $\widehat{y}_{ji}^+$ and $\widehat{y}_{ji}^-$ represent the global prediction of the jth sample in $\mathcal{S}_i^+$ and $\mathcal{S}_i^-$, respectively.

Based on the learned thresholds, the output labels of each testing sample should follow the hierarchical structure. All the labels with positive output can be linked into one or multiple continuous paths from the root to the bottom in hierarchy $\mathcal{H}$. Here we apply a bottom-up strategy to synchronize the output labels. Given a testing sample s^t with global prediction $\widehat{\mathbf{y}}^t = [\widehat{y}_1^t, \widehat{y}_2^t, \ldots, \widehat{y}_l^t]$, its final output $\mathbf{o}^t = [o_1^t, o_2^t, \ldots, o_l^t]$ is decided by:

$$o_i^t = \begin{cases} 1 & \widehat{y}_i^t > d_i \\ 1 & \widehat{y}_k^t > d_k, c_i = \uparrow c_k \text{ or } \Uparrow c_k \\ 0 & otherwise \end{cases}. \tag{15}$$

Note that from the above rule, we might obtain multiple valid paths as the final output. This is appropriate for some applications, such as gene function prediction, where each gene can have more than one path in the "FunCat" hierarchy. However, in other applications, such as image annotation and visual recognition, the ideal output is one path of the conceptual hierarchy that indicates the exact content of each image region. In this case, we average the global predictions on each continuous path and return the maximum path. For a DAG-structured class hierarchy, if any node in the maximum path has more than one parent node, we also link them from the root for final prediction. The pseudo-code of the proposed framework is summarized in Algorithm 1.

Algorithm 1: The Fully Associative Ensemble Learning.

Input: $\mathcal{S}^r = \{s_1^r, s_2^r, \ldots, s_n^r\}$, $\mathcal{C} = \{c_1, c_2, \ldots, c_l\}$, $\mathcal{H}$, $Y^r = \{y_{ij}^r\} \in \mathbb{R}^{n \times l}$ and $\mathcal{S}^t = \{s_1^t, s_2^t, \ldots, s_m^t\}$

Output: $\widehat{Y}^t = \{\widehat{y}_{ij}^t\} \in \mathbb{R}^{m \times l}$ and $O^t = \{o_{ij}^t\} \in \mathbb{R}^{m \times l}$

```
for i ← 1 to l do
    Select positive and negative examples for node i
    Build a local classifier f_i on node i
    Compute the local prediction of S^r on node i, f_i(S^r)
Select binary constraint pairs and obtain M
Compute W with (2), (5) or (13)
Compute d for all the nodes with (14)
for i ← 1 to m do
    Compute the local prediction of s_i^t on each node, z_i^t = f(s_i^t)
    Compute the global prediction of s_i^t with ŷ_i^t = z_i^t × W and (6)
    Compute the final output with (15)
return {Ŷ^t, O^t} ;
```

4. Experiments

This section presents the datasets and experimental methodology used to evaluate the proposed framework and compare it to other baseline methods. The sensitivity analysis of all the parameters and statistical analysis are also discussed.

4.1. Datasets and experimental methodology

4.1.1. Image annotation

We present our evaluation of the proposed models on the extended IAPR TC-12 image collection [27]. In this dataset, every image is segmented into several regions and each region is annotated by a set of labels from a tree structured conceptual hierarchy. Fig. 2 depicts a sample image and its corresponding labels. The whole conceptual hierarchy comprises 275 nodes located in six main branches: "animal", "landscape", "man-made", "human", "food", and "other". Considering their conceptual differences and hierarchy size, we build five separate sub-hierarchies with the first five main branches. Their detailed descriptions are shown in Table 1. The "other" branch is excluded because it has only six child nodes with the same depth. Given the original features from the dataset, each region is viewed as a sample. To build three-fold cross-validation, we ignore the nodes that have fewer than ten samples. Inner three-fold cross-validation is applied to select the best parameters on each fold of training data. Then we apply the best parameters to testing data. Based on [27], we use Random Forests as the basic classifier under the one-versus-all sample selection technique. The number of trees in Random Forests is set to 100. Downsampling is applied to keep the balance between positive and negative samples.

Table 1
The extended IAPR TC-12 sub-hierarchy descriptions.

Sub-hierarchies	Sample number	Node number	Tree depth
Animal	1999	41	5
Food	861	5	3
Human	17,011	14	4
Landscape	45,048	42	4
Man-made	33,984	99	5

4.1.2. Gene function prediction

Gene function prediction is another complex tree-structured HMC problem. We use six yeast datasets integrated in [22]. Their descriptions are summarized in Table 2. To compare with the results in [22], we use the same experimental settings.

4.1.3. Visual recognition

We also evaluate the proposed models on a more challenging DAG-structured visual recognition problem with ImageNet [51]. ImageNet is organized according to the WordNet hierarchy. It includes over 14 million images distributed on over 20,000 nodes. Here we use a subset with up to 686 nodes. Each leaf node has 100 images. The CaffeNet model [52] is used to extract 1000 deep learning features for each image. The Linear Support Vector Machine (LSVM) is built as the local classifier for each local node with $C = 1$. The negative sample is selected based on the one-versus-all technique. To overcome the unbalanced data issue between positive and negative images, we randomly select the same greatest

Fig. 2. Sample image with hierarchical annotations.

Table 2
The gene function dataset descriptions.

Datasets	Description	Sample number	Feature number	Node number	Tree depth
Pfam-1	Protein domain binary data from Pfam data	3529	4950	211	5
Pfam-2	Protein domain log E data from Pfam data	3529	5724	211	5
Expr	Gene Expression data	4532	250	230	5
PPI-BG	PPI data from BioGRID	4531	5367	232	5
PPI-VM	PPI data from Von Mering experiments	2338	2559	177	5
SP-sim	Sequence Pairwise similarity data	3527	6349	211	5

Table 3
The ImageNet sub-hierarchy descriptions.

Hierarchy	Number of leaf nodes	Number of total nodes	Depth
Sub-1	100	204	17
Sub-2	200	375	19
Sub-3	300	505	19
Sub-4	400	686	19

number of negative images and positive images to build each local classifier. To fully understand the performance, we build the models on 4 sub-hierarchies with different numbers of leaf nodes. The detailed information is summarized in Table 3.

4.1.4. Baseline and measurements

We compare the proposed models with the Top-Down (TD) algorithm, TPR and weighted TPR (TPR-w) [22] under F-measure and Hierarchical F-measure (HF-measure). F-measure, also known as F1 score, is used to measure the flat classification performance. It is the harmonic mean of precision and recall. By integrating structural information of prediction, HF-measure is a more appropriate performance metric in HMC [22,53]. It can capture the partially correct paths in the hierarchical taxonomy. All the experiments were run ten times with different random seeds.

4.2. Norm comparison

In this section, we first analyze the sensitivity of λ_1 and λ_2 for Frobenius norm and l_1 norm, respectively. We denote the two models as FAEL and Sparse FAEL (S-FAEL). Then, we use three-fold cross-validation to evaluate their performance, inner three-fold cross-validation is used to select the best parameters from each fold of training data. We set different ranges for parameters based on observation. In image annotation and gene function prediction datasets, we set $\lambda_1 = \{0, 10, 20, \ldots, 200\}$ and $\lambda_2 = \{0, 0.001, 0.002, \ldots, 0.02\}$. In visual recognition dataset, we set $\lambda_1 = \{0, 1, 2, \ldots, 20\}$ and $\lambda_2 = \{0, 0.001, 0.002, \ldots, 0.02\}$. The sensitivity performances are depicted in Figs. 3–8. The prediction performance from cross-validation is summarized in Table 4.

In Fig. 3 we can observe that the FAEL model is not very sensitive to choice of λ_1. Both F-measure and HF-measure remain stable against changing values of λ_1 except F-measure of "animal" and "food" hierarchies. In Fig. 4 we observe that, as λ_1 increases, the F-measure performance goes down on five datasets. Performance on the Expr dataset is the most stable one. Under HF-measure,

Table 4
FAEL and S-FAEL performance on different datasets.

