**基于WEKA的数据挖掘算法学习（七）**

# 多标签分类算法

## 1.1多标签分类介绍

传统的单标签分类：学习是从一个只属于一个标签l的样本集合中学习，其中每一个标签属于一个互斥的标签集合L |L| > 1。当|L|=2 时，这就是一个二分类问题，或文本和网页数据的过滤（filtering）问题。当|L| > 2 时是多分类问题。

多标签分类：每个样本属于一个L样本集合的一个子集，多标签分类(multilabel classification)分类器会给一个新的实例指定多个类别。分类模型有很广泛的实际应用，如：一个文档可能同时属于多个分类；一个蛋白质可能具有多个功能。

并且，多个标签之间可能存在一定的依赖或约束关系，如蛋白质的所有功能组成的gene ontology。这个依赖或约束关系具有层次特性，经常可以描述为树或有向无环图结构，也称为层次多标签分类。由于模型的输出具有层次结构，因此层次多标签分类又属于另外一个近来非常活跃的研究领域：结构预测。层次多标签分类和结构预测都是崭新的、富有挑战性的研究领域。在过去，多标签分类由文本分类和医学分析而产生和推动的。现在，现代的许多应用对多标签分类方法需求持续增长，比如蛋白质分类，音乐归类，和语义场景分类。

## 1.2相关/相似问题

一个同属于监督学习并和多标签分类很相关的问题就是排序问题（ranking）。排序任务是对一个标签集排序，使得排在前面的标签与相应实例更相关。

在特定分类问题中，标签属于一个层次结构（hierarchical structure）。当数据集标签属于一个层次结构的时候，我们这个任务为层次分类，如果一个样本与层次结构的多个节点相关， 那么这个任务就被称为层次多标签分类。

多实例学习（multiple-instance learning）是监督学习的一个变种，用的比较少 ，就不说了。

## 1.3多标签分类方法

方法分为两种：1、将问题转化为传统的分类问题；2、调整现有的[算法](http://lib.csdn.net/base/datastructure)来适应多标签的分类。

下面是两种方法的介绍：

问题转换法的主要思想是将多标签数据的学习转换为一个或者多个单标签数据的学习，不受特定算法的限制。由于单标签数据挖掘现在已经有很多成熟的算法，例如，支持向量机、贝叶斯分类器 、ｋ近邻方法等， 因此通过将多标签问题转化为单标签问题就可继续沿用这些传统的方法解决多标签问题。

算法适应法则是通过直接改进某种现存的单标签数据学习算法使之能够适应多标签数据的处理。

常用的转化方法有好几种：1、对每个实例确定或随机的分配一个标签，或只保留只有一个标签的训练样本，然后用传统的多分类方法去训练。这种方法会导致训练样本的损失，所以不推荐使用；2、将训练样本按每个标签构造一个训练集，每个样本属于或不属于这个标签，对每个标签单独训练一个分类器，然后将多个分类器的结果合成；3、将每个多标签单独看做一个新标签，在一个更多的标签集上做多分类。当多标签样本比较少时，这个方法就比较受限；4、对每个有多个标签的样本，复制该样本，每个样本具有不同的标签，加入训练集后使用覆盖（coverage based）分类法。

调整的算法：1、通过调整boost kNN、SVM等实现多分类，这些调整通常也会用到上面的转换。其中调整kNN实现的多标签分类可以加入先验概率，并能对输出标签排序2、基于SVM的算法中，有人在训练集中加入了|L|个二分类的训练结果，然后再进行一次分类，这个方法考虑到了不同标签之间的依赖，也是应用栈（Stacking 多个分类器的叠加）的一个特殊情况；3、还可以利用相关规则挖掘的方法

## 1.4多标签评价指标

1、**Hamming loss:** 该指标衡量了预测所得标记与样本实际标记之间的不一致程度，即样本具有标记但未被识别出，或不具有标记却别误判的可能性。越小越好。

2、**AveragePrecision:** 反映了所有样本的预测标签排序中,排在相关标签前面的也是相关标签的概率的平均。越大越好。

3、**One-error:** 度量平均对每个样本的预测标签排序中，排在第一位的标签不在该样本的相关标签集中的概率。越小越好。

4、**Coverage：**反映了平均每个样本的预测标签排序中，需要在标签排序列表中最少查找到第几位才可找出所有与该样本相关的标签。越小越好。

5、**Ranking loss:** 反映了所有样本的预测标签排序中，不相关标签排在相关标签前面的概率的平均。越小越好。

# 算法介绍

## 2.1 BR

BR二元关系法，为每一个标签产生一个单独的数据集，总共产生n个数据集(j=1…n)，每个数据集包含训练集中的全部样本。对于数据集，包含在该数据集中的所有样本仅被标注是否属于，用该数据集训练一个仅输出样本是否属于的二分类器，为每一个标签训练一个二分类器。对于一个未知类别的样本，将全部二分类器结果组合在一起即得到该样本最终的标签分类结果。

## 2.2 ECC

ECC组合分类器链是对于BR方法的一种改进，针对BR未能考虑标签之间的联系,而导致一定程度上的信息损失,采用了CC(ClassfierChain)分类器链，即通过将BR产生的ｎ 个二分类器连接成一条链，训练样本每经过一个二分类器， 就将其预测结果添至样本属性向量继续代入下一个二分类器中训练。但是由于CC中二分类器不同顺序对结果有较大影响，因此ECC采取多条随机产生的不同标签序列的CC组合,以减轻单个CC由内部二分类器排列顺序问题而带来的不利影响。

# 3.weka算法实现

## 3.1数据集的准备

1、MULAN多标签数据集：emotions

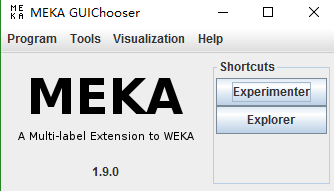
2、MULAN多标签数据集：flags

## 3.2 算法实现

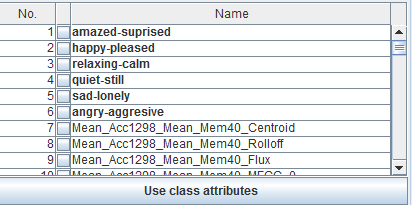
### