哈爾濱工業大學

毕业设计（论文）中期报告

题 目：面向 DAG 标签结构数据的分类方法设计

专　　业　测控技术与仪器

学　　生　李慧莹

学　　号　1170100603

指导教师　乔家庆

日　　期　2021 年 3 月 29 日

哈尔滨工业大学教务处制

目录

1. 课题主要研究内容及进度情况

1.1 课题主要研究内容

本课题主要内容为设计 DAG 标签结构数据分类算法，算法基于已有的针对求解最小化条件风险的数学模型，主要包含三个方面，针对有向无环图分类的损失函数设计，将层级多标签分类转换为二元分类的二元分类器选择，以及为满足层级约束的算法设计。针对有向无环图结构，在层级多标签分类问题中，分类结果必须符合预先定义的层级约束关系，这种关系可以概括为：若一个样本属于一个特定类别，则它同时属于这个类别的所有祖先；若一个样本不属于一个特定类别，那么同时它一定不属于这个类别的所有子孙类别。

1.2 课题进度情况

2020 年 11 月 30 日至 2021 年 12 月 4 日 学习相关理论及背景，撰写开题报告

2020 年 12 月 5 日至 2021 年 1 月 10 日 分析支持向量机等分类器基本原理

2021 年 1 月 11 日至 2021 年 3 月 7 日 实现多标签分类方法的处理流程

2021 年 3 月 8 日至 2021 年 4 月 8 日 进行对比实验，分析评估本文所述层级多标签分类方法，撰写中期报告

2. 目前已完成的研究工作及结果

2.1 总体研究方案

针对层级多标签分类问题（Hierarchical Multi-Label Classification）设计的算法必须能够将样本数据标记为符合层次结构中的一个或多个路径。为此，算法必须对损失函数进行局部或全局优化，这两种优化方式对应于目前已有的两类 HMC 分类算法，局部分类方法和全局分类方法。执行局部学习的算法试图

发现在层次结构的特定区域中类别关系的特殊性，并结合局部预测以生成最终分类。其思想是每个分类器负责预测特定节点或特定层次结构，然后经使分类结果满足层级信息的规则或策略进行节点预测的修正。而 HMC 的全局方法通常仅由单个分类器组成，它能够将样本数据与其在整个层次结构中的对应类相关联。使用全局或局部方法有各自的优点和缺点，全局方法通常比局部方法计算量小，并且它们不会受到错误传播问题的影响，但它们无法从层次结构中捕获局部信息，可能会出现欠拟合的现象。局部方法由于依赖针对单个节点的分类器的独立分类(LCN/LCPN)或针对每一层的分类器的级联(LCL)，因此计算开销要大得多，但是它们更适合于从层次结构中提取信息。本文所述方法属于局部分类方法，将层级多标签分类问题转换成一组二分类问题，针对每个节点设计一个分类器，并将层级结构信息单独加入考虑，在保证架构灵活的同时，减小了训练工作量。

2.2 多标签基础分类

2.2.1 数据预处理

实验所用数据集来自生物学实验数据长链非编码 RNA (lncRNAs)，在 lncRNA 共表达网络中，拓扑结构相似的节点可能具有相似的功能，文献采用了 DCA 策略提取 lncRNAs 的低维拓扑信息，即特征值信息。目前没有 lncRNAs 的公共 GO 注释，但基于目标 lncRNA 可能与 lncRNA 蛋白结合网络中的直接相邻蛋白具有非常相似的功能，文献根据已知的蛋白质 GO 注释，采用邻域计数法对一些 lncRNAs 进行注释。

由于当某些 GO 标签包含的样本过少时，没有足够的信息可以用于分类器的训练，故对节点进行筛选，筛选策略为，若全部数据集中含某节点数小于阈值，则丢弃此节点，采用此方法一方面可以减少分类节点数量降低训练成本，另一方面保证每个节点有一定数量的正样本可供分类训练使用。此外，由于原始节点分布在三大生物学功能 Molecular Function(MF), Cellular Component(CC)和 Biological Process(BP)中，针对每一种生物学功能有各自的根节点，对应于三种独立的 DAG 层级结构，而对前一步骤筛选后的节点的分析表明，仅有少量不属于 BP 功能的节点存在，且这些不属于 BP 功能的节点构成的层级结构规模较小、层级信息不完整，于是采用二次筛选，留下属于 BP 功能含根节点在内的 205 个节点，构成了含根节点在内的 12 级层次结构，并对节点进行标号，提取出通往节点路径、父子节点索引、所处层级深度等基本信息。