Models	F-measure		HF-measure	
Datasets	FAEL	S-FAEL	FAEL	S-FAEL
Animal	**0.224**	0.220	**0.432**	0.411
Food	0.401	**0.403**	**0.495**	0.466
Human	**0.315**	0.303	**0.636**	0.625
Landscape	0.347	**0.348**	**0.571**	0.566
Man-made	**0.134**	0.131	**0.281**	0.268
Pfam-1	**0.398**	0.297	**0.459**	0.448
Pfam-2	**0.304**	0.245	**0.456**	0.436
Expr	**0.132**	0.112	**0.590**	0.573
PPI-BG	**0.281**	0.211	0.519	**0.529**
PPI-VM	**0.395**	0.297	**0.468**	0.435
SP-smi	**0.341**	0.257	0.384	**0.394**
Sub-1	**0.513**	0.372	**0.906**	0.893
Sub-2	**0.493**	0.248	**0.909**	0.884
Sub-3	**0.461**	0.191	**0.912**	0.887
Sub-4	**0.464**	0.139	**0.906**	0.872

the performance first goes up and then becomes stable on most datasets. In Fig. 5, we can observe that the performance of FAEL is stable in the parameter range. Also, the hierarchical performance of the model is not sensitive to the increase of nodes. Under HF-measure, sub-3, sub-2, and sub-4 perform even better than sub-1. In sparse model, from Figs. 6–8, the performance is more fluctuant compared to that of the Frobenius norm model. The reason is that sparse model generates spare weight matrix, which conflicts with our goal to learn a fully associative weight matrix. In the sparse weight matrix, only part of the hierarchy relationships are captured. From the results in Table 4, we can see FEAL performs better than S-FAEL on most datasets. S-FAEL can achieve comparable results on smaller size hierarchies, such as "food" and "landscape", in image annotation datasets. But for larger hierarchies in gene function prediction and visual recognition datasets, FAEL achieves better results on most datasets.

4.3. Kernel model evaluation

In the Kernel FAEL model (K-FAEL), we evaluate the performance with different values of λ_1 using three different kernels: Gaussian kernel ($\sigma = 0.05$), Laplace kernel ($\sigma = 0.05$) and Polynomial kernel ($degree = 2,\ scale = 1,\ offset = 1$). In the large scale dataset of visual recognition, we apply the sample selection technique to the training sets with over 1000 samples ($n_k = 1000$). The results on the image annotation dataset are shown in Figs. 9 and

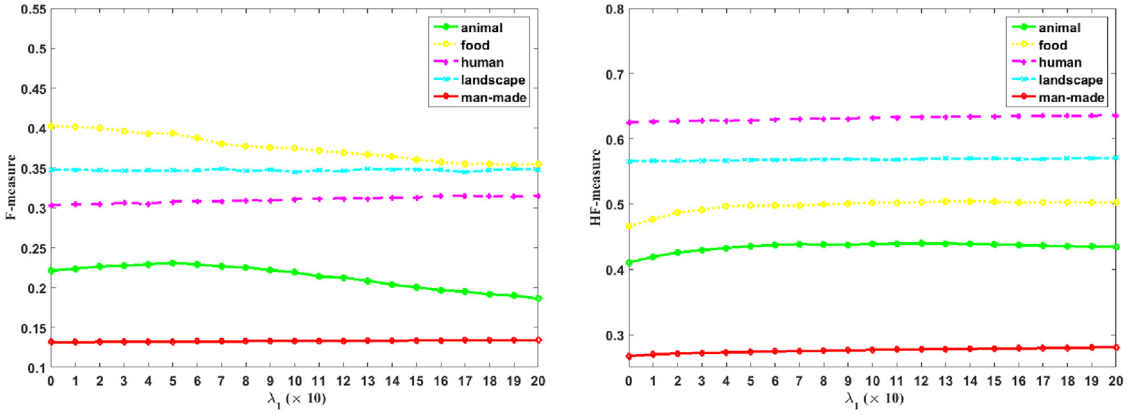

Fig. 3. FAEL performance of different λ_1 on the image annotation dataset. (L) F-measure. (R) HF-measure.

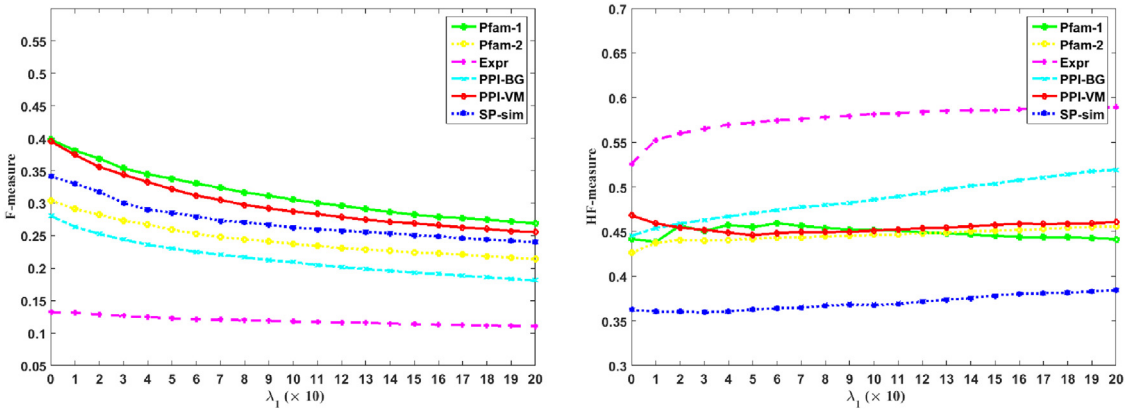

Fig. 4. FAEL performance of different λ_1 on the gene function datasets. (L) F-measure. (R) HF-measure.

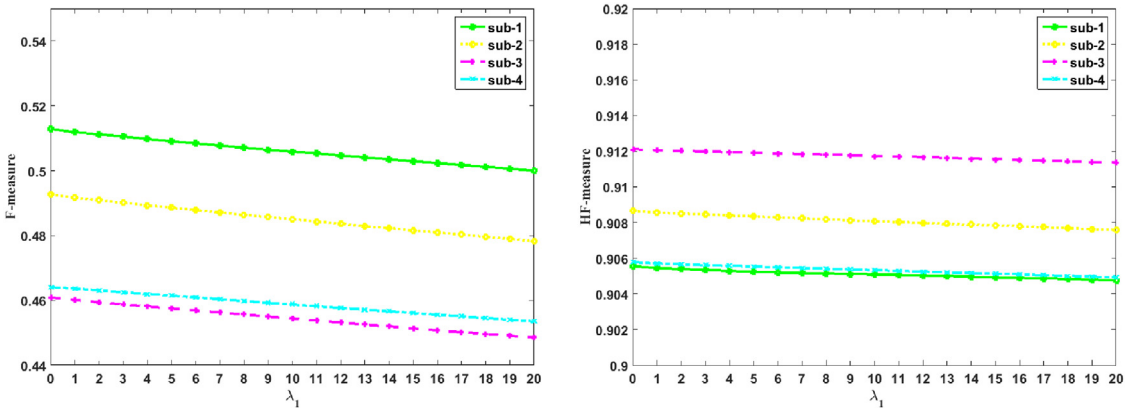

Fig. 5. FAEL performance of different λ_1 on the ImageNet sub-hierarchy dataset. (L) F-measure. (R) HF-measure.

10. The results on the gene function datasets and the ImageNet sub-hierarchy dataset are presented in Supplementary material.

In Figs. 9 and 10, different kernel functions perform differently under different hierarchies. The performance of Gaussian kernel and Laplace kernel are close, because of their similar forms of mapping function. Polynomial kernel performs worse than the other two kernels. Note that the parameter range of λ_1 is set to capture the best performance of K-FAEL.

