# 由粗到细的分层特征选择

刘浩阳[1,2],林耀进[1,2],刘景华[3],吴镒澦[1,2],毛 煜[1,2],李绍滋[4]
(1. 闽南师范大学计算机学院,福建漳州 363000; 2. 数据科学与智能应用福建省高等学校重点实验室(闽南师范大学),福建漳州 363000;
3. 华侨大学计算机科学与技术学院,福建厦门 361021; 4. 厦门大学人工智能系,福建厦门 361005)

**摘 要:** 利用数据类别间层次结构关系进行分类学习任务广泛存在于疾病诊断、图像标注等领域.然而,数据特征空间的高维性,使得分层分类学习面临着时间复杂度高和存储负担大等问题.另外,现有研究工作都假设训练集标记粒度是充分细化,与实际分层分类学习中划分细粒度标记代价高,类别标记间存在语义歧义性等矛盾.为解决上述问题,提出一种由粗到细的分层特征选择算法.该算法考虑类内一致性和兄弟节点间的差异性以选择有代表性特征,同时在特征选择的过程中实现预测训练样本未知的细粒度标记.在7个基准数据集上的实验结果表明,所提算法的分类性能优于一些先进的对比算法,且能处理标记粒度不够细化的情况.

**关键词:** 特征选择; 分层分类; 标记层次结构; 标记粒度; 递归正则化; 稀疏优化; 全局最优解
**中图分类号:** TP181 **文献标识码:** A **文章编号:** 0372-2112(2022)11-2778-12
**电子学报 URL:**http://www.ejournal.org.cn **DOI:**10.12263/DZXB.20211263

# Hierarchical Feature Selection from Coarse to Fine

LIU Hao-yang[1,2], LIN Yao-jin[1,2], LIU Jing-hua[3], WU Yi-lin[1,2], MAO Yu[1,2], LI Shao-zi[4]
(1. *School of Computer Science, Minnan Normal University, Zhangzhou, Fujian* 363000, *China*;
2. *Key Laboratory of Data Science and Intelligence Application (Minnan Normal University), Fujian Province University, Zhangzhou, Fujian* 363000, *China*; 3. *Department of Computer Science and Technology, Huaqiao University, Xiamen, Fujian* 361021, *China*;
4. *Department of Artificial Intelligence, Xiamen University, Xiamen, Fujian* 361005, *China*)

**Abstract:** The task of classification learning using hierarchy of categories in data exists widely in many practical applications such as disease diagnosis, image annotation, etc. However, the high dimensionality of data feature space makes hierarchical classification learning confront problems such as high time and space complexity. In addition, existing research works assume that the training set label granularity is sufficiently fine-grained, which is contradictory to the actual hierarchical classification learning, i.e., dividing fine-grained labels is costly and ambiguity exists among category labels. To solve the above problems, we propose a coarse-to-fine hierarchical feature selection algorithm. We consider intra-class consistency and inter-sibling variability to select representative features and the unknown fine-grained labels of the training samples are predicted during feature selection. Experimental results on seven benchmark datasets show that the proposed algorithm outperforms some advanced comparative algorithms in classification performance and can handle the case where the label granularity is not fine-grained enough.

**Key words:** feature selection; hierarchical classification; label hierarchical structure; label granularity; recursive regularization; sparse optimization; global optimal solution

## 1 引言

在大数据时代,分类学习任务中样本数量、特征维数、样本类别的快速增长无可避免[1].例如,在 ImageNet[2]中描述图像的特征有数千个,样本的类标记更是达到数万个.粒计算作为一种在不同粒度级别上分析、理解、表示和解决现实世界问题的工具,能够将人类智能的多粒度思维模式与问题解决模式相结合来有效地处理大数据[3],已经受到了许多研究人员的关注.Yao[4]等人总结粒计算的基本概念,并给出粒计算中各种研究流派的分类和描述.Bargiela[5]等人提出一种以人为中心的信息处理的粒计算理论,通过构造算法和非算

收稿日期:2021-09-14;修回日期:2021-11-02;责任编辑:覃怀银
基金项目:国家自然科学基金面上项目(No.62076116);福建省自然科学基金重点项目(No.2021J02049)

法信息处理的结构化组合,来模仿人类从信息中智能地合成知识.基于此,在面对大规模数据的分类学习问题时,研究人员从人类认知事物的不同层次、不同角度作为粒度,将这些数据类别按照从抽象到具体的方式组成一个层次化结构进行研究[6].在具有层次化结构的标记空间中,样本对应的标记也自然呈现出粗粒度与细粒度之分[7],如一只动物属于东北虎(细粒度),同时也属于猫科动物(粗粒度).因此,样本的标记粒度越粗则所表示的语义越抽象、范围越广;相反,其标记粒度越细所表示的语义越具体、范围越窄.

根据对分层分类的相关研究[8],分层分类方法大致分为:平面分类方法、局部分类方法和全局分类方法.平面分类方法完全忽略标记的层次结构信息,相当于传统的分类方法.局部分类方法则为标记的层次结构中每个内部节点训练一个分类器,并通常以自上而下的方式进行预测.与局部分类方法不同,全局分类方法将层次结构作为一个整体来考虑,并训练统一的分类器.

随着分层分类学习中类别数量的迅速增加,从原来的若干类,到几十类,发展到现有的数万类学习任务.这些超大规模类别条件下的分层分类学习比以往传统的平面分类学习面临着更加严重的维数灾难问题.虽然传统平面特征选择算法可以从高维特征中挑选具有区分多个类别能力的特征子集,但是当类别数量大且存在层次化语义结构关系时,如何挑选出具有判别性的特征子集成为一个挑战.

为了应对上述挑战,已有研究[9~11]利用层次结构的特性,为层次结构数据在每一个内部节点上分别评估特征,以降低分层分类任务中的特征维度,并且提高分类准确率.然而,这些工作只是在分类任务中独立地为每个内部节点选择不同特征子集,并未将类别间的层次信息嵌入到特征选择任务中.类别的层次结构间隐含着依赖关系,如父子关系、兄弟关系等.近年来,许多研究人员[12~15]更深入地考虑层次化结构数据类别间的关联,更充分地利用数据隐含的层次信息提高分层任务的准确性,并且拓宽应用场景.然而,现有工作都未考虑层次化数据样本细粒度标记的稀缺问题.

在文本分类、目标识别、基因功能预测等实际应用场景中,获得标记到细粒度类别的样本面临着标注者领域知识窄,标注成本昂贵、周期长、错误率较高等问题.因而,获取完全细化的标记矩阵是不切实际的,分层特征选择面临着数据集中样本细粒度标记的稀缺性.在无法获得样本所有细粒度标记信息情况下,若利用样本的粗粒度信息进行特征选择,则不仅有助于降低获取样本标记的成本,还能够避免错误或缺失的细粒度标记带来的负面影响.

针对上述问题,本文提出针对样本标记粒度不够细化下的一种由粗到细的分层特征选择模型.首先,基于稀疏学习的方法构建模型的基本框架.其次,惩罚兄弟节点所选特征之间的相似度来保证兄弟节点间特征子集的差异.最后,使用线性回归模型来预测样本的细粒度标记以挖掘更多信息.此外,加入一致性正则项使特征上相似的样本输出相同的标记,从而保证预测的准确率.为了评估所提方法的有效性,与近几年提出的分层特征选择算法在7个基准分层数据集上进行对比,并尝试只用粗粒度信息进行特征选择,实验结果验证了所提方法的有效性.

## 2　由粗到细的分层特征选择

在本节中,首先对问题进行陈述.然后,介绍分层特征选择模型的基本框架.接着,讨论如何针对样本标记粒度不够细化的问题利用层次信息进行建模,获取更多未知的细粒度信息以增强模型在复杂情况下的表现.最后,给出模型的优化算法.

### 2.1　问题陈述

在大规模层次结构数据中,层次结构一般分为树结构和图结构,本文关注的是前者.层次树结构可以用有序对$(Y,\prec)$来表示,其中$Y$是一个包括所有类的集合,$\prec$表示继承关系,该关系有如下特性[16]:

(1) 反对称性:如果$i\prec j,\forall i,j\in Y$则$j\nprec i$.

(2) 反自反性:$\forall i\in Y$有$i\nprec i$.

(3) 传递性:对$\forall i,j,k\in Y$,如果$i\prec j$且$j\prec k$,有$i\prec k$.

树结构中子节点对于它们的父节点或是祖先节点而言,其表示的类别更为精细,相对应称为细粒度节点.作为层次结构中粒度最细的叶子节点的样本标记相较于其他节点,获取难度大得多.因此,在样本标记粒度不够细化的情况中,样本标记往往标记到叶子节点的父节点,而非其他祖先节点.因此,本文将针对未划分叶子节点标记的数据集进行建模.