将全部样本数据集划分为 5 个子数据集，在每个子数据集中按 6：2：1 划

分训练集，验证集以及测试集，为每个样本数据进行符合 GO 层级结构的完全标注，数据集信息如表 2-1 所示：

Dataset	属性值数量	训练样本数量	验证样本数量	测试样本数量	GO 标签数量
Dataset1	50	537	179	90	204
Dataset2	50	537	179	90	204
Dataset3	50	537	179	90	204
Dataset4	50	537	179	90	204
Dataset5	50	538	179	90	204

表 2-1 实验数据集描述

2.2.2 基础分类

实验中采用的基础分类器为 SVM 分类器，Support vector machine(SVM)具有坚实的统计学理论基础，它可以很好地应用于高维数据，避免维灾难问题，。SVM 分类器既可以用于线性可分的数据，又可以通过映射扩展到非线性可分的数据上。SVM 学习问题可以表示为凸优化问题，因此可以利用已知的有效算法发现目标函数的全局最小值。而其他的分类方法（如基于规则的分类器和人工神经网络）都采用一种基于贪心算法的策略来搜索假设空间，这种方法一般只能获得局部最优解。SVM 分类器模型如图 2-1 所示。

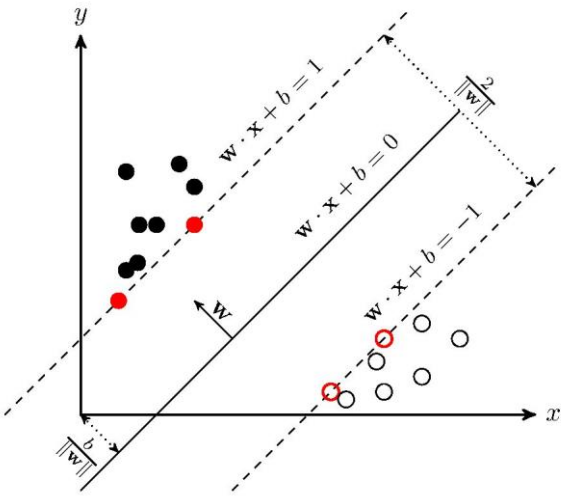

图 2-1 SVM 分类器模型

线性 SVM 寻找具有最大边缘的超平面，如果与直线 1 平行的两直线间距离，是能区分开正类和负类的一组平行直线间的最大距离，那么直线 1 为最大边缘超平面。具有较大边缘的决策边界比具有较小边缘的边界具有更好的泛化误差，如果边缘较小，决策边界的轻微扰动都可能对分类产生较大影响。支持向量机根据有限的样本信息在模型的复杂性（即对特定训练样本的学习精度，Accuracy）和学习能力（即无错误地识别任意样本的能力）之间寻求最佳折衷，以期获得最好的推广能力（或称泛化能力）。一个线性分类器的决策边界可以写成如下形式：

$$\mathrm{w}\cdot\mathrm{x}+\mathrm{b}=0 \qquad (2-1)$$

如果标记所有的正类的类标号为+1，所有负类的类标号为-1，则可以用以下方式预测任意测试样本的类标号 y：

$$y=\begin{cases}1, w\cdot z+b>0\\ -1, w\cdot z+b<0\end{cases} \qquad (2-2)$$

线性 SVM 的学习任务可以形式化地描述为以下被约束的优化问题

$$\min_{\mathrm{w}}\frac{\|\mathrm{w}\|^2}{2} \qquad (2-3)$$

受限于$y_i(w\cdot x_i+b)\geq 1$, i=1, 2, ⋯, N

对于非线性决策边界数据集，将数据从原来的坐标空间 x 变换到一个新的坐标空间ϕ(x)中，从而可以在变换后的坐标空间中使用一个线性的决策边界来划分样本。

非线性 SVM 的学习任务可以形式化地描述为以下被约束的优化问题

$$\min_{\mathrm{w}}\frac{\|\mathrm{w}\|^2}{2} \qquad (2-4)$$

受限于$y_i(w\cdot\phi(x_i)+b)\geq 1$, i=1, 2, ⋯, N

同线性 SVM 相比，主要区别在，学习任务是在变换后的属性$\phi(x)$，而不是在原属性 x 上执行的。经拉格朗日函数变换后，为避免空间中向量对之间的点积计算导致的维灾难问题，引入核函数

$$\mathrm{K}(x_i,x_j)=\left(\Phi(x_i)\cdot\Phi(x_j)\right)=\langle\Phi(x_i)\cdot\Phi(x_j)\rangle \qquad (2-5)$$