4.4. Binary constraint model evaluation

To test the performance of the Binary constraint FAEL model (B-FAEL), we first evaluate the sensitivity of λ_3 and μ. In image annotation and gene function prediction datasets, we set $\lambda_3 = \{0, 10, 20, \ldots, 200\}$. In the visual recognition dataset, we set $\lambda_3 = \{0, 10, 20, \ldots, 200\}$. In all datasets, we also evaluate the sensitivity of μ in the range of $\{0, 1, 2, \ldots, 10\}$. Figs. 11–16 depict the performance of B-FAEL regarding λ_3 and μ with the best combination of λ_1.

In Fig. 11, B-FAEL improves both F-measure and HF-measure performance on four sub-hierarchies. As λ_3 increases, the performance first goes up and then becomes stable after reaching a peak. With a small hierarchy size of five nodes, the performance on the "food" hierarchy is basically unchanged. In Fig. 12, compared with FAEL and K-FAEL, the B-FAEL model achieves better performance in HF-measure. On the other hand, as λ_2 becomes larger, the F-

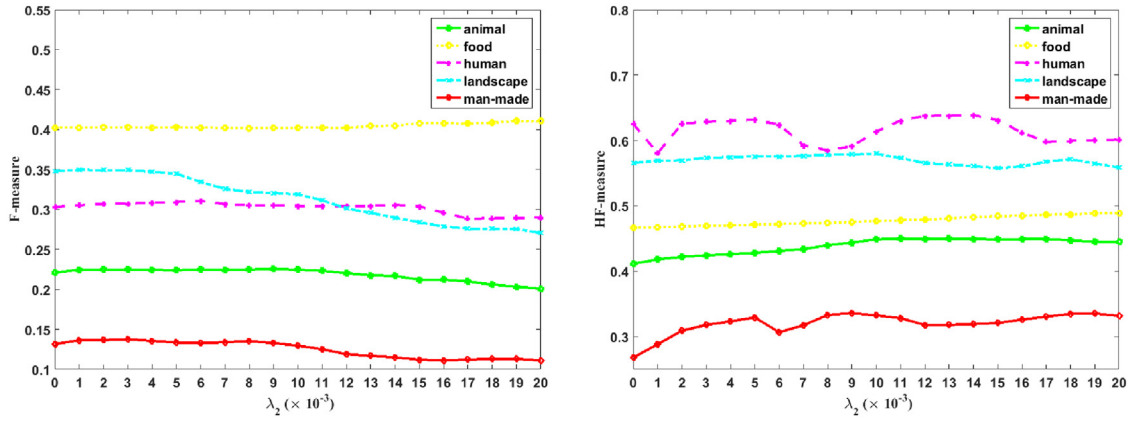

Fig. 6. S-FAEL performance of different λ_2 on the image annotation dataset. (L) F-measure. (R) HF-measure.

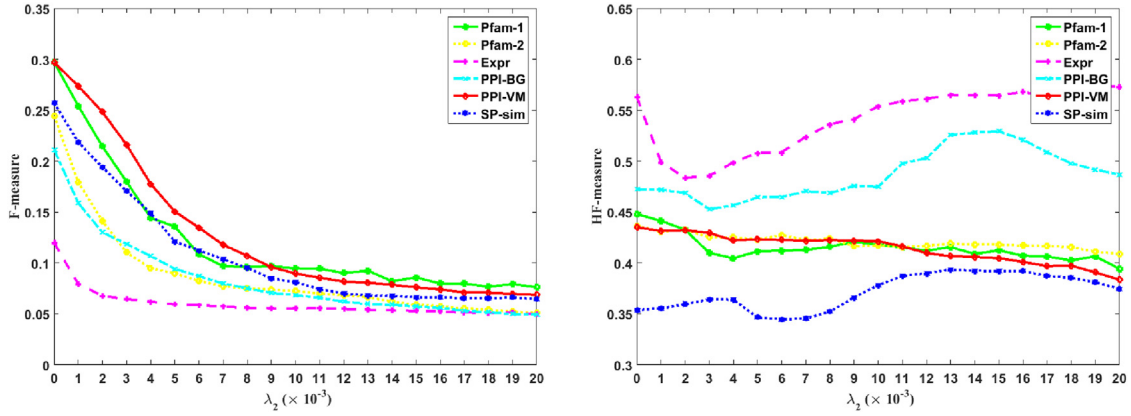

Fig. 7. S-FAEL performance of different λ_2 on the gene function datasets. (L) F-measure. (R) HF-measure.

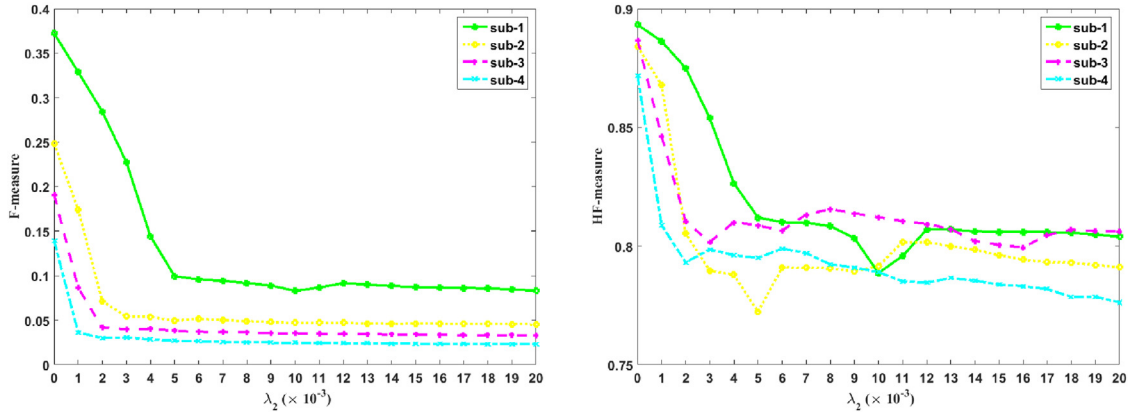

Fig. 8. S-FAEL performance of different λ_2 on the ImageNet sub-hierarchy dataset. (L) F-measure. (R) HF-measure.

measure performance of B-FAEL is worse than that of FAEL and K-FAEL. There are two reasons. First, the binary constraint model enforces the hierarchical consistency, which might weaken the independent discriminative ability of some nodes. Second, the "FunCat" hierarchy has large size and high complexity. With the given features, the binary constraint model cannot optimize both flat and hierarchical performance. In Fig. 13, we observe that B-FAEL does not achieve better results than FAEL under both F-measure and HF-measure. From Figs. 14–16, we can observe similar performances. As we increase the value of μ, the performance on the image annotation dataset is almost stable. In gene function predication datasets, the model achieves better HF-measure while sacrificing F-measure performance. In visual recognition, the performance goes down as we increase the value of μ.

4.5. Overall performance

In this section, we compare our four models with three baseline methods. The values of parameters are learned from inner cross-validation of training data. The results are summarized in Tables 5 and 6.