为了更好地介绍模型,预先定义本文用到的符号.将层次结构中所有非叶子节点统称为内部节点,定义其数量为$N$+1,可以将每个内部节点的样本矩阵定义为$\boldsymbol{X}_0,\boldsymbol{X}_1,\cdots,\boldsymbol{X}_N$,其中$\boldsymbol{X}_0$代表根节点所对应的样本矩阵.令$\boldsymbol{X}_i$为内部节点$i$的样本矩阵,它的每个元素$x_{jk}\in\mathbf{R}$($1\leqslant j\leqslant n_i$,$1\leqslant k\leqslant m$),其中$n_i$和$m$分别是样本个数和特征的数量.类似地,定义$\boldsymbol{Y}_0,\boldsymbol{Y}_1,\cdots,\boldsymbol{Y}_N$为内部节点$i$对应的类别标记矩阵.令$\boldsymbol{Y}_i$为内部节点$i$的标记矩阵,它的每个元素$y_{jk}\in\{0,1\}$($1\leqslant j\leqslant n_i$,$1\leqslant k\leqslant d$),其中$d$是内部节点中类别标记数的最大值.最后,为每个内部节点$i$计算权重矩阵$\boldsymbol{W}_i$,它的每个元素$w_{jk}\in\mathbf{R}$($1\leqslant j\leqslant$

$m_i, 1\leqslant k\leqslant d$).

### 2.2 所提模型

在特征选择中,稀疏学习方法已被证明是一种有效的手段[17]. 该方法最小化拟合误差,并使用稀疏正则化项约束特征系数. 它的模型通常形式化为

$$\min_{\boldsymbol{W}} L(\boldsymbol{W};\ \boldsymbol{X},\boldsymbol{Y})+\lambda\Gamma(\boldsymbol{W}) \tag{1}$$

其中,损失项$L$有最小二乘损失、铰链损失或逻辑损失等. 本文采用最小二乘损失以便于求解,则损失函数定义为

$$L(\boldsymbol{W};\ \boldsymbol{X},\boldsymbol{Y})=\left\|\boldsymbol{X}_i\boldsymbol{W}_i-\boldsymbol{Y}_i\right\|_{\mathrm{F}}^2 \tag{2}$$

尽管$\ell_{2,0}$范数有着相当理想的特征稀疏性,但由于它非凸非光滑的性质,在求解上有很大难度. 相比之下,$\ell_{2,1}$范数既能满足特征稀疏性的要求,并且是凸的,容易全局优化[18]. 定义非负常数$\lambda$,用来调整稀疏程度. 综上,式(1)改写为

$$\min_{\boldsymbol{W}}\sum_{i=0}^{N}\left(\left\|\boldsymbol{X}_i\boldsymbol{W}_i-\boldsymbol{Y}_i\right\|_{\mathrm{F}}^2+\lambda\left\|\boldsymbol{W}_i\right\|_{2,1}\right) \tag{3}$$

在分层特征选择任务中,通常为每一个内部类标记节点选择特征子集,尽管内部兄弟节点具有相同的父节点,但它们的后代属于不同的子树. 因此,为区分各自子节点,互为兄弟的节点所选择的特征应该是不同的. 通过利用核依赖度量将变量投影到重构核希尔伯特空间,可以测量出原始分布之间的高阶联合矩[19],故使用下式来惩罚该节点与其兄弟节点上所选特征之间的依赖关系:

$$\mathrm{HSIC}(\boldsymbol{W}_i,\boldsymbol{W}_l)=\mathrm{tr}(\boldsymbol{W}_i\boldsymbol{W}_i^{\mathrm{T}}\boldsymbol{H}\boldsymbol{W}_l\boldsymbol{W}_l^{\mathrm{T}}) \tag{4}$$

其中,$1\leqslant l\leqslant\left|S_i\right|$,$S_i$表示内部节点$i$的所有兄弟节点的集合. 式中$\boldsymbol{H}=\boldsymbol{I}-\frac{1}{n_i}\mathbf{1}_{n_i}\mathbf{1}_{n_i}^{\mathrm{T}}\in\mathbf{R}^{n_i\times m}$,$\mathbf{1}_{n_i}$是由$n_i$个1组成的列向量.

样本类别标记粒度不够细化的情况损失了细粒度标记所提供的信息,使得特征选择更加困难. 所以,解决标记粒度不够细化问题的关键点在于如何挖掘出更多的细粒度信息. 通过式$\tilde{\boldsymbol{Y}}_i=\boldsymbol{X}_i\boldsymbol{W}_i$预测细粒度标记,并在每一次迭代的过程中不断调整. 另外,对于单个样本,在同一层级的标记最多只有一个,为了避免产生平凡解,令

$$y_i^{jk}=\begin{cases}1,\ \text{if}\ \tilde{y}_i^{jk}=\tilde{y}_i^{j\max}\\ 0,\ \text{otherwise}\end{cases},\ \tilde{y}_i^{j\max}=\max_{1\leqslant k\leqslant d}\tilde{y}_i^{jk} \tag{5}$$

同时,为提高预测细粒度标记的准确率,根据特征值相近的两个样本具有一致类别标记的准则,在模型中加入一致性正则项,保证特征上相似的样本输出相同的标记:

$$\mathrm{tr}(\boldsymbol{W}_i^{\mathrm{T}}\boldsymbol{X}_i^{\mathrm{T}}\boldsymbol{L}\boldsymbol{X}_i\boldsymbol{W}_i) \tag{6}$$

其中,$\boldsymbol{L}$是Laplacian矩阵.

结合式(3)、式(4)和式(6),最终模型定义为

$$\begin{aligned}F(\boldsymbol{W}_0,\boldsymbol{W}_1,\cdots,\boldsymbol{W}_N)=&\min_{\boldsymbol{W}_0,\boldsymbol{W}_1,\cdots,\boldsymbol{W}_N}\sum_{i=0}^{N}\left(\left\|\boldsymbol{X}_i\boldsymbol{W}_i-\boldsymbol{Y}_i\right\|_{\mathrm{F}}^2+\lambda\left\|\boldsymbol{W}_i\right\|_{2,1}\right.\\&\left.+\alpha\mathrm{tr}(\boldsymbol{W}_i^{\mathrm{T}}\boldsymbol{X}_i^{\mathrm{T}}\boldsymbol{L}\boldsymbol{X}_i\boldsymbol{W}_i)\right)+\beta\sum_{i=1}^{N}\sum_{l\in S_i}\mathrm{HSIC}(\boldsymbol{W}_i,\boldsymbol{W}_l)\end{aligned} \tag{7}$$

其中,参数$\alpha$和$\beta$都是非负常数,分别控制着一致性和兄弟正则化.

### 2.3 模型优化与算法伪代码

从式(7)可知,最终目标是得到每个内部节点的权重矩阵$\boldsymbol{W}$. 由于$\ell_{2,1}$的非光滑性,可根据文献[18]进行推导:

$$\frac{\partial\left\|\boldsymbol{W}_i\right\|_{2,1}}{\partial\boldsymbol{W}_i}=\frac{\partial\mathrm{tr}(\boldsymbol{W}_i^{\mathrm{T}}\boldsymbol{D}\boldsymbol{W}_i)}{\partial\boldsymbol{W}_i}=2\boldsymbol{D}_i\boldsymbol{W}_i \tag{8}$$

其中,$\boldsymbol{D}_i\in\mathbf{R}^{d\times d}$是对角矩阵,其第$j$个对角元素是$D_i^{jj}=\frac{1}{2\left\|\boldsymbol{w}_i^j\right\|_2}$.

因此,令$\boldsymbol{U}_l=\boldsymbol{H}\boldsymbol{W}_l\boldsymbol{W}_l^{\mathrm{T}}\boldsymbol{H}$,通过将内部节点$i$的权值矩阵$\boldsymbol{W}_i$的导数设置为0,推导式(7)可以得到:

$$\begin{aligned}\frac{\partial F}{\partial\boldsymbol{W}_i}=&2\boldsymbol{X}_i^{\mathrm{T}}(\boldsymbol{X}_i\boldsymbol{W}_i-\boldsymbol{Y}_i)+2\lambda\boldsymbol{D}_i\boldsymbol{W}_i\\&+\alpha\boldsymbol{X}_i^{\mathrm{T}}(\boldsymbol{L}+\boldsymbol{L}^{\mathrm{T}})\boldsymbol{X}_i\boldsymbol{W}_i+\beta\sum_{l\in S_i}(\boldsymbol{U}_l+\boldsymbol{U}_l^{\mathrm{T}})\boldsymbol{W}_i\\=&(2\boldsymbol{X}_i^{\mathrm{T}}\boldsymbol{X}_i+2\lambda\boldsymbol{D}_i+\alpha\boldsymbol{X}_i^{\mathrm{T}}(\boldsymbol{L}+\boldsymbol{L}^{\mathrm{T}})\boldsymbol{X}_i\\&+\beta\sum_{l\in S_i}(\boldsymbol{U}_l+\boldsymbol{U}_l^{\mathrm{T}}))\boldsymbol{W}_i-2\boldsymbol{X}_i^{\mathrm{T}}\boldsymbol{Y}_i=0\end{aligned} \tag{9}$$

最后,求得$\boldsymbol{W}_i$:

$$\begin{aligned}\boldsymbol{W}_i=&(\boldsymbol{X}_i^{\mathrm{T}}\boldsymbol{X}_i+\lambda\boldsymbol{D}_i+\alpha\boldsymbol{X}_i^{\mathrm{T}}(\boldsymbol{L}+\boldsymbol{L}^{\mathrm{T}})\boldsymbol{X}_i\\&+\beta\sum_{l\in S_i}(\boldsymbol{U}_l+\boldsymbol{U}_l^{\mathrm{T}}))^{-1}(\boldsymbol{X}_i^{\mathrm{T}}\boldsymbol{Y}_i)\end{aligned} \tag{10}$$

特别地,由于根节点没有兄弟节点,只需要在计算$\boldsymbol{W}_0$时令$\beta=0$,即

$$\boldsymbol{W}_0=(\boldsymbol{X}_0^{\mathrm{T}}\boldsymbol{X}_0+\lambda\boldsymbol{D}_0+\alpha\boldsymbol{X}_0^{\mathrm{T}}(\boldsymbol{L}+\boldsymbol{L}^{\mathrm{T}})\boldsymbol{X}_0)^{-1}(\boldsymbol{X}_0^{\mathrm{T}}\boldsymbol{Y}_0) \tag{11}$$

根据式(5)、式(10)和式(11)可提出算法1. 通过算法1,可以得到权重矩阵$\boldsymbol{W}=[\boldsymbol{W}_0,\boldsymbol{W}_1,\cdots,\boldsymbol{W}_N]$,分别对每个权重矩阵进行降序排序,最终获得权重值较大的特征来完成对各个内部节点的特征选择.