变换后的空间中的点积就可以用原空间中的相似度函数函数表示，这样，非线性 SVM 问题在使用核函数后就转化为了线性分类问题，便于 SVM 进行分类操作。

在实验中，支持向量机 SVM 使用 LIBSVM 软件包实现。LIBSVM 是台湾大学林智仁教授开发的一个实现 SVM 算法的软件包，该软件运算速度快，提供源代码，便于研究者进行扩展和修改，并且可以集成到多个平台下。作者为 SVM 提供了很多默认的参数，这使得 SVM 的使用和调整参数的过程变得简单，因此是目前国内实现 SVM 比较流行的方案。实验中的核函数选择了

RBF(Radial Basis Function)径向基函数，分类遵循如下步骤，转换数据至 SVM 格式，执行数据归一化，使用径向基核函数，用交叉验证找到最优参数 C 和 γ，最后使用最优的参数进行整个数据集的训练，并进行测试。

2.3 层级约束实现

2.3.1 损失函数模型构建

将真实标签为 y 的样本 x 分为标签$\hat{y}$的损失，由损失函数$\mathrm{L}(y,\hat{y})$给出，其中$\hat{y}=f(x)$。此时，将样本采取决策，分为标签$\hat{y}$的条件风险 $\mathrm{R}(\hat{y},x)$为：$\mathrm{R}(\hat{y},x)=\sum_{y\in\{0,1\}^c}L(y,\hat{y})P(y|x)$。根据基于最小风险的贝叶斯决策原理，对于样本 x，使条件风险最小的标签$\hat{y}^*$即为样本 x 的预测标签，因此对样本 x，标签的预测问题转换成为优化问题：$\hat{y}^*=\arg\min_{\hat{y}\in\{0,1\}^c}\sum_{y\in\{0,1\}^c}L(y,\hat{y})P(y|x)$。

在符合层级约束条件的情况下，在 DAG 层级结构中可能会发生的预测错误类型分为如下四种：

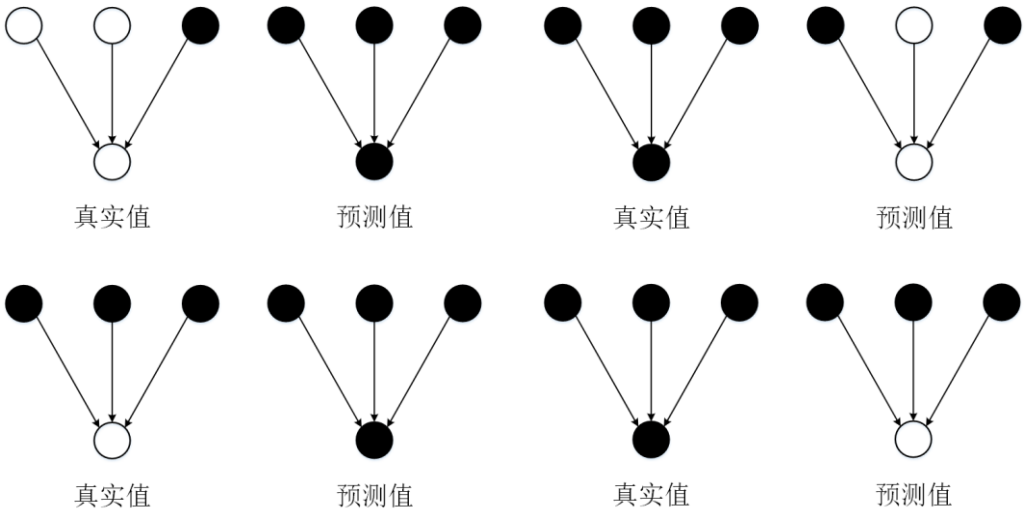

图 2-2 层级多标签分类中四种分类错误示意图

由此，层级多标签分类问题的损失函数$\mathrm{L}(y,\hat{y})$可由四部分组成，DAGH 损失函数的具体公式为：

$$L_{DAGH}(\hat{y},\mathbf{y})=\ell_1+\ell_2+\ell_3+\ell_4 \qquad (2-6)$$

$$\begin{aligned}\ell_1 &= w_1 \sum_{i=1}^{N-1} C_i y_i \tilde{y}_i \prod_{j\in par(i)} y_j \hat{y}_j \\ \ell_2 &= w_2 \sum_{i=1}^{N-1} C_i y_i \tilde{y}_i \sum_{j\in par(i)} y_j \tilde{y}_j \\ \ell_3 &= w_3 \sum_{i=1}^{N-1} C_i \tilde{y}_i \hat{y}_i \prod_{j\in par(i)} y_j \hat{y}_j \\ \ell_4 &= w_4 \sum_{i=1}^{N-1} C_i \tilde{y}_i \hat{y}_i \sum_{j\in par(i)} \tilde{y}_j \hat{y}_j\end{aligned} \tag{2-7}$$