In Table 5 we can observe that the proposed models perform better than other HMC algorithms. As we know, the classic F-measure is designed for unstructured flat classification problems. Here, it evaluates the average prediction performance of all the

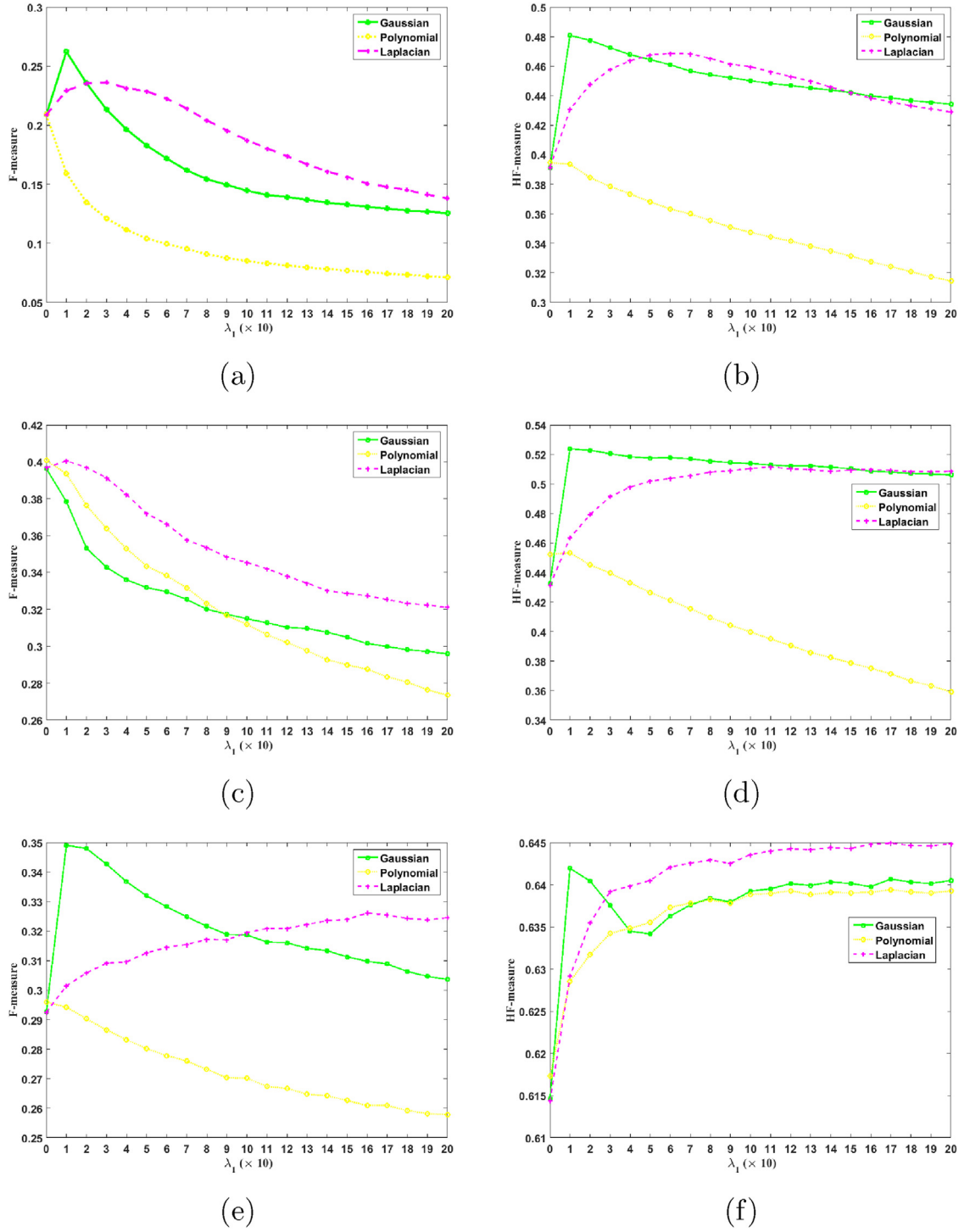

Fig. 9. K-FAEL performance of different λ_1 on the image annotation dataset I. (a)–(b), (c)–(d) and (e)–(f) represent the F-measure and the HF-measure of "animal", "food" and "human", respectively.

nodes. In image annotation dataset, S-FAEL achieves the best result on one small sub-hierarchy ("food"). K-FAEL achieves the best results on two sub-hierarchies ("human" and "landscape") while B-FAEL achieves the best results on the other two sub-hierarchies ("animal" and "man-made"). In gene function prediction, FAEL achieves better performance on three datasets (Pfam-2, Expr, PPI-VM). The results on Pfam-1 and PPI-BG are competitive with the best from TPR-w. K-FAEL achieves the best result on SP-smi. On the visual recognition dataset, we can observe that the baseline methods fail to achieve valid performance on this complex DAG-structured dataset under F-measure. The proposed K-FAEL model obtains the best performance.

In Table 6, we can observe that the best performance is achieved by our models on all datasets. In image annotation datasets, K-FAEL achieves better performance on three sub-hierarchies ("food", "human" and "landscape") while B-FAEL achieves the best performance on the other two sub-hierarchies ("animal" and "man-made"). In gene function prediction datasets,

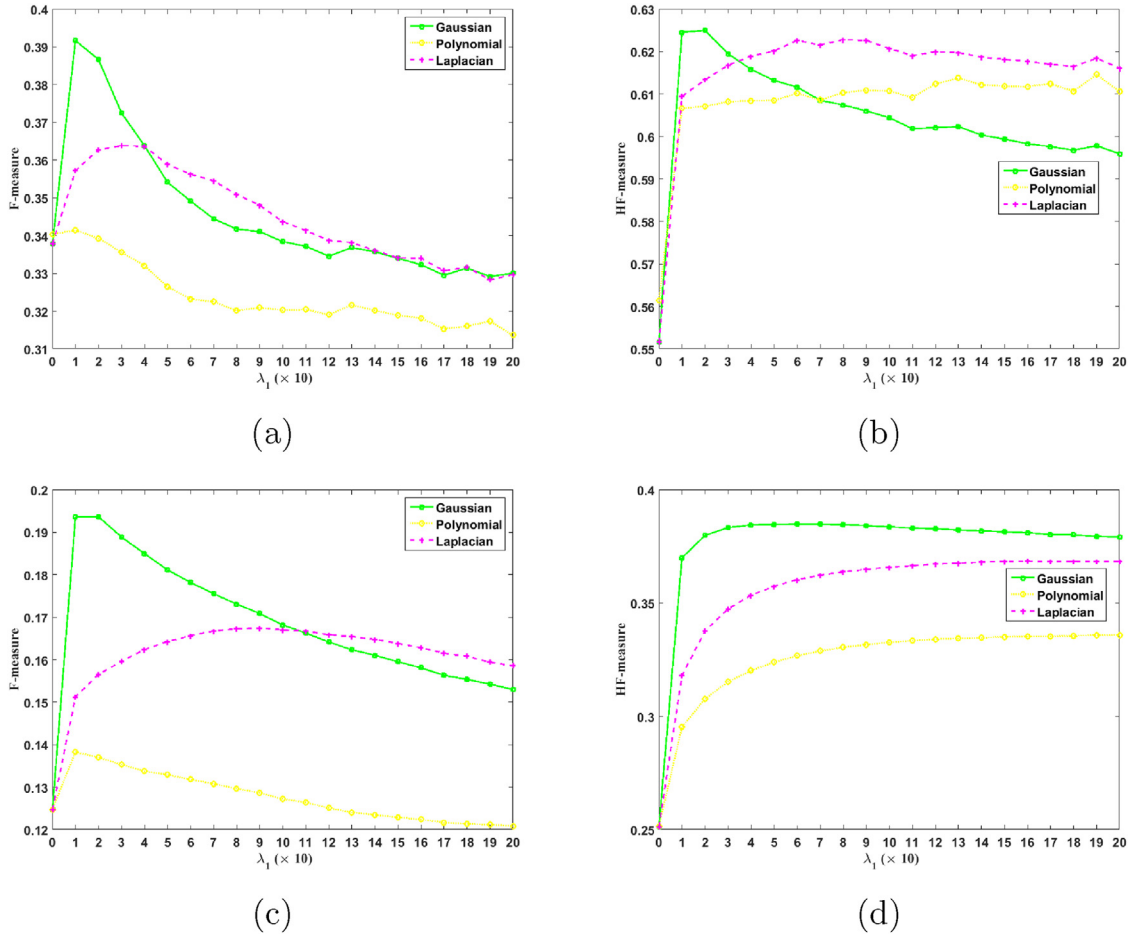

Fig. 10. K-FAEL performance of different λ_1 on the image annotation dataset II. (a)–(b) and (c)–(d) represent the F-measure and the HF-measure of "landscape" and "man-made", respectively.