算法1中每次迭代的复杂度主要与更新$\boldsymbol{W}$的计算有关. 在内部节点$i$对特征权重$\boldsymbol{W}_i$迭代一次所需的时间复杂度$O(m^3+m^2d+m^2n_i+mn_id)$,其中$m$表示特征数,$d$表示内部最大类别数,$n_i$是内部节点$i$的样本数. 令$I$表示迭代总次数,由于$\boldsymbol{X}_i^{\mathrm{T}}\boldsymbol{X}_i$和$\boldsymbol{X}_i^{\mathrm{T}}\boldsymbol{Y}_i$只需要计算一次,因此,用$n$表示样本总数,则算法1的时间复杂度为$O(I(m^3+m^2d)+m^2n+mnd)$.

**算法 1　由粗到细的分层特征选择**

**输入:** 数据矩阵 $\boldsymbol{X}_0,\boldsymbol{X}_1,\cdots,\boldsymbol{X}_N$,
标记矩阵 $\boldsymbol{Y}_0,\boldsymbol{Y}_1,\cdots,\boldsymbol{Y}_N$,
正则化参数 $\lambda,\alpha,\beta$.

**输出:** 权重矩阵 $\boldsymbol{W}_0,\boldsymbol{W}_1,\cdots,\boldsymbol{W}_N$

1:随机初始化矩阵 $\boldsymbol{W}_0,\boldsymbol{W}_1,\cdots,\boldsymbol{W}_N$;
2:REPEAT
3: 更新 $\boldsymbol{W}_0$ 通过式(11);
4: FOR $i=1:N$ do
5: 通过式(10)更新 $\boldsymbol{W}_i$;
6: 根据 $\tilde{\boldsymbol{Y}}_i=\boldsymbol{X}_i\boldsymbol{W}_i$ 计算 $\tilde{\boldsymbol{Y}}_i$;
7: 根据式(5)调整 $\boldsymbol{Y}_i$;
8: END FOR
9:更新 $\boldsymbol{W}_0,\boldsymbol{W}_1,\cdots,\boldsymbol{W}_N$;
10:UNTIL 满足收敛.
11:RETURN $\boldsymbol{W}_0,\boldsymbol{W}_1,\cdots,\boldsymbol{W}_N$.

### 2.4　收敛性分析

通过式(8),已经将式(7)转化为一个等价的凸问题[18],本小节将证明算法 1 能够收敛到式(7)的全局最优解,在证明这点之前,我们需要以下引理:

**引理 1**　[20]对于任意两个正数 $a$ 和 $b$,可以得到不等式如式(12)所示:

$$a-\frac{a^2}{2b}\le b-\frac{b^2}{2b} \tag{12}$$

**证明**　已知 $(a-b)^2\ge 0$,可得 $a^2+b^2\ge 2ab$,移项后可以推出 $2a-\frac{a^2}{b}\le b$,因此 $2a-\frac{a^2}{b}\le b-\frac{b^2}{2b}$. 证毕.

接着将算法 1 的收敛性总结为定理 1,并给出证明.

**定理 1**　算法 1 在每次迭代中单调递减式(7)目标值,并收敛到问题的全局最优.

**证明**　在第 $t$ 次迭代时,任取内部节点 $i$ 的权重更新根据式(8)可以表示为

$$\begin{aligned}\boldsymbol{W}_i^{(t+1)}=&\arg\min_{\boldsymbol{W}_i}\left\|\boldsymbol{X}_i\boldsymbol{W}_i^{(t)}-\boldsymbol{Y}_i\right\|_{\mathrm{F}}^2+\lambda\mathrm{tr}(\boldsymbol{W}_i^{(t)\mathrm{T}}\boldsymbol{D}\boldsymbol{W}_i^{(t)})\\&+\alpha\mathrm{tr}(\boldsymbol{W}_i^{(t)\mathrm{T}}\boldsymbol{X}_i^{\mathrm{T}}\boldsymbol{L}\boldsymbol{X}_i\boldsymbol{W}_i^{(t)})+\beta\sum_{l\in S_i}\mathrm{HSIC}(\boldsymbol{W}_i^{(t)},\boldsymbol{W}_l^{(t)})\end{aligned} \tag{13}$$

据此可推出结果如下:

$$\begin{aligned}&\left\|\boldsymbol{X}_i\boldsymbol{W}_i^{(t+1)}-\boldsymbol{Y}_i\right\|_{\mathrm{F}}^2+\lambda\mathrm{tr}(\boldsymbol{W}_i^{(t+1)\mathrm{T}}\boldsymbol{D}\boldsymbol{W}_i^{(t+1)})\\&+\alpha\mathrm{tr}(\boldsymbol{W}_i^{(t+1)\mathrm{T}}\boldsymbol{X}_i^{\mathrm{T}}\boldsymbol{L}\boldsymbol{X}_i\boldsymbol{W}_i^{(t+1)})+\beta\sum_{l\in S_i}\mathrm{HSIC}(\boldsymbol{W}_i^{(t+1)},\boldsymbol{W}_l^{(t+1)})\\\le&\left\|\boldsymbol{X}_i\boldsymbol{W}_i^{(t)}-\boldsymbol{Y}_i\right\|_{\mathrm{F}}^2+\lambda\mathrm{tr}(\boldsymbol{W}_i^{(t)\mathrm{T}}\boldsymbol{D}\boldsymbol{W}_i^{(t)})\\&+\alpha\mathrm{tr}(\boldsymbol{W}_i^{(t)\mathrm{T}}\boldsymbol{X}_i^{\mathrm{T}}\boldsymbol{L}\boldsymbol{X}_i\boldsymbol{W}_i^{(t)})+\beta\sum_{l\in S_i}\mathrm{HSIC}(\boldsymbol{W}_i^{(t)},\boldsymbol{W}_l^{(t)})\end{aligned} \tag{14}$$

由 $\left\|\boldsymbol{W}_i^{(t+1)}\right\|_{2,1}=\sum_{j=1}^{m}\left\|(\boldsymbol{w}_{ij})^{(t+1)}\right\|_2$,可重写式(14)为

$$\begin{aligned}&\left\|\boldsymbol{X}_i\boldsymbol{W}_i^{(t+1)}-\boldsymbol{Y}_i\right\|_{\mathrm{F}}^2+\lambda\sum_{j=1}^{m}\frac{\left\|(\boldsymbol{w}_{ij})^{(t+1)}\right\|_2}{2\left\|(\boldsymbol{w}_{ij})^{(t)}\right\|_2}\\&+\alpha\mathrm{tr}(\boldsymbol{W}_i^{(t+1)\mathrm{T}}\boldsymbol{X}_i^{\mathrm{T}}\boldsymbol{L}\boldsymbol{X}_i\boldsymbol{W}_i^{(t+1)}))+\beta\sum_{l\in S_i}\mathrm{HSIC}(\boldsymbol{W}_i^{(t+1)},\boldsymbol{W}_l^{(t+1)})\\\le&\left\|\boldsymbol{X}_i\boldsymbol{W}_i^{(t)}-\boldsymbol{Y}_i\right\|_{\mathrm{F}}^2+\lambda\sum_{j=1}^{m}\frac{\left\|(\boldsymbol{w}_{ij})^{(t)}\right\|_2}{2\left\|(\boldsymbol{w}_{ij})^{(t)}\right\|_2}\\&+\alpha\mathrm{tr}(\boldsymbol{W}_i^{(t)\mathrm{T}}\boldsymbol{X}_i^{\mathrm{T}}\boldsymbol{L}\boldsymbol{X}_i\boldsymbol{W}_i^{(t)})+\beta\sum_{l\in S_i}\mathrm{HSIC}(\boldsymbol{W}_i^{(t)},\boldsymbol{W}_l^{(t)})\end{aligned} \tag{15}$$