其中，w_1, w_2, w_3, w_4均为权值常数，表示不同错误在损失函数中所占的权重。C_i是节点 i 的误分类代价：

$$C_i = \begin{cases} 1, & i = 0 \\ \sum_{j\in par(i)} \dfrac{C_j}{|child(i)|}, & i > 0 \end{cases} \tag{2-8}$$

对于一个样本 x ，其采用 DAGH 损失函数的条件风险的具体形式为：

$$\begin{aligned} R(\hat{\mathbf{y}} \mid \mathbf{x}) &= \sum_{\mathbf{y}\in\{0,1\}^N} L_{DAGH}(\mathbf{y}, \hat{\mathbf{y}}) P(\mathbf{y} \mid \mathbf{x}) \\ &= w_1 \sum_{i=1}^{N-1} C_i \tilde{y}_i p_i \prod_{j\in par(i)} \hat{y}_j \\ &+ w_2 \sum_{i=1}^{N-1} C_i \tilde{y}_i p_i \sum_{j\in par(i)} \tilde{y}_j \\ &+ w_3 \sum_{i=1}^{N-1} C_i \hat{y}_i \left(\prod_{j\in pa(i)} p_j - p_i \right) \prod_{j\in par(i)} \hat{y}_j \\ &+ w_4 \sum_{i=1}^{N-1} C_i \hat{y}_i \sum_{j\in par(i)} \hat{y}_j (1 - p_j) \end{aligned} \tag{2-9}$$

于是，采用 DAGH 损失函数，标签预测问题可以转化为如下优化问题：

$$\begin{aligned} \hat{\mathbf{y}}^* &= \arg\min_{\hat{y}\in\Psi} R(\hat{\mathbf{y}} \mid \mathbf{x}) \\ &= \arg\min_{\hat{y}\in\Psi} \sum_{\mathbf{y}\in\{0,1\}^N} L_{DAGH}(\mathbf{y}, \hat{\mathbf{y}}) P(\mathbf{y} \mid \mathbf{x}) \end{aligned} \tag{2-10}$$

2.3.2 损失函数优化问题

求取风险函数最小化的问题等价于如下优化问题：

$$\hat{\mathbf{y}}^* = \arg\max_{\hat{y}\in\Psi} LE_{\delta}(\hat{\mathbf{y}}, \mathbf{x}) \tag{2-11}$$

其中$LE_\delta(\hat{\mathbf{y}},\mathbf{x})$函数定义为：

$$LE_\delta(\hat{\mathbf{y}},\mathbf{x}) = w_2\sum_{i=1}^{N-1} C_i p_i \sum_{j\in \mathrm{par}(i)} \hat{y}_j - w_1 \sum_{i=1}^{N-1} C_i p_i \prod_{j\in \mathrm{par}(i)} \hat{y}_j$$
$$+\sum_{i=1}^{N-1} \hat{y}_i \left[w_1 C_i p_i - w_3 C_i \left(\prod_{j\in \mathrm{par}(i)} p_j - p_i \right) - w_4 C_i \sum_{j\in \mathrm{par}(i)} (1-p_j) \right] \quad (2-12)$$

定义节点函数σ(·)，对于某节点 i 有

$$\sigma(i) = \begin{cases} \sigma_1(i), i = 0 \\ \sigma_1(i) + \sigma_2(i), i > 0 \end{cases} \quad (2-13)$$

其中

$$\sigma_1(i) = \sum_{j\in child(i)} w_2 C_j p_j - \prod_{j\in child(i)} w_1 C_j p_j$$
$$\sigma_2(i) = w_1 C_i p_i - w_3 C_i \left(\prod_{j\in par(i)} p_j - p_i \right) - w_4 C_i \sum_{j\in par(i)} (1-p_j) \quad (2-14)$$

在函数$\sigma_1(i)$ 中，当节点 i 的子节点集合为空集时，该函数值为 0；即当 child(i) =∅ 时，有 $\sigma_1(i)$ = 0。函数 $\sigma_2(i)$的定义不包括根结点。