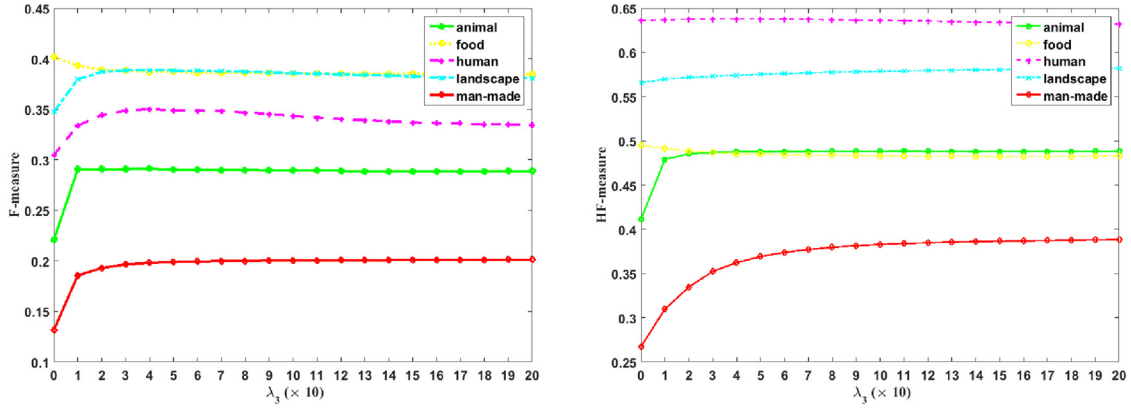

Fig. 11. B-FAEL performance of different λ_3 on the image annotation dataset. (L) F-measure. (R) HF-measure.

K-FAEL performs better than other methods on 4 datasets, except on Expr and PPI-BG where B-FAEL achieves the best performance. On visual recognition dataset, K-FAEL also performs the best on all the sub-hierarchies.

4.6. Statistical analysis

In this section, we perform statistical analysis for the seven methods (TD, TPR, TPR-w, FAEL, S-FAEL, K-FAEL, B-FAEL) over 15 datasets in the above experiments (five from image annotation, six from gene function prediction and four from visual recognition). From [54], we use the Friedman test [55,56] and the two tailed Bonferroni–Dunn test [57] to compare multiple methods over multiple datasets. Let r_i^j represent the rank of the jth of k algorithm on the ith of N datasets. The Friedman test compares the average ranks of different methods, by $R_j = \frac{1}{N}\sum_i r_i^j$. The null-hypothesis states that all the methods are equal so their ranks R_j should be equivalent. The original Friedman statistic [55,56],

$$\mathcal{X}_F^2 = \frac{12N}{k(k+1)}\left[\sum_j R_j^2 - \frac{k(k+1)^2}{4}\right], \tag{16}$$

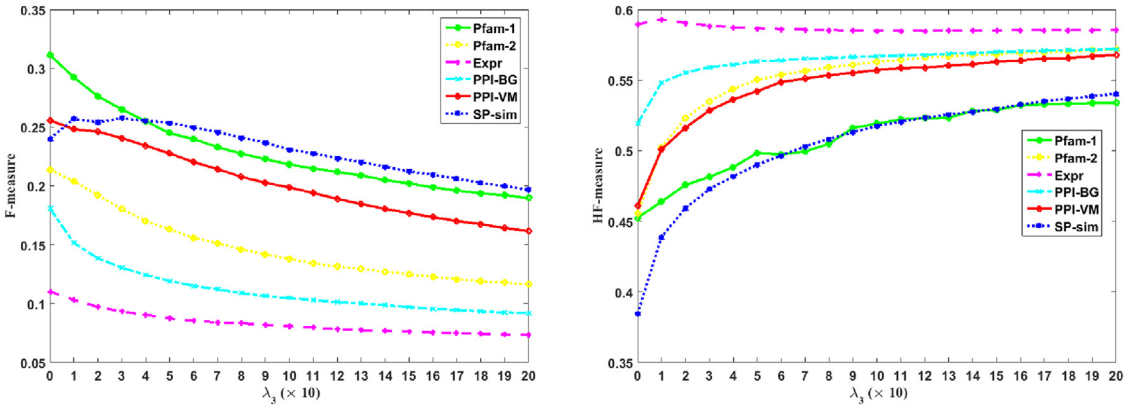

Fig. 12. B-FAEL performance of different λ_3 on the gene function datasets. (L) F-measure. (R) HF-measure.

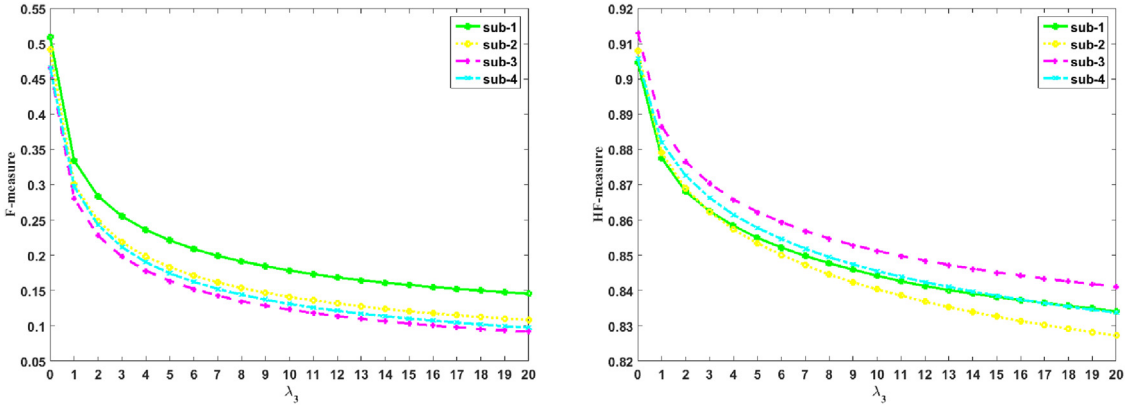

Fig. 13. B-FAEL performance of different λ_3 on the ImageNet sub-hierarchy dataset. (L) F-measure. (R) HF-measure.

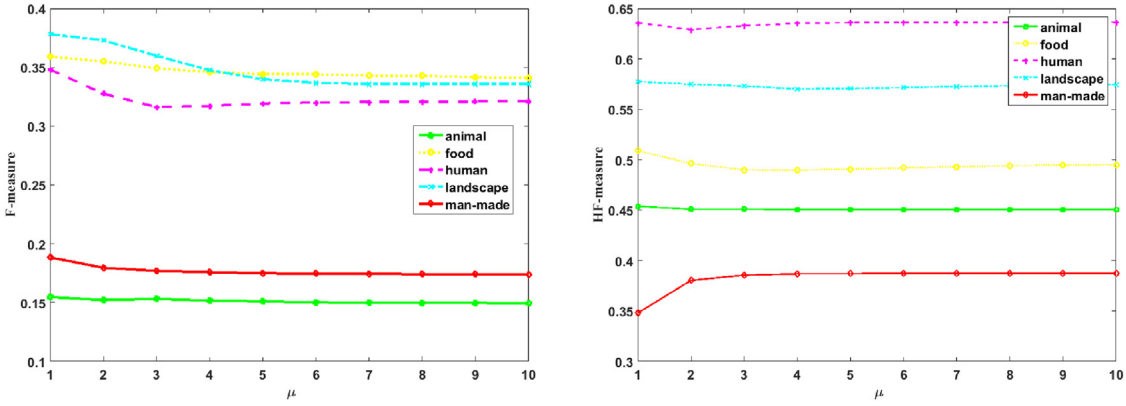

Fig. 14. B-FAEL performance of different μ on the image annotation dataset. (L) F-measure. (R) HF-measure.

is distributed according to χ_F^2 with $k-1$ degree of freedom. Due to its undesirable conservative property, Iman et al. [58] derived a better statistic

$$F_F = \frac{(N-1)\chi_F^2}{N(k-1)-\chi_F^2}, \tag{17}$$

which is distributed according to the F-distribution with $k-1$ and $(k-1)\times(N-1)$ degrees of freedom. First we compute the average ranks of each method; the results are summarized in Table 7. The F_F statistical values of F-measure and HF-measure based on (17) are computed as 25.569 and 118.432. With seven methods and 15 datasets, F_F is distributed with $7-1$ and $(7-1)\times(15-1)=84$ degree of freedom. The critical value of $F(6, 84)$ for $\alpha=0.10$ is $2.762 < 24.050$ or 103.521, so we reject the null-hypothesis. Then, we apply two tailed Bonferroni–Dunn test to compare each pair of methods by the critical difference:

$$CD = q_\alpha\sqrt{\frac{k(k+1)}{6N}}, \tag{18}$$

where q_α is the critical values. If the average rank between two methods is larger than critical difference, the two methods are significantly different. According to Table 5 in [54], the critical value of seven methods when $p=0.10$ is 2.394. From Table 7, we can compute the critical difference $CD = 2.394\sqrt{\frac{7\times8}{6\times15}} = 1.888$. Then we can conclude that, under F-measure, FAEL, K-FAEL, B-FAEL perform significantly better than TD, TPR, TPR-w (the difference between the lowest rank from FAEL and the highest rank from TPR-w,

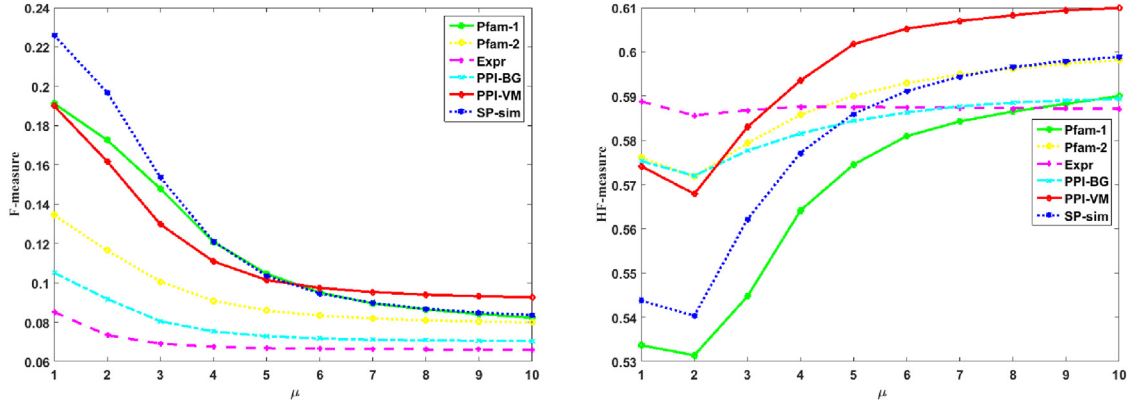

Fig. 15. B-FAEL performance of different μ on the gene function datasets. (L) F-measure. (R) HF-measure.

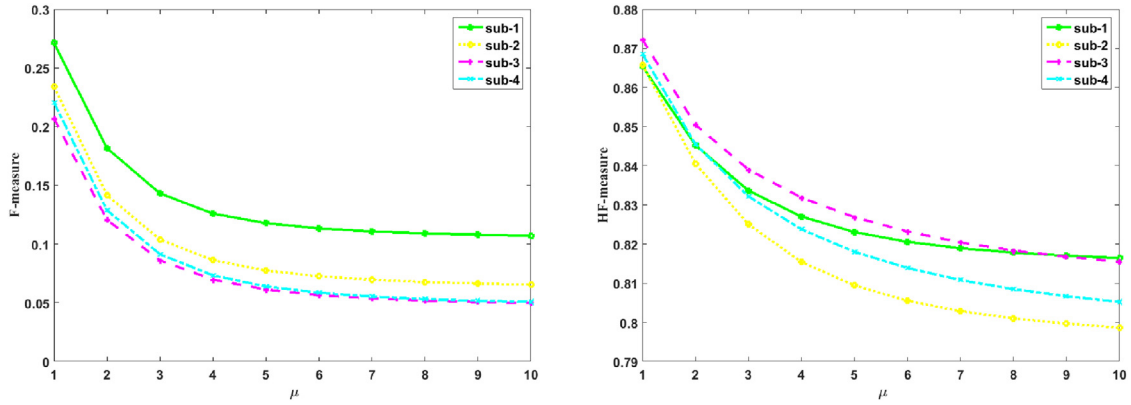

Fig. 16. B-FAEL performance of different μ on the ImageNet sub-hierarchy dataset. (L) F-measure. (R) HF-measure.

Table 5
F-measure performance on different datasets.

Datasets	TD	TPR	TPR-w	FAEL	S-FAEL	K-FAEL	B-FAEL
Animal	0.128	0.137	0.137	0.224	0.220	0.262	**0.291**
Food	0.363	0.364	0.364	0.401	**0.403**	0.401	0.398
Human	0.230	0.231	0.231	0.315	0.303	**0.349**	0.345
Landscape	0.264	0.272	0.272	0.347	0.348	**0.392**	0.387
Man-made	0.069	0.074	0.076	0.134	0.131	0.194	**0.201**
Pfam-1	**0.404**	0.362	**0.404**	0.398	0.297	0.396	0.398
Pfam-2	0.206	0.156	0.220	**0.304**	0.245	0.260	**0.304**
Expr	0.062	0.070	0.077	**0.132**	0.112	0.125	**0.132**
PPI-BG	0.269	0.234	**0.295**	0.281	0.211	0.286	0.281
PPI-VM	0.359	0.261	0.356	**0.395**	0.297	0.371	**0.395**
SP-smi	0.249	0.131	0.254	0.341	0.257	**0.347**	0.341
Sub-1	0.037	0.042	0.067	0.513	0.372	**0.638**	0.513
Sub-2	0.021	0.024	0.041	0.493	0.248	**0.601**	0.493
Sub-3	0.015	0.027	0.032	0.461	0.191	**0.547**	0.461
Sub-4	0.006	0.005	0.023	0.464	0.139	**0.518**	0.464

Table 6
HF-measure performance on different datasets.

Datasets	TD	TPR	TPR-w	FAEL	S-FAEL	K-FAEL	B-FAEL
Animal	0.319	0.327	0.328	0.432	0.411	0.481	**0.488**
Food	0.385	0.386	0.386	0.495	0.466	**0.524**	0.495
Human	0.605	0.606	0.606	0.636	0.625	**0.645**	0.636
Landscape	0.501	0.503	0.504	0.571	0.566	**0.625**	0.582
Man-made	0.178	0.186	0.188	0.281	0.268	0.385	**0.388**
Pfam-1	0.412	0.308	0.413	0.459	0.448	**0.590**	0.534
Pfam-2	0.341	0.268	0.370	0.456	0.436	**0.608**	0.598
Expr	0.117	0.170	0.178	0.590	0.573	0.592	**0.593**
PPI-BG	0.323	0.267	0.349	0.519	0.529	0.582	**0.588**
PPI-VM	0.398	0.280	0.400	0.468	0.435	**0.610**	0.609
SP-smi	0.425	0.226	0.447	0.384	0.394	**0.613**	0.598
Sub-1	0.570	0.413	0.727	0.906	0.893	**0.925**	0.906
Sub-2	0.551	0.359	0.715	0.909	0.884	**0.923**	0.909
Sub-3	0.535	0.500	0.733	0.912	0.887	**0.921**	0.912
Sub-4	0.328	0.219	0.720	0.906	0.872	**0.913**	0.906

$4.633 - 2.567 = 2.066 > 1.888$). S-FAEL performs statistically better than TD and TPR. But the average rank difference between TPR-w and S-FAEL ($4.633 - 4.200 = 0.433$) is smaller than the critical value 1.888, so they are not significantly different. Under HF-measure, K-FAEL and B-FAEL perform statistically better than TD, TPR, TPR-w. The average rank difference between FAEL and TPR-w ($4.933 - 3.067 = 1.866$) is slightly smaller than the critical value 1.888, they are not significantly different. S-FAEL performs statistically better than TD and TPR; there is no significant difference between S-FAEL and TPR-w.