根据引理 1,令 $a=\left\|(\boldsymbol{w}_{ij})^{(t+1)}\right\|_2$, $b=\left\|(\boldsymbol{w}_{ij})^{(t)}\right\|_2$ 可得

$$\left\|(\boldsymbol{w}_{ij})^{(t+1)}\right\|_2-\frac{\left\|(\boldsymbol{w}_{ij})^{(t+1)}\right\|_2^2}{2\left\|(\boldsymbol{w}_{ij})^{(t)}\right\|_2}\le\left\|(\boldsymbol{w}_{ij})^{(t)}\right\|_2-\frac{\left\|(\boldsymbol{w}_{ij})^{(t)}\right\|_2^2}{2\left\|(\boldsymbol{w}_{ij})^{(t)}\right\|_2} \tag{16}$$

结合式(15)和式(16)可推出结果:

$$\begin{aligned}&\left\|\boldsymbol{X}_i\boldsymbol{W}_i^{(t+1)}-\boldsymbol{Y}_i\right\|_{\mathrm{F}}^2+\lambda\left\|\boldsymbol{W}_i\right\|_{2,1}^{(t+1)}\\&+\alpha\mathrm{tr}(\boldsymbol{W}_i^{(t+1)\mathrm{T}}\boldsymbol{X}_i^{\mathrm{T}}\boldsymbol{L}\boldsymbol{X}_i\boldsymbol{W}_i^{(t+1)})+\beta\sum_{l\in S_i}\mathrm{HSIC}(\boldsymbol{W}_i^{(t+1)},\boldsymbol{W}_l^{(t+1)})\\\le&\left\|\boldsymbol{X}_i\boldsymbol{W}_i^{(t)}-\boldsymbol{Y}_i\right\|_{\mathrm{F}}^2+\lambda\left\|\boldsymbol{W}_i\right\|_{2,1}^{(t)}\\&+\alpha\mathrm{tr}(\boldsymbol{W}_i^{(t)\mathrm{T}}\boldsymbol{X}_i^{\mathrm{T}}\boldsymbol{L}\boldsymbol{X}_i\boldsymbol{W}_i^{(t)})+\beta\sum_{l\in S_i}\mathrm{HSIC}(\boldsymbol{W}_i^{(t)},\boldsymbol{W}_l^{(t)})\end{aligned} \tag{17}$$

最终推广式(17)如式(18)所示:

$$\begin{aligned}&\sum_{i=0}^{N}(\left\|\boldsymbol{X}_i\boldsymbol{W}_i^{(t+1)}-\boldsymbol{Y}_i\right\|_{\mathrm{F}}^2+\lambda\left\|\boldsymbol{W}_i\right\|_{2,1}^{(t+1)}\\&+\alpha\mathrm{tr}(\boldsymbol{W}_i^{(t+1)\mathrm{T}}\boldsymbol{X}_i^{\mathrm{T}}\boldsymbol{L}\boldsymbol{X}_i\boldsymbol{W}_i^{(t+1)}))+\beta\sum_{i=1}^{N}\sum_{l\in S_i}\mathrm{HSIC}(\boldsymbol{W}_i^{(t+1)},\boldsymbol{W}_l^{(t+1)})\\\le&\sum_{i=0}^{N}(\left\|\boldsymbol{X}_i\boldsymbol{W}_i^{(t)}-\boldsymbol{Y}_i\right\|_{\mathrm{F}}^2+\lambda\left\|\boldsymbol{W}_i\right\|_{2,1}^{(t)}+\alpha\mathrm{tr}(\boldsymbol{W}_i^{(t)\mathrm{T}}\boldsymbol{X}_i^{\mathrm{T}}\boldsymbol{L}\boldsymbol{X}_i\boldsymbol{W}_i^{(t)}))\\&+\beta\sum_{i=1}^{N}\sum_{l\in S_i}\mathrm{HSIC}(\boldsymbol{W}_i^{(t)},\boldsymbol{W}_l^{(t)})\end{aligned} \tag{18}$$

根据式(18)可知算法 1 单调递减式(7)目标值,并收敛到问题的全局最优. 证毕.

## 3　实验分析

在本节中,首先依次介绍实验中使用的数据集、对比算法、评价指标和实验设置. 接着,与 5 种层次特征选择算法进行比较分析. 然后,HFSCF 算法只利用粗粒度信息进行特征选择,验证其在数据集标记粒度不够细化情况下的有效性. 此外,通过消融实验研究模型不同部分的有效性. 最后,分析算法的收敛性、参数敏感性和效率.

### 3.1　数据集

本次实验选取 7 个基准层次数据集,包括 2 个蛋白质数据集和 5 个图像数据集. 数据集详细信息参见表 1.

表1 数据集详细信息

| 序号 | 数据集 | 训练集 | 测试集 | 特征数 | 节点数 | 叶子节点数 | 层数 |
|---|---|---|---|---|---|---|---|
| 1 | F194 | 7105 | 1420 | 473 | 202 | 194 | 3 |
| 2 | DD | 3020 | 605 | 473 | 32 | 27 | 3 |
| 3 | ILSVRC65 | 12346 | 11845 | 4096 | 65 | 57 | 4 |
| 4 | Sun | 45109 | 22556 | 4096 | 343 | 324 | 4 |
| 5 | VOC | 7178 | 5105 | 1000 | 30 | 20 | 5 |
| 6 | Cifar100 | 50000 | 10000 | 4096 | 121 | 100 | 3 |
| 7 | CLEF | 8368 | 939 | 80 | 88 | 63 | 4 |

### 3.2 对比算法

将HFSCF与5种先进的层次特征选择方法进行比较,以评估HFSCF用于层次特征选择的有效性,所比较的算法如下:

(1) HierFisher算法[21]在层次结构的每个内部节点上选择使类内距离尽可能小,类间距离尽可能大的特征.

(2) HierFSNM算法[20]在损失和正则化项上通过联合$\ell_{2,1}$范数最小化为每个子分类任务选择合适的特征子集.

(3) HiermRMR算法[10]根据最小冗余性和最大相关性准则为分层分类的每个子任务选择特征.

(4) Hier-FS算法[12]不考虑类之间的亲子关系和兄弟关系,只用稀疏性正则化来获得特征子集.

(5) HiRRfam-FS算法[12]考虑层次粒度之间的关系,利用类层次结构中的父子和兄弟关系来优化特征选择过程.

### 3.3 评价指标

由于分层分类不同于一般平面分类,传统的平面分类评价指标并不适合用来评估分层分类算法的性能.实验中采用分层评价指标Tree Induced Error (TIE)和Hierarchical F1-measure来更好地评价算法之间的优劣.

TIE指标通过计算预测标记$\breve{y}$到真实标记$y$在层次结构中节点间的总边数来反映分类错误的程度:

$$\mathrm{TIE}(y,\breve{y})=\left|E_H(y,\breve{y})\right| \tag{19}$$

其中,$E_{\mathrm{H}}(y,\breve{y})$表示从$y$到$\breve{y}$节点之间边的集合.

Hierarchical F1-measure是分层准确率和召回率的综合反映:

$$F_{\mathrm{H}}=\frac{2\times P_{\mathrm{H}}\times R_{\mathrm{H}}}{P_{\mathrm{H}}+R_{\mathrm{H}}} \tag{20}$$

其中,用增广集合$Y_{\mathrm{aug}}$表示真实标记$y$与其祖先节点的集合,$\breve{Y}_{\mathrm{aug}}$表示预测标记$\breve{y}$与其祖先节点的集合,则$P_{\mathrm{H}}$、$R_{\mathrm{H}}$表示为

$$P_{\mathrm{H}}=\frac{|Y_{\mathrm{aug}}\bigcap\breve{Y}_{\mathrm{aug}}|}{|Y_{\mathrm{aug}}|},\ R_{\mathrm{H}}=\frac{|Y_{\mathrm{aug}}\bigcap\breve{Y}_{\mathrm{aug}}|}{|\breve{Y}_{\mathrm{aug}}|} \tag{21}$$

TIE的值越大,则分类错误越严重.而Hierarchical F1-measure的值越大,则代表算法的效果越好.

### 3.4 实验设置

实验中所提算法的参数$\lambda$和$\beta$采用网格搜索法在$\{10^{-2},10^{-1},10^{0},10^{1},10^{2}\}$范围内调整,$\alpha$在$\{5\times10^{-5},5\times10^{-4},5\times10^{-3},5\times10^{-2},5\times10^{-1}\}$范围内调整.参照文献[12],将Hier-FS的参数设置为10,HiRRfam-FS算法的参数$\lambda$、$\alpha$、$\beta$分别设置为10、1、1.同时,与Hier-FS、HiRRfam-FS算法保持一致,为蛋白质数据集和图像数据集分别选择前10%和20%的特征.此外,为了公平起见,统一使用自顶向下的线性支持向量机作为基分类器,并进行十折交叉验证.

在完整数据集上,对于所提算法,为所有数据集设置一个统一的参数:$\lambda=10,\alpha=0.005,\beta=1$.

在标记粒度不够细化的数据集上,为所提算法在ILSVRC65、Cifar100数据集设置参数:$\lambda=100,\alpha=0.005,\beta=1$,为其他数据集设置参数:$\lambda=10,\alpha=0.005,\beta=1$.

本文实验系统环境为:一台配有8 GB内存、2.60 GHz默认频率的四核八线程Intel Core i7-6700HQ CPU和Windows 10系统的个人主机.编程环境为Matlab 2016a软件.

### 3.5 实验结果与分析

#### 3.5.1 完整数据集上的性能比较

将所提算法和5个对比算法在7个数据集上进行比较,表2列出不同算法的TIE值,为了便于在不同数据集之间比较,结果进行标准化处理,并用黑色粗体标出最好的结果.结果显示在F194等六个数据集上,使用HFSCF算法所选特征训练出的分类器分类错误程度低于其他对比算法,而在ILSVRC65数据集上分类错误程度略高于Hier-FS、HiRRfam-FS算法.表3给出不同算法Hierarchical F1-measure值,结果反映在F194等六个数据集上,算法HFSCF效果优于其他对比算法,而在ILSVRC65数据集上效果略差于Hier-FS、HiRRfam-FS算法.

为了进一步验证实验结果,引入统计测试.首先进行Friedman测试[22]来检测不同算法之间是否存在显著性差异.给定$k$个算法和$N$个数据集,$r_j^i$是第$j$个算法在第$i$个数据集上评价指标的序值,在表2和表3分别计算出TIE和Hierarchical F1-measure的平均序值$R_i=\frac{1}{N}\sum_{i=1}^{N}r_j^i$,在零假设情况下,所有方法等价,计算Friedman统计值:

$$F_{\mathrm{F}}=\frac{(N-1)\chi_{\mathrm{F}}^2}{N(k-1)-\chi_{\mathrm{F}}^2},\ \chi_{\mathrm{F}}^2=\frac{12N}{k(k+1)}\left(\sum_{i=1}^{N}R_j^2-\frac{k(k+1)^2}{4}\right)\quad(22)$$

通过式(22)求得 TIE 和 Hierarchical F1-measure 的 $F_{\mathrm{F}}=17.56$. 而 6 个算法和 7 个数据集的临界值 $F(6-1,(6-1)\times(7-1))=F(5,30)$ 在显著性水平 $\alpha=0.05$ 下取值 2.534. 因此拒绝零假设,算法间存在显著差异.

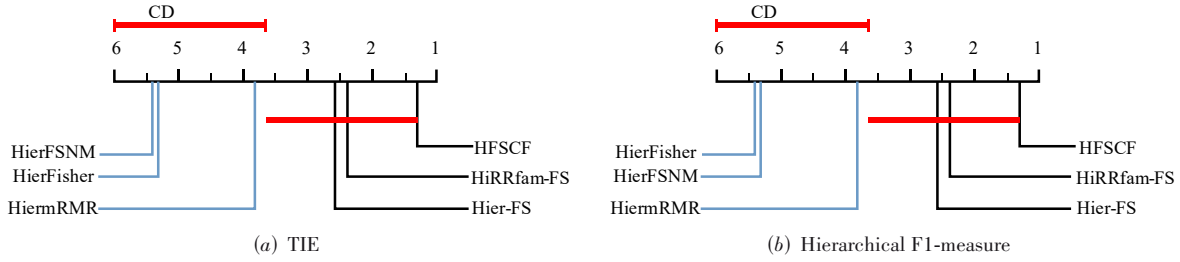
由此,进一步采用 Bonferroni-Dunn[23]来继续比较不同算法间性能的差异. 该方法比较两种算法平均序值与临界值(Critical Distance, CD)的大小,其中临界值可通过式 $\mathrm{CD}_{\alpha}=q_{\alpha}\sqrt{\frac{k(k+1)}{6N}}$ 计算得到. 当平均序值大于临界值时,两种算法性能显著不同. 参照文献[24]表 5,当 $k=6$ 时,$q_{0.1}=2.326$,因此 $\mathrm{CD}_{0.1}=2.326$. 图 1 显示了对 7 个数据集进行 $\alpha=0.1$ 的 Bonferroni-Dunn 检验结果. 结果表明,HFSCF 算法对于两种评价指标在统计上均优于 HierFSNM、HierFisher 和 HiermRMR 算法.

**表 2　不同特征选择算法在不同数据集上的标准化 TIE 结果**

| Dataset | HierFisher | HierFSNM | HiermRMR | Hier-FS | HiRRfam-FS | HFSCF |
|---|---|---|---|---|---|---|
| F194 | 0.1945(5) | 0.2123(6) | 0.1800(4) | 0.1746(3) | 0.1730(2) | **0.1701(1)** |
| DD | 0.1355(6) | 0.0886(4) | 0.0919(5) | 0.0850(3) | 0.0836(2) | **0.0830(1)** |
| ILSVRC65 | 0.0336(5) | 0.0350(6) | 0.0335(3) | **0.0328(1)** | 0.0329(2) | 0.0335(3) |
| Sun | 0.1341(5) | —(6) | 0.1322(4) | 0.1280(3) | 0.1271(2) | **0.1270(1)** |
| VOC | 0.2271(6) | 0.2144(4) | 0.2188(5) | 0.2143(3) | 0.2138(2) | **0.2129(1)** |
| Cifar100 | 0.1285(5) | —(6) | 0.1273(4) | 0.1269(2) | 0.1272(3) | **0.1268(1)** |
| CLEF | 0.2011(5) | 0.2077(6) | 0.1825(2) | 0.1826(3) | 0.1833(4) | **0.1794(1)** |
| 平均排名 | 5.29 | 5.43 | 3.86 | 2.57 | 2.43 | 1.29 |

**表 3　不同特征选择算法在不同数据集上的 Hierarchical F1-measure 结果**

| Dataset | HierFisher | HierFSNM | HiermRMR | Hier-FS | HiRRfam-FS | HFSCF |
|---|---|---|---|---|---|---|
| F194 | 0.6758(5) | 0.6462(6) | 0.7000(4) | 0.7089(3) | 0.7117(2) | **0.7164(1)** |
| DD | 0.7741(6) | 0.8524(4) | 0.8468(5) | 0.8584(3) | 0.8606(2) | **0.8618(1)** |
| ILSVRC65 | 0.9580(5) | 0.9563(6) | 0.9581(3) | **0.9591(1)** | 0.9588(2) | 0.9581(3) |
| Sun | 0.8324(5) | —(6) | 0.8348(4) | 0.8400(3) | 0.8411(2) | **0.8413(1)** |
| VOC | 0.6576(6) | 0.6739(4) | 0.6669(5) | 0.6754(3) | 0.6758(2) | **0.6768(1)** |
| Cifar100 | 0.7859(5) | —(6) | 0.7879(4) | 0.7885(2) | 0.7880(3) | **0.7887(1)** |
| CLEF | 0.7395(6) | 0.7396(5) | 0.7635(2) | 0.7631(3) | 0.7623(4) | **0.7680(1)** |
| 平均排名 | 5.43 | 5.29 | 3.86 | 2.57 | 2.43 | 1.29 |




(a) TIE　　(b) Hierarchical F1-measure

图 1　分别对于不同评价指标采用 Bonferroni-Dunn 检验比较 HFSCF 算法与其他算法的性能

为了展现所提算法与其他算法的特征选择结果，将HFSCF算法与性能最优的Hier-FS算法和HiRRfam-FS算法进行比较. 简单起见，选择不同算法在DD数据集上选择的前10%特征进行分析，其中，DD数据集的特征分为全局特征、1-gram特征、2-gram特征，以及结构剖面特征[25]. 表4表示不同算法获取的前10%特征（即47个特征）中用于区分对应节点包含不同特征类别的数量. 如表4所示，三种算法在根节点上都选择了6个全局特征，而在其他4个内部节点上只出现了1或2个全局特征，说明在根节点对下面四个子节点进行分类时需要较多全局特征的参与，而在节点1~4等四个中间节点上区分其各自子节点时，全局特征将不再是当前节点分类任务的有效特征. HFSCF算法在性能比较上优于其他算法，是因为HFSCF算法相对于其他算法在区分细粒度节点时选择了更少的全局特征和更多的其他特征，而其他特征比全局特征更利于区分细粒度节点.

### 3.5.2 标记粒度不够细化的数据集的性能比较

正如本文在3.1节中所阐述的那样，叶子标记的获取成本最为昂贵. 本节实验中所提算法将不使用样本的叶子标记，仅通过其他粗粒度标记选择特征. 由于现有的分层特征选择算法均无法在这种情况下工作，因此仍与对比算法在完整数据集上的结果进行比较. 图2比较不同算法的标准化TIE和Hierarchical F1-measure值，最优结果用不同的颜色标注. 其中标准化TIE结果显示所提算法HFSCF在不使用叶子标记的情况下，在F194等6个数据集上分类错误程度低于其他对比算法，而在Sun数据集上略低于HiRRfam-FS算法. 此外，图2中Hierarchical F1-measure结果反映HFSCF算法在不使用叶子标记的情况下在F194等4个数据集上效果优于其他对比算法. 在Sun和VOC两个数据集上只略差于HiRRfam-FS算法，仍优于其他算法.

进一步采取Friedman测试，此时TIE和Hierarchical F1-measure的$F_{\mathrm{F}}$分别为29.21和26.02，大于临界值$F(5,30)$在显著性水平$\alpha=0.05$下取值2.534.

图3显示了对7个数据集进行$\alpha=0.1$的Bonferroni-Dunn检验结果. 结果表明，HFSCF算法在不使用叶子标记的情况下对于两种评价指标在统计上都优于HierFSNM、HierFisher和HiermRMR算法.

综上，所提算法在所有数据集上维持稳定，并保持与完整数据集相近的结果，意味着HFSCF算法只利用粗粒度标记就能完成特征选择，从而使获取样本标记的成本将大大降低.

### 3.5.3 消融实验

本小节通过消除模型的不同部分，研究HFSCF算法中一致性正则项和兄弟关系正则项的有效性. 式(7)中各部分的组合表示如下：

(1) HFSCF-$\alpha$：该函数由式(7)中目标函数的前两项和第三项组成.

$$\min_{\boldsymbol{W}_0,\boldsymbol{W}_1,\cdots,\boldsymbol{W}_N}\sum_{i=0}^{N}\left(\left\|\boldsymbol{X}_i\boldsymbol{W}_i-\boldsymbol{Y}_i\right\|_F^2+\lambda\left\|\boldsymbol{W}_i\right\|_{2,1}+\alpha\mathrm{tr}(\boldsymbol{W}_i^{\mathrm{T}}\boldsymbol{X}_i^{\mathrm{T}}\boldsymbol{L}\boldsymbol{X}_i\boldsymbol{W}_i)\right) \tag{23}$$

(2) HFSCF-$\beta$：该函数由式(7)中目标函数的前两项和第四项组成.

$$\min_{\boldsymbol{W}_0,\boldsymbol{W}_1,\cdots,\boldsymbol{W}_N}\sum_{i=0}^{N}\left(\left\|\boldsymbol{X}_i\boldsymbol{W}_i-\boldsymbol{Y}_i\right\|_F^2+\lambda\left\|\boldsymbol{W}_i\right\|_{2,1}\right)+\beta\sum_{i=1}^{N}\sum_{l\in S_i}\mathrm{HSIC}(\boldsymbol{W}_i,\boldsymbol{W}_l) \tag{24}$$

简单起见，给出两个不同领域的数据集DD和Sun的实验结果. 图4比较HFSCF算法中不同部分的Hierarchical F1-measure结果，其中缺少叶子节点标记的数据相应用*进行标注，并用褐色标注出最优结果.. 如图4所示，所提算法HFSCF优于HFSCF-$\alpha$和HFSCF-$\beta$算法. 尽管HFSCF-$\beta$算法在整体上表现最差，但在Sun*数据集上优于HFSCF-$\alpha$算法. 这表明仅依靠一致性正则项或兄弟正则项都无法还原出准确的细粒度信息. 因此，结合一致性和兄弟关系正则项对挖掘细粒度信息是重要的.

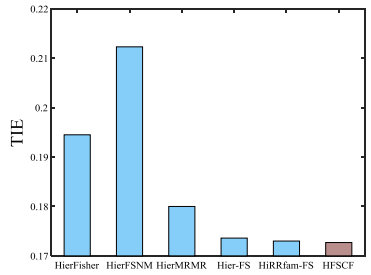
### 3.5.4 算法的收敛性分析

本小节通过实验进一步研究HFSCF算法的收敛性，7个数据集的实验结果如图5所示，其中缺少叶子节点标记的数据相应用*进行标注. 在完整的数据集上，HFSCF算法能够在4到6次迭代内快速收敛. 当数据集缺少叶子节点标记时，HFSCF算法在蛋白质数据集上迭代30到40次后收敛，而在图像数据集上仍保持快速收敛.

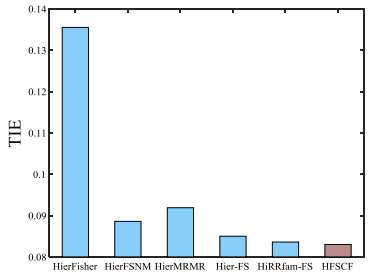
### 3.5.5 参数敏感性分析

本小节分析所提算法HFSCF的参数$\lambda$、$\alpha$和$\beta$的敏感性，其中$\lambda$控制着所选特征稀疏程度，$\alpha$和$\beta$分别控制一致性和兄弟正则化. 参数$\lambda$、$\beta$从集合$\{10^{-2},10^{-1},10^{0},10^{1},10^{2}\}$内选择，参数$\alpha$从集合$\{5\times10^{-5},5\times10^{-4},5\times10^{-3},5\times10^{-2},5\times10^{-1}\}$内选择. 通过在范围内调整其中一个参数的值，同时固定另外两个参数，来观察算法对变化参数的敏感性.

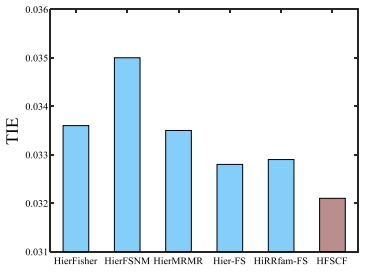
图6给出在VOC数据集上，HFSCF算法的敏感性评估结果，可以看出在范围内所提算法对参数$\alpha$和$\beta$都不敏感. 相比之下，算法对参数$\lambda$较为敏感，原因是过于稀疏的特征系数会忽略掉一些有用的特征，过于稠密的稀疏特征系数会保留一些无用特征，故应保持$\lambda$在网格范围内适中的位置.



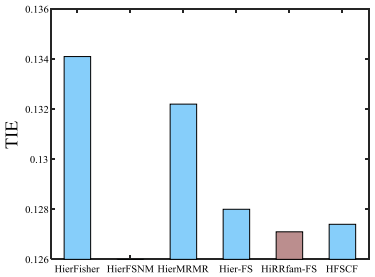

(*a*) F194(TIE)



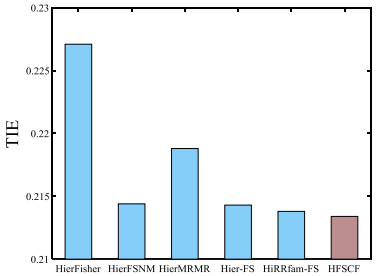

(*b*) DD(TIE)



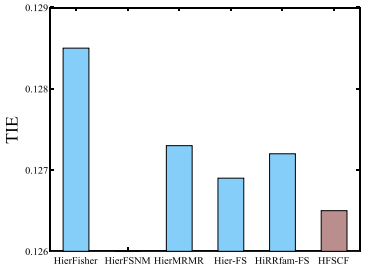

(*c*) ILSVRC65(TIE)



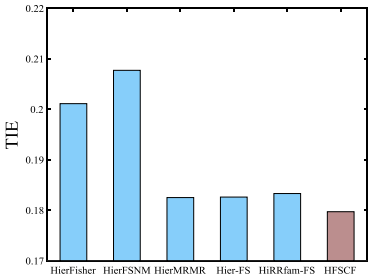

(*d*) Sun(TIE)



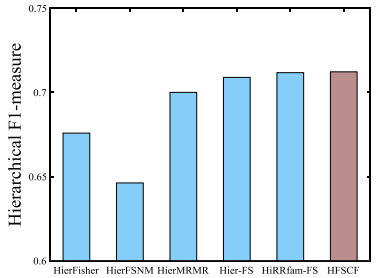

(*e*) VOC(TIE)



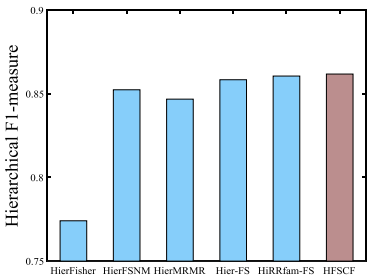

(*f*) Cifar100(TIE)



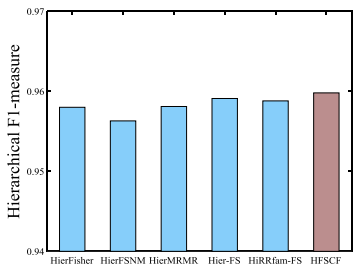

(*g*) CLEF(TIE)



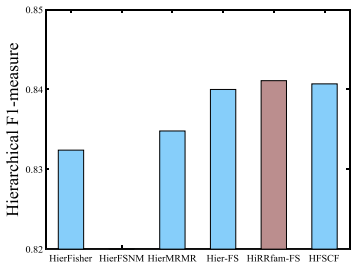

(*h*) F194(Hierarchical F1-measure)



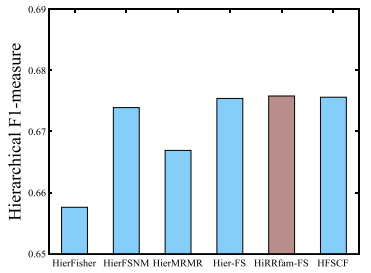

(*i*) DD (Hierarchical F1-measure)



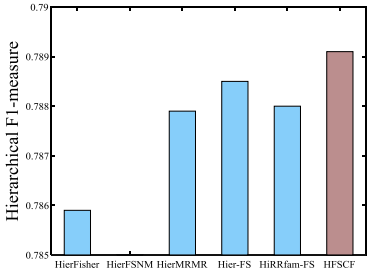

(*j*) ILSVRC65(Hierarchical F1-measure)



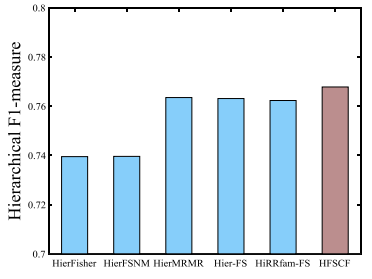

(*k*) Sun (Hierarchical F1-measure)




(*l*) VOC (Hierarchical F1-measure)

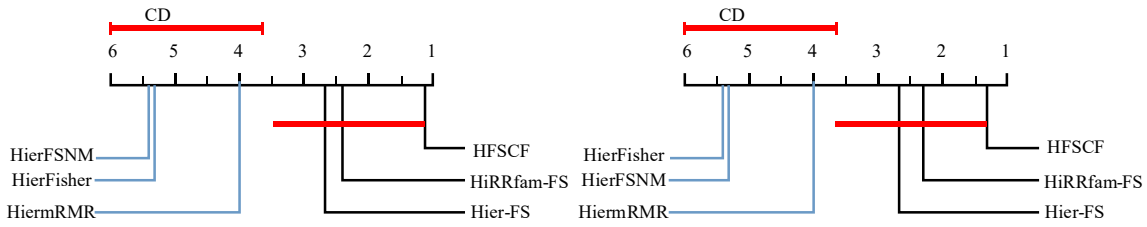



(*m*) Cifar100(Hierarchical F1-measure)




(*n*) CLEF(Hierarchical F1-measure)

图2　不使用叶子标记的HFSCF算法与其他算法在不同数据集上的性能

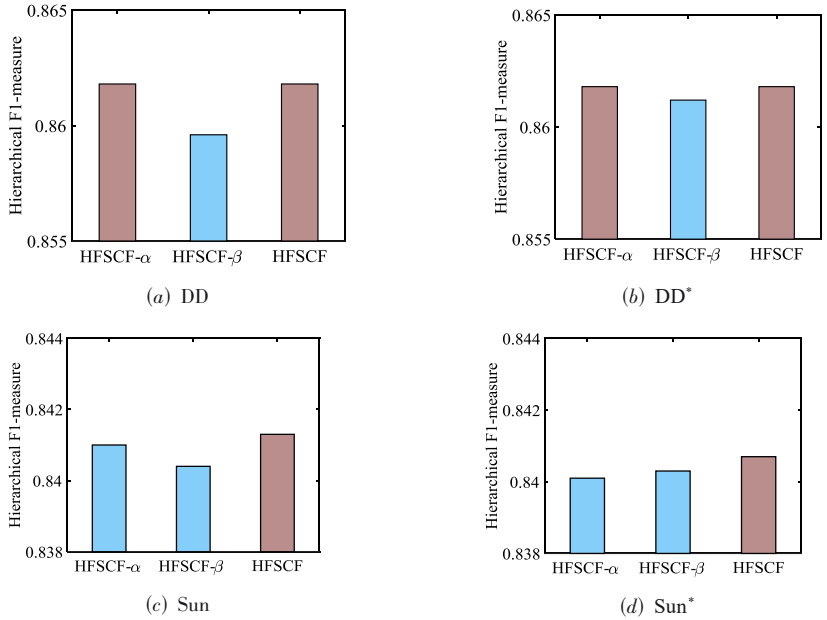



(*a*) TIE (*b*) Hierarchical F1-measure

图3 分别对于不同评价指标采用Bonferroni-Dunn检验比较不使用叶子标记的HFSCF算法与其他算法的性能

**表4 不同特征选择算法在DD数据集上选择的前10%特征中各特征类别数量**

| | Hier-FS算法 | | | HiRRfam-FS算法 | | | HFSCF算法 | | |
|---|---|---|---|---|---|---|---|---|---|
| | 全局特征 | 2-gram特征 | 结构剖面特征 | 全局特征 | 2-gram特征 | 结构剖面特征 | 全局特征 | 2-gram特征 | 结构剖面特征 |
| 根节点 | 6 | 17 | 24 | 6 | 17 | 24 | 6 | 17 | 24 |
| 节点1 | 2 | 20 | 25 | 1 | 20 | 26 | 1 | 20 | 26 |
| 节点2 | 2 | 19 | 26 | 2 | 19 | 26 | 1 | 19 | 27 |
| 节点3 | 2 | 20 | 25 | 2 | 20 | 25 | 2 | 20 | 25 |
| 节点4 | 2 | 18 | 27 | 2 | 18 | 27 | 2 | 20 | 25 |




(*a*) DD (*b*) DD* (*c*) Sun (*d*) Sun*

图4 基于DD和Sun数据集的消融结果

### 3.5.6 算法效率分析

本小节比较不同算法在层次结构中选择特征子集需要的时间. 表5给出各算法在不同数据集上的运行时间与排名情况. 从3.3节可知所提算法的时间复杂度与特征和样本数量有关,但由于算法能够在4到6次迭代内快速收敛,因此在运行时间上优于HierFSNM、HiermRMR、HiRRfam-FS算法,这种优势在大规模数据集ILSVRC65、Sun、Cifar100上更为明显. HierFisher算法为每个内部节点独立地选择特征,并未考虑节点间的联系,速度上比HFSCF算法快,但选择的特征子集分类性能较差. 相比Hier-FS算法只用稀疏性正则化来获得特征子集,HFSCF算法考虑类内一致性与兄弟节点间的差异性,在正则项的计算上用时更多,同时提高了性能.