在引入了节点函数 σ(i)的概念后，$LE_\delta(\hat{\mathbf{y}},\mathbf{x})$可以用 σ$(i)$ 进行表示

$$LE_\delta(\hat{\mathbf{y}},\mathbf{x}) = \sum_i \hat{y}_i \sigma(i) \quad (2-15)$$

对于一个有向无环图结构的层级多标签分类问题，当采用 DAGH 损失函数时，根据贝叶斯决策理论，对样本的分类问题可转化为如下优化问题：

$$\hat{\mathbf{y}}^* = \arg\max_{\hat{\mathbf{y}}\in\Psi} \sum_i \hat{y}_i \sigma(i) \quad (2-16)$$

此公式即为有向无环图结构的层级多标签分类问题的数学模型。当对公式描述的这一优化问题进行求解时，只需求得样本在各节点的后验概率即可。

2.3.3 DAGlabel 介绍

以上损失函数的构建需满足层级约束要求这一先决条件，即不会出现祖先节点预测值为 0，而子孙节点预测值为 1 的情况。而在多个节点独立训练的过程中，此现象时有发生。为解决违反层级约束关系的问题，提出层级约束算法 DAGlabel 算法，来处理σ(i)矩阵。算法思路如下：

将每个标签节点的σ(i)值从大到小排序，然后选取σ(i)值最大的节点，记为 t，查找t的全部父节点，记录其中σ(i)值小于 0 的个数，记为 N。如果 N 等于 0，则将t与其σ(i)值最小的父节点合成为一个超级节点，超级节点的σ(i)值为被合成节点的平均值；若 N 大于等于 1，则将t与其σ(i)值小于 0 的所有父节点合

成为一个超级节点，超级节点的$\sigma(i)$值为被合成节点的平均值。

其中有一种情形需要单独考虑：如图 2-3 所示的特殊情况，当子节点为 3，且将与父节点 1 合成超级节点时，此时如果直接合成，新节点就会和节点 2 构成环路。所以，在这种情形下，子节点 3 与父节点 1 父节点 2 一起合并为超级节点。

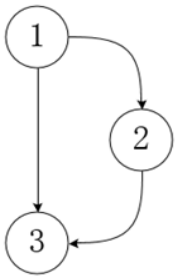

图 2-3 DAGlabel 中特殊情况示意图

直至最大的$\sigma(i)$值小于 0，或所有节点均已经标记，DAGlabel 算法结束。

该贪婪算法可以在不知道预测实例的最大标签数的情况下求出最优的分类结果，且分类结果满足层级约束要求。DAGlabel 算法通过搜索层级结构中σ值最大节点，并将其与可能违反层级约束的父节点合成新节点，来逐步简化 DAG 并求出最优分类结果，通过将σ值小的父节点与子节点进行合并来提高父节点被选为正类或子节点选为负类的可能，经 DAGlabel 算法后的分类可满足层级约束条件。算法 2-1 中描述了 DAGlabel 算法的伪代码。

Input：

H：有向无环图层级结构；σ：待分类样本在各节点的 σ 值；

Output：

$\hat{y}$：待分类样本的最终分类标签向量。

1:初始化变量L，L = {0}；初始化变量U，U = H\{0}。

2: while TRUE do

3: 找到 U 中具有最大 σ 值的节点i

4: if σ(i) < 0 then

5: return $\hat{y}$ = L

6: end if

7: if 节点 i 的所有父节点均被标记 then

8: 将 i 存入 L ，并将 i 从 U 中删除

9: else

10: 计算节点 i 的在集合 U 中父节点中 σ 值小于零的个数 N

11: if N ≥ 1 then

12: 在 U 中找到 i 的 σ 小于零的父节点集合$par_{neg}(i)$

13: if 若存在节点集合 ϕ ，该集合中的节点既是 i 的父节点，同时是 $par_{neg}(i)$ 中节点的子节点 then

14: 将集合 ϕ 中节点与节点 i 和 $par_{neg}(i)$ 中节点合成一个超级节点 i^*

15: else

16: 将节点 i 和集合 ϕ 中节点合成一个超级节点 i^*

17: end if

18: else

19: 在 U 中找到 i 的具有最小 σ 值的父节点 p

20: 将节点 i 和节点 p 合成一个超级节点 i^*

21: end if

22: 计算进行合并的各节点 σ 值的平均值作为超级节点 i^*的 σ 值 $\sigma(i^*)$

23: 将超级节点 i^* 存入 U 中

24: end if

25: end while

算法 2-1 DAGlabel **算法**

2.4 实验结果

2.4.1 DAGlabel 实验结果

将从 SVM 分类器中直接预测的结果与加入 DAGlabel 之后的结果进行对比，由于本分类问题面向层级多标签分类问题，故采用经典评价指标在层级分类领域的推广，微平均 F 值与宏平均 F 值作为评价指标。$\hat{P}_i$是第 i 个样本最详细的预测类别及所有祖先类别构成的集合，$\hat{T}_i$是第 i 个样本最详细的真实类别及其所有祖先类别构成的集合。下面给出微平均F_1值和宏平均F_1值指标的定义。

设一个数据集共包含 m 个样本，这些标签之间符合预定的层级结构关系，微平均形式下的精准率$hPre^{\mu}$、召回率$hRec^{\mu}$和F_1值hF_1^{μ}计算公式如下：

$$hPre^{\mu} = \frac{\sum_{i=1}^{m} \left|\hat{P}_i \cap \hat{T}_i\right|}{\sum_{i=1}^{m} \left|\hat{P}_i\right|} \tag{2-17}$$

$$h\mathrm{Rec}^{\mu} = \frac{\sum_{i=1}^{m} \left|\hat{P}_i \cap \hat{T}_i\right|}{\sum_{i=1}^{m} \left|\hat{T}_i\right|} \tag{2-18}$$

$$hF_1^{\mu} = \frac{2 \times hPre^{\mu} \times hRec^{\mu}}{hPre^{\mu} + h\mathrm{Rec}^{\mu}} \quad (2-19)$$

宏平均形式下的精准率$hPre^{M}$、召回率$hRec^{M}$和F_1值hF_1^{M}的计算公式如下：

$$hPre^{M} = \frac{\sum_{i=1}^{m} hPre_i}{m} \quad (2-20)$$

$$h\mathrm{Rec}^{M} = \frac{\sum_{i=1}^{m} h\mathrm{Rec}_i}{m} \quad (2-21)$$

$$hF_1^{M} = \frac{\sum_{i=1}^{m} hF_{1,i}}{m} \quad (2-22)$$

在五个数据集中进行分类，加入 DAGlabel 前后的分类结果如表 2-2 所示

Dataset	Macro.hm		Micro.hm	
	original	HMC-DAG-SVM	original	HMC-DAG-SVM
Dataset1	0.932334	0.929143	0.935129	0.931499
Dataset2	0.91604	0.91556	0.919785	0.919566
Dataset3	0.948706	0.949173	0.944818	0.945036
Dataset4	0.927479	0.926701	0.932607	0.932605
Dataset5	0.954988	0.955511	0.953178	0.953047

表 2-2 加入 DAGlabel 前后的分类结果

对比结果发现，HMC-DAG-SVM 算法在微平均 F 值与宏平均 F 值方面均略逊于基础分类器直接输出不加 DAGlabel 的结果，然而从层级多标签分类的实际意义出发，加入 DAGlabel 使得层级分类满足约束要求，避免了高层次的节点分类结果与其子节点分类结果发生冲突造成的分类无意义情况的出现。

2.4.2 对比实验

目前适用于有向无环图结构的分类算法较少，通过整理相关文献，将 DAGlabel 算法与目前较为流行的 TPR，TOP-DOWN，DOWNTOP，CLUS-HMC，R-SVM 五种算法进行对比，下面简要介绍这五种算法。

TPR 算法被称为真路径规则，整个算法过程整体上可分为两步，自底向上(downtop step)自顶向下(topdown step)，具体做法为，首先从下至上遍历整个层级结构，将下层节点的正类预测值传递给上层节点，使其对上层节点的判定产生影响。此过程结束后，再自根节点自上而下访问层级结构，将预测结果仍为负类的上层节点结果传递给相关的下层节点。

CLUS-HMC 算法是一种全局算法，方法对层级结构的不同标签设置不同权

重，生成归纳决策树一次性对全部标签进行分类，用加权的欧氏距离作为度量，并采用交叉验证的方式确定所需的层级参数。 FTest 是停止标准，只有当 FTest 在某一级子集内部方差显著减少时，节点才会被分割。CLUS-HMC 方法可以设置一组 FTest 并进行优化，在这种情况下，将选择最小的 FTest 以最小化所提供验证集上的 RMSE 度量。

HR－SVM 算法是基于阈值可调支持向量机的局部分类方法，它在分类过程中考虑了不平衡数据集的影响，即节点越远离根节点，负类样本越多，正类样本越少，越不易被标记为正，R-SVM 采用了潜在最佳阈值选择(potential best threshold selection)的策略，选择一组最重要的 SVM 阈值，然后采用最佳阈值估计(best threshold estimation)的方法计算出最佳阈值,应用到 SVM 上，它可以提升 SVM 对不平衡数据集的处理能力，在得到分类结果后，应用 TOP-DOWN 算法对整体进行修正，保证分类结果满足层级约束要求。

TOP-DOWN 方法即自顶向下方法，在确定一个样本的最终分类结果时，对有向无环图中各节点自上而下进行遍历，若样本在该节点的基础分类器给出的分类结果为正，则设定该节点的最终分类结果为正。若样本在该节点的基础分类器给出的分类结果为负，则设定该节点的最终分类结果为负，并将该节点的所有子孙节点的预测结构都设置为负。

DOWN-TOP 方法即自底向上方法，其与 TOP-DOWN 方法的处理过程正好相反。在确定一个样本的最终分类结果时，对有向无环图中各节点自底而上进行遍历，若样本在该节点的基础分类器给出的分类结果为负，则设定该节点的最终分类结果为负。若样本在该节点的基础分类器给出的分类结果为正，则设定该节点的最终分类结果为正，同时将该节点的所有子孙节点的预测结构都设置为正。

图 2-4 与图 2-5 给出基于以上六种方法的微平均 F 值与宏平均 F 值的柱状图结果对比。

2.4.3 算法评估

上述结果可以看出，DAGlabel 算法的分类效果整体上比 CLUS-HMC 的分类效果好，但仍略逊于 TPR 以及 HR-SVM 算法，考虑到 TPR 算法、TOPDOWN 算法以及 DOWNTOP 算法属于较为简单的层级约束，DAGlabel 算法仍无法获得结果上的优势，在此对 DAGlabel 算法与 TPR 算法进行对比，并从结果出发，在原理上说明 DAGlabel 的局限性。