Table 7
Average ranks of each method under F-measure and HF-measure.

Measurements	TD	TPR	TPR-w	FAEL	S-FAEL	K-FAEL	B-FAEL
F-measure	6.100	6.200	4.633	2.567	4.200	1.900	2.267
HF-measure	6.267	6.533	4.933	3.067	4.000	1.267	1.933

5. Conclusion

This paper introduces a novel HMC framework. We build a multi-variable regression model between the global and local predictions of all the nodes. The basic model is extended to the sparse model, the kernel model and the binary constraint model. Our work also raises several potential issues that we plan to address in the future. As the number of classes increases, the proposed fully associative model may suffer from both computation and performance limitations. A large-scale, fully associative weight matrix requires a large amount of discriminative training data. For this problem, we can build the fully associative model for each class branch separately, which will effectively reduce both computation and performance burden. Meanwhile, we use parallel computing techniques in all experiments to reduce computation complexity. In addition, the performance of the local HMC model is also influenced by the thresholds selected for global prediction. A better threshold learning algorithm may help to achieve better performance.

Acknowledgments

This material is based upon work supported by the U.S. Department of Homeland Security under Grant Award Number 2015-ST-061-BSH001. This grant is awarded to the Borders, Trade, and Immigration (BTI) Institute: A DHS Center of Excellence led by the University of Houston, and includes support for the project “Image and Video Person Identification in an Operational Environment: Phase I” awarded to the University of Houston. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the U.S. Department of Homeland Security.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at 10.1016/j.patcog.2017.05.007.

References

[1] T. Li, S. Zhu, M. Ogihara, Hierarchical document classification using automatically generated hierarchy, J. Intell. Inf. Syst. 29 (2) (2007) 211–230.
[2] I. Dimitrovski, D. Kocev, S. Loskovska, S. Džeroski, Hierarchical annotation of medical images, Pattern Recognit. 44 (10) (2011) 2436–2449.
[3] N. Cesa-Bianchi, M. Re, G. Valentini, Synergy of multi-label hierarchical ensembles, data fusion, and cost-sensitive methods for gene functional inference, Mach. Learn. 88 (1–2) (2012) 209–241.
[4] P.N. Robinson, M. Frasca, S. Köhler, M. Notaro, M. Re, G. Valentini, A hierarchical ensemble method for dag-structured taxonomies, in: Multiple Classifier Systems, Springer, 2015, pp. 15–26.
[5] C.J. Silla, A.A. Freitas, A survey of hierarchical classification across different application domains, Data Min. Knowl. Discov. 22 (1–2) (2011) 31–72.
[6] C.N. Silla, A.A. Freitas, Novel top-down approaches for hierarchical classification and their application to automatic music genre classification, in: Proceedings of IEEE International Conference on Systems, Man and Cybernetics, San Antonio, Texas, USA, 2009, pp. 3499–3504.
[7] T. Fagni, F. Sebastiani, On the selection of negative examples for hierarchical text categorization, in: Proceedings of Language and Technology Conference, Poznań, Poland, 2007, pp. 24–28.
[8] L. Zhang, S.K. Shah, I.A. Kakadiaris, Fully associative ensemble learning for hierarchical multi-label classification, in: Proceedings of British Machine Vision Conference, Nottingham, UK, 2014.
[9] K. Wang, S. Zhou, Y. He, Hierarchical classification of real life documents, in: Proceedings of SIAM International Conference on Data Mining, Chicago, IL, USA, 2001, pp. 1–16.
[10] C. Vens, J. Struyf, L. Schietgat, S. Džeroski, H. Blockeel, Decision trees for hierarchical multi-label classification, Mach. Learn. 73 (2) (2008) 185–214.
[11] W. Bi, J.T. Kwok, Multi-label classification on tree-and DAG-structured hierarchies, in: Proceedings of International Conference on Machine Learning, Bellevue, WA, 2011, pp. 17–24.
[12] I. Dimitrovski, D. Kocev, S. Loskovska, S. Džeroski, Hierarchical classification of diatom images using ensembles of predictive clustering trees, Ecol. Inform. 7 (1) (2012) 19–29.
[13] R. Cerri, R.C. Barros, A.C. de Carvalho, A genetic algorithm for hierarchical multi-label classification, in: Proceedings of Annual ACM Symposium on Applied Computing, Trento, Italy, 2012, pp. 250–255.
[14] R.C. Barros, R. Cerri, A.A. Freitas, A.C. de Carvalho, Probabilistic clustering for hierarchical multi-label classification of protein functions, in: Proceedings of Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Prague, Czech Republic, 2013, pp. 385–400.
[15] S. Dumais, H. Chen, Hierarchical classification of web content, in: Proceedings of ACM/SIGIR International Conference on Research and Development in Information Retrieval, Athens, Greece, 2000, pp. 256–263.
[16] Z. Barutcuoglu, C. DeCoro, Hierarchical shape classification using Bayesian aggregation, in: Proceedings of IEEE International Conference on Shape Modeling and Applications, Matsushima, Japan, 2006.
[17] N. Cesa-Bianchi, C. Gentile, L. Zaniboni, Incremental algorithms for hierarchical classification, J. Mach. Learn. Res. 7 (January) (2006) 31–54.
[18] N. Alaydie, C.K. Reddy, F. Fotouhi, Exploiting label dependency for hierarchical multi-label classification, in: Proceedings of Pacific-Asia Conference on Knowledge Discovery and Data Mining, Kuala Lumpur, Malaysia, 2012, pp. 294–305.
[19] Z. Ren, M.-H. Peetz, S. Liang, W. Van Dolen, M. De Rijke, Hierarchical multi-label classification of social text streams, in: Proceedings of International ACM SIGIR Conference on Research & Development in Information Retrieval, Queensland, Australia, 2014, pp. 213–222.
[20] R. Cerri, R.C. Barros, A.C. De Carvalho, Hierarchical multi-label classification using local neural networks, J. Comput. Syst. Sci. 80 (1) (2014) 39–56.
[21] P. Vateekul, M. Kubat, K. Sarinnapakorn, Hierarchical multi-label classification with SVMs: a case study in gene function prediction, Intell. Data Anal. 18 (4) (2014) 717–738.
[22] G. Valentini, True path rule hierarchical ensembles for genome-wide gene function prediction, IEEE/ACM Trans. Comput. Biol. Bioinf. 8 (3) (2011) 832–847.
[23] G. Valentini, S. Köhler, M. Re, M. Notaro, P.N. Robinson, Prediction of human gene-phenotype associations by exploiting the hierarchical structure of the human phenotype ontology, in: Bioinformatics and Biomedical Engineering, Springer, 2015, pp. 66–77.
[24] X. Jiang, N. Nariai, M. Steffen, S. Kasif, E. Kolaczyk, Integration of relational and hierarchical network information for protein function prediction, BMC Bioinform. 9 (1) (2008) 350.
[25] P.N. Bennett, N. Nguyen, Refined experts: improving classification in large taxonomies, in: Proceedings of ACM/SIGIR International Conference on Research and Development in Information Retrieval, Boston, MA, USA, 2009, pp. 11–18.
[26] Y. Guan, C.L. Myers, D.C. Hess, Z. Barutcuoglu, A. Caudy, O.G. Troyanskaya, Predicting gene function in a hierarchical context with an ensemble of classifiers, Genome Biol. 9 (Suppl 1) (2008) S3.
[27] H.J. Escalante, C.A. Hernández, J.A. Gonzalez, A. López-López, M. Montes, E.F. Morales, L.E. Sucar, L. Villaseñor, M. Grubinger, The segmented and annotated IAPR TC-12 benchmark, Comput. Vision Image Understanding 114 (4) (2010) 419–428.
[28] S. Ji, L. Tang, S. Yu, J. Ye, A shared-subspace learning framework for multi-label classification, ACM Trans. Knowl. Discovery Data (TKDD) 4 (2) (2010) 8.
[29] X. Zhu, X. Li, S. Zhang, Block-row sparse multiview multilabel learning for image classification, IEEE Trans. Cybern. 46 (2) (2016) 450–461.
[30] Y. Luo, D. Tao, B. Geng, C. Xu, S.J. Maybank, Manifold regularized multitask learning for semi-supervised multilabel image classification, IEEE Trans. Image Process. 22 (2) (2013) 523–536.
[31] Y. Luo, T. Liu, D. Tao, C. Xu, Multiview matrix completion for multilabel image classification, IEEE Trans. Image Process. 24 (8) (2015) 2355–2368.
[32] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: Proceedings of Neural Information Processing Systems, Lake Tahoe, NV, 2012, pp. 1097–1105.
[33] C. Szegedy, A. Toshev, D. Erhan, Deep neural networks for object detection, in: Proceedings of Advances in Neural Information Processing Systems, Lake Tahoe, Nevada, 2013, pp. 2553–2561.
[34] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in: Proceedings of Computer Vision and Pattern Recognition, Boston, MA, 2015.
[35] Y. Wei, W. Xia, M. Lin, J. Huang, B. Ni, J. Dong, Y. Zhao, S. Yan, HCP: a flexible CNN framework for multi-label image classification, IEEE Trans. Pattern Anal. Mach. Intell. 38 (9) (2016) 1901–1907.
[36] J. Wang, Y. Yang, J. Mao, Z. Huang, C. Huang, W. Xu, CNN-RNN: a unified framework for multi-label image classification, in: Proceedings of Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, 2016, pp. 2285–2294.
[37] R.-W. Zhao, J. Li, Y. Chen, J.-M. Liu, Y.-G. Jiang, X. Xue, Regional gating neural networks for multi-label image classification, in: Proceedings of British Machine Vision Conference, York, UK, 6, 2016.
[38] J. Zhou, L. Yuan, J. Liu, J. Ye, A multi-task learning formulation for predicting disease progression, in: Proceedings of ACM/SIGKDD International Conference on Knowledge Discovery and Data Mining, San Diego, CA, USA, 2011, pp. 814–822.
[39] A. Charuvaka, H. Rangwala, Multi-task learning for classifying proteins using dual hierarchies., in: Proceedings of IEEE International Conference on Data Mining, Brussels, Belgium, 2012, pp. 834–839.
[40] L. Jacob, F. Bach, J.P. Vert, Clustered multi-task learning: a convex formulation., in: Proceedings of Annual Conference on Neural Information Processing Systems, Vancouver, B.C., Canada, 2008, pp. 745–752.
[41] J. Zhou, J. Liu, A.N. Vaibhav, J. Ye, Modeling disease progression via multi-task learning, Neuroimage 78 (2013) 233–248.

[42] S. Kim, E.P. Xing, Tree-guided group lasso for multi-response regression with structured sparsity, with an application to eQTL mapping, Ann. Appl. Stat. 6 (3) (2012) 1095–1117.
[43] S. Ji, L. Yuan, Y. Li, Z. Zhou, S. Kumar, J. Ye, Drosophila gene expression pattern annotation using sparse features and term-term interactions, in: Proceedings of ACM/SIGKDD International Conference on Knowledge Discovery and Data Mining, Paris, France, 2009, pp. 407–416.
[44] C. Xu, T. Liu, D. Tao, C. Xu, Local Rademacher complexity for multi-label learning, IEEE Trans. Image Process. 25 (3) (2016a) 1495–1507.
[45] C. Xu, D. Tao, C. Xu, Robust extreme multi-label learning, in: Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016b, pp. 13–17.
[46] H.-F. Yu, P. Jain, P. Kar, I.S. Dhillon, Large-scale multi-label learning with missing labels., in: Proceedings of International Conference on Machine Learning, Beijing, China, 2014, pp. 593–601.
[47] Y. Nesterov, Introductory Lectures on Convex Optimization: A Basic Course, 87, Springer Science & Business Media, 2013.
[48] J. Liu, S. Ji, J. Ye, SLEP: sparse learning with efficient projections, Arizona State University (2009).
[49] S. An, W. Liu, S. Venkatesh, Face recognition using kernel ridge regression, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis, MN, USA, 2007, pp. 1–7.
[50] K.B. Petersen, M.S. Pedersen, et al., The Matrix Cookbook, Technical University of Denmark 7 (2008) 15.
[51] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet: a large-scale hierarchical image database, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Miami, Florida, USA, 2009, pp. 248–255.
[52] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, T. Darrell, Caffe: convolutional architecture for fast feature embedding, in: Proceedings of ACM International Conference on Multimedia, Orlando, Florida, USA, 2014, pp. 675–678.
[53] K. Verspoor, J. Cohn, S. Mniszewski, C. Joslyn, A categorization approach to automated ontological function annotation, Protein Sci. 15 (6) (2006) 1544–1549.
[54] J. Demšar, Statistical comparisons of classifiers over multiple data sets, J. Mach. Learn. Res. 7 (January) (2006) 1–30.
[55] M. Friedman, The use of ranks to avoid the assumption of normality implicit in the analysis of variance, J. Am. Stat. Assoc. 32 (200) (1937) 675–701.
[56] M. Friedman, A comparison of alternative tests of significance for the problem of m rankings, Ann. Math. Stat. 11 (1) (1940) 86–92.
[57] O.J. Dunn, Multiple comparisons among means, J. Am. Stat. Assoc. 56 (293) (1961) 52–64.
[58] R.L. Iman, J.M. Davenport, Approximations of the critical region of the fbietkan statistic, Commun. Stat.-Theor. Methods 9 (6) (1980) 571–595.

Lingfeng Zhang received the B.S. degree in Mathematics and the M.S. in computer science from the Chongqing University, Chongqing, China. He is currently a Ph.D. student in Department of Computer Science, University of Houston, Houston, TX, USA. He joined Computational Biomedicine Laboratory in 2012. His current research interests include machine learning, deep learning, image processing, computer vision, big data analysis.

Shishir K. Shah received the B.S. degree in mechanical engineering and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Texas, Austin, TX, USA. He is currently a Professor with the Department of Computer Science, University of Houston, Houston, TX, USA. He joined the department in 2005. He has co-edited one book and authored numerous papers on object recognition, sensor fusion, statistical pattern analysis, biometrics, and video analytics. He directs research at the Quantitative Imaging Laboratory. His current research interests include fundamentals of computer vision, pattern recognition, and statistical methods in image and data analysis with applications in multimodality sensing, video analytics, object recognition, biometrics, and microscope image analysis.

Ioannis A. Kakadiaris serves as the Director of the Borders, Trade, and Immigration Institute, a Department of Homeland Security Center of Excellence led by the University of Houston (UH). As director for BTI Institute, Ioannis oversees multiple projects, undertaken with seventeen partners across nine states, which provide homeland security enterprise education and workforce development and which study complex, multi-disciplinary issues related to flows of people, goods, and data across borders. A Hugh Roy and Lillie Cranz Cullen Distinguished University Professor of Computer Science, Ioannis is also an international expert in facial recognition and data/video analytics. He earned his B.S. in physics at the University of Athens in Greece, his M.S. in computer science from Northeastern University, and his Ph.D. in computer science at the University of Pennsylvania. In addition to twice winning the UH Computer Science Research Excellence Award, Ioannis has been recognized for his work with several distinguished honors, including the NSF Early Career Development Award, the Schlumberger Technical Foundation Award, the UH Enron Teaching Excellence Award, and the James Muller Vulnerable Plaque Young Investigator Prize.