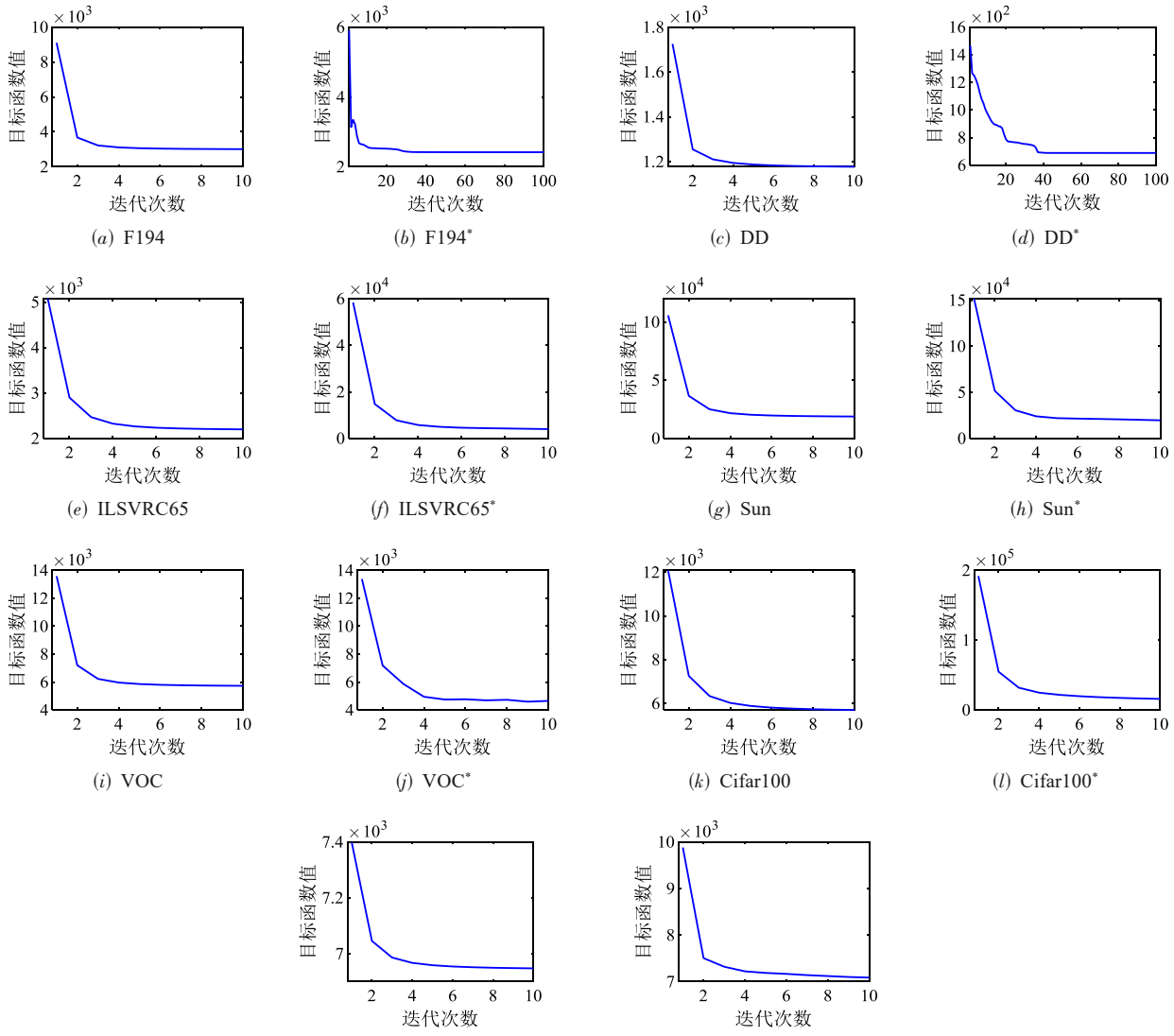



图5　目标函数值的收敛曲线

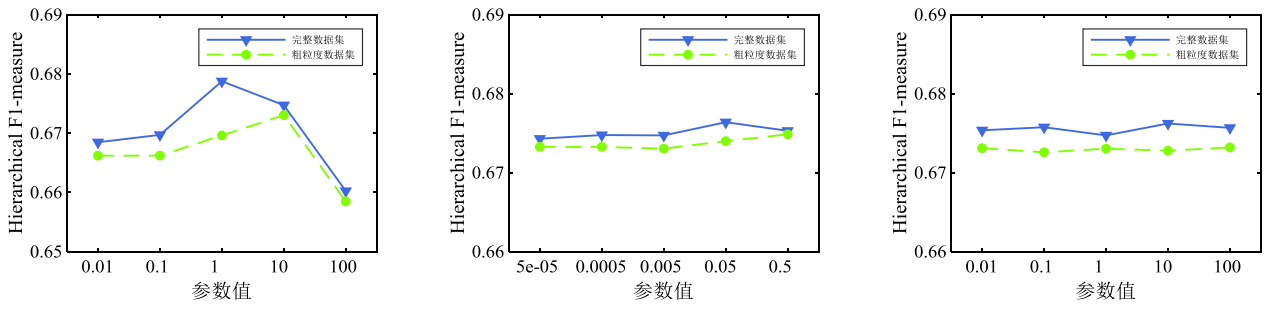



图6　基于VOC数据集的参数敏感性评估

表5 不同特征选择算法在不同数据集上的运行时间 单位:s

| Dataset | HierFisher | HierFSNM | HiermRMR | Hier-FS | HiRRfam-FS | HFSCF |
|---|---|---|---|---|---|---|
| F194 | 2.22(1) | 79.02(6) | 23.14(5) | 2.40(2) | 9.29(4) | 8.53(3) |
| DD | 0.67(1) | 10.50(6) | 8.73(5) | 1.27(2) | 3.63(4) | 2.80(3) |
| ILSVRC65 | 12.84(1) | 621.96(5) | 4815.07(6) | 344.85(3) | 546.11(4) | 218.54(2) |
| Sun | 59.46(1) | —(6) | 16033.21(5) | 821.84(2) | 2028.49(4) | 1196.58(3) |
| VOC | 0.98(1) | 83.37(5) | 123.67(6) | 12.96(2) | 22.58(4) | 16.04(3) |
| Cifar100 | 20.82(1) | —(6) | 9928.17(5) | 846.76(2) | 6829.04(4) | 2910.07(3) |
| CLEF | 0.28(1) | 115.73(6) | 1.29(4) | 0.46(2) | 0.60(3) | 4.28(5) |

## 4 总结

本文提出一种针对标记粒度不够细化情况下的由粗到细的分层特征选择模型. 通过鲁棒线性回归预测细粒度标记,并且通过有效的约束选择有区分能力的特征子集. 实验结果验证所提算法HFSCF不仅在完整的数据集上优于现有的层次树结构特征选择算法,同时能够处理数据集标记粒度不够细化的情况.

该模型仅实现对树结构数据进行特征选择,无法处理其他类型的层次结构数据. 今后,将考虑在缺少细粒度标记的情况下,为图结构数据设计特征选择方法. 此外,未来将通过改进损失函数,更充分地挖掘细粒度信息,以提升分类性能.

### 参考文献


[1] 王忠伟, 陈叶芳, 钱江波, 等. 基于LSH的高维大数据$k$近邻搜索算法[J]. 电子学报, 2016, 44(4): 906-912.
WANG Zhong-wei, CHEN Yie-fang, QIAN Jiang-bo, et al. LSH-Based algorithm for $k$ nearest neighbor search on big data[J]. Acta Electronica Sinica, 2016, 44(4): 906-912. (in Chinese)

[2] DENG J, DONG W, SOCHER R, et al. ImageNet: a large-scale hierarchical image database[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE Press, 2009: 248-255.

[3] WANG G Y, YANG J, XU J. Granular computing: from granularity optimization to multi-granularity joint problem solving[J]. Granular Computing, 2017, 2(3): 105-120.

[4] YAO J T, VASILAKOS A V, et al. Granular computing: perspectives and challenges[J]. IEEE Transactions on Cybernetics, 2013, 43(6): 1977-1989.

[5] BARGIELA A, PEDRYCZ W. Toward a theory of granular computing for human-centered information processing [J]. IEEE Transactions on Fuzzy Systems, 2008, 16(2): 320-330.

[6] 胡清华, 王煜, 周玉灿, 等. 大规模分类任务的分层学习方法综述[J]. 中国科学: 信息科学, 2018, 48(5): 487-500.
HU Q H, WANG Y, ZHOU Y C, et al. A review on hierarchical learning methods for large scale classification task [J]. Scientia Sinica Informationis, 2018, 48(5): 487-500. (in Chinese)

[7] GUO S X, ZHAO H. Hierarchical classification with multi-path selection based on granular computing[J]. Artificial Intelligence Review, 2021, 54(3): 2067-2089.

[8] SILLA C N, FREITAS A A. A survey of hierarchical classification across different application domains[J]. Data Mining and Knowledge Discovery, 2011, 22(1-2): 31-72.

[9] FREEMAN C, KULIC D, BASIR O. Joint feature selection and hierarchical classifier design[C]//Proceedings of the International Conference on Systems, Man, and Cybernetics. Piscataway: IEEE Press, 2011: 1728-1734.

[10] GRIMAUDO L, MELLIA M, BARALIS E. Hierarchical learning for fine grained internet traffic classification[C]// Proceedings of International Wireless Communications and Mobile Computing Conference. Piscataway: IEEE Press, 2012: 463-468.

[11] SONG J, ZHANG P Z, QIN S J, et al. A method of the feature selection in hierarchical text classification based on the category discrimination and position information [J]. IEEE Transactions on Engineering Management, 2015, 53(4): 555-569.

[12] ZHAO H, HU Q H, ZHU P F, et al. A recursive regularization based feature selection framework for hierarchical classification[J]. IEEE Transactions on Knowledge and Data Engineering, 2021, 33(7): 2833-2846.

[13] TUO Q J, ZHAO H, HU Q H. Hierarchical feature selection with subtree based graph regularization[J]. Knowledge Based Systems, 2018, 163(1): 996-1008.

[14] 白盛兴, 林耀进, 王晨曦, 等. 基于邻域粗糙集的大规模层次分类在线流特征选择[J]. 模式识别与人工智能, 2019, 32(9): 811-820.
BAI Shengxing, LIN Yaojin, WANG Chenxi, et al. Large-scale hierarchical classification online streaming feature selection based on neighborhood rough set[J]. Pattern Recognition and Artificial Intelligence, 2019, 32(9):

811-820. (in Chinese)
[15] LIU X X, ZHOU Y C, ZHAO H. Robust hierarchical feature selection driven by data and knowledge[J]. Information Sciences, 2021, 551: 341-357.
[16] KOSMOPOULOS A, PARTALAS I, GAUSSIER É, et al. Evaluation measures for hierarchical classification: a unified view and novel approaches[J]. Data Mining and Knowledge Discovery, 2015, 29(3): 820-865.
[17] 刘洪涛, 李航, 王进, 等. 基于标签特定特征的多目标回归稀疏集成方法[J]. 电子学报, 2020, 48(5): 906-913.
LIU Hong-tao, LI Hang, WANG Jin, et al. Multi-target regression via sparse integration and label-specific features [J]. Acta Electronica Sinica, 2016, 48(5): 906-912. (in Chinese)
[18] ARGYRIOU A, EVGENIOU T, PONTIL M. Multi-task feature learning[C]//Proceedings of the Annual Conference on Neural Information Processing Systems. Cambridge: MIT Press, 2006: 41-48.
[19] GRETTON A, BOUSQUET O, SMOLA A, et al. Measuring statistical dependence with hilbert-Schmidt norms[C]// Proceedings of the International Conference on Algorithmic Learning Theory. Berlin: Springer, 2005: 63-77.
[20] NIE F P, HUANG H, CAI X, et al. Efficient and robust feature selection via joint L2, 1-norms minimization[C]// Proceedings of the Annual Conference on Neural Information Processing Systems. Cambridge: MIT Press, 2010: 1813-1821.
[21] GU Q Q, LI Z H, HAN J W. Generalized fisher score for feature selection[C]//Proceedings of the Twenty-Seventh Conference on Uncertainty in Artificial Intelligence. Virginia: AUAI Press, 2011: 266-273.
[22] FRIEDMAN M. A comparison of alternative tests of significance for the problem of m rankings[J]. The Annals of Mathematical Statistics, 1940, 11(1): 86-92
[23] DUNN O J. Multiple comparisons among means[J]. Journal of the American Statistical Association, 1961, 56 (293): 52-64
[24] DEMSAR J. Statistical comparisons of classifiers over multiple data sets[J]. Journal of Machine Learning Research, 2006, 7(1): 1-30
[25] WEI L Y, LIAO M H, GAO X, et al. An improved protein structural prediction method by incorporating both sequence and structure information[J]. IEEE Transactions on NanoBioscience, 2015, 14(4): 339-349.

## 作者简介

**刘浩阳**　男, 1998年7月生于福建省龙岩市.现为闽南师范大学硕士研究生.主要研究方向为数据挖掘.
E-mail: liuhaoyang98@163.com

**林耀进(通讯作者)**　男, 1980年10月生于福建省漳州市.现为闽南师范大学计算机学院教授.主要研究方向为数据挖掘、机器学习.
E-mail: zzlinyaojin@163.com