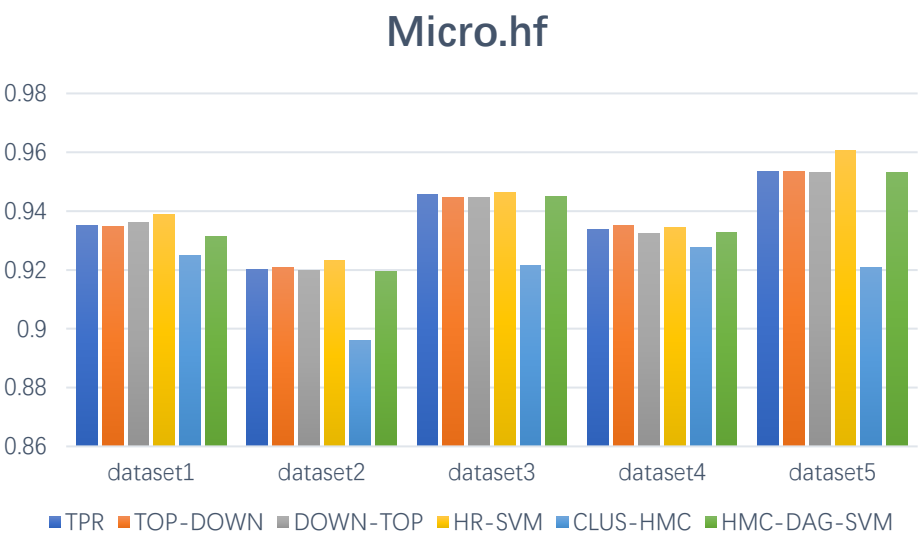

图 2-4 六种算法微平均 F 值对比

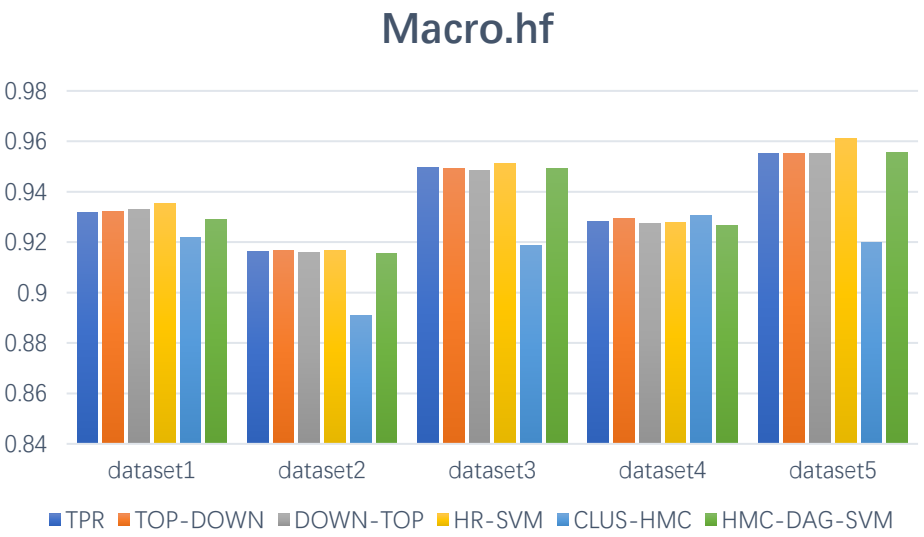

图 2-5 六种算法宏平均 F 值对比

Dataset index	tochange labels	DAG change	TPR change	DAG work	TPR work	DAG right	TPR right
1	144	237	52	144	144	43	38
2	21	143	11	21	21	12	18
3	44	145	27	44	44	18	35
4	126	170	38	126	126	58	61
5	11	90	5	11	11	9	11

表 2-2 DAGlabel 算法与 TPR 算法对节点的修正描述

tochangelabels 表示数据集中违反层级约束的标签的对数，DAGchange 和 TPRchange 是 DAGlabel 算法与 TPR 算法为使最终分类结果满足层级约束修改的标签对数，DAGwork 和 TPRwork 分别表示在需要改正的标签对中，

DAGlabel 算法与 TPR 算法的修正起作用使得修改后满足层级约束的标签对数，DAGright 和 TPRright 分别表示 DAGlabel 算法与 TPR 算法的修改使得分类正确的标签对数。

表中结果可以看出，在全部 5 个数据集中，DAGlabel 的应用修正了原来违反层级约束条件的全部标签对，使得面向 DAG 标签结构数据的分类问题结果满足层级约束条件，但相比于同样可以解决层级约束问题的 TPR 算法，在改动标签对的数量更多的情况下，改对的标签对相对较少，改错的标签对较多，造成了分类结果不如预期的情况。

从 DAGlabel 的算法实现过程出发，算法首先选取σ最大的节点进行搜寻，搜寻σ值小于 0 的父节点，把它们合成一个超级节点，若没有σ小于 0 的节点，就将其和σ最小的节点合并成超级节点。由于将多个节点合成为了一个超级节点，多个节点将会作为一个整体被标记，节点的被标记结果将全部表现为正类或负类，不存在上下两层级被标记结果为正类、负类，或被合成的σ值小于 0 的同一层级的父节点中部分被标记为正，部分被标记为负的存在。除此之外，由于σ值的大小与层级深度无关，且超级节点σ值的计算原则较为简单，猜测离根节点较近的节点发生预测错误的情况下，层级结构间可能会出现错误传递的现象，即“越分越不准”的情况存在。

3. 后期拟完成的研究工作及进度安排

3.1 后期拟完成的研究工作

根据已构建好的面向 DAG 结构的层级多标签分类模型，优化层级约束算法，在保证层级约束的同时使其在数据集上拥有更好的分类效果，并在原理上拥有可解释性；更换基础分类器，以提升基础分类的效果。

3.2 进度安排

2021 年 4 月 9 日至 2021 年 5 月 31 日 优化层级约束算法，更换基础分类分类器。撰写毕业论文

2021 年 6 月 完成论文撰写，修改论文，参与结题答辩

4. 存在的困难与问题

现有针对每个节点设计一个基础分类器的层级约束算法仍集中在有关 TPR 规则以及 TPR 规则的扩展中，此规则针对基础分类器分类过后的后验概率进行优化，可由层级关系直观调整概率大小以适合层级约束。但目前并无针对本文构建模型的其他层级约束算法，影响σ值的因素较多，无法直观的通过改变不同层级深度节点的σ值来满足层级约束，因此在算法优化方面可能存在困难。另外，由于所用数据集在不考虑层级约束的情况下，已获得较好的基础分类效果，因此在保证分类结果符合层级关系的情况下获得更好的分类效果可能存在难度。

5. 如期完成全部论文工作的可能性

目前本课题进度与此前安排基本一致。剩余的工作主要是探究算法优化方法，分析总结实验结果；撰写毕业论文。按照目前的进度，可以在剩余的时间完成毕业设计相关工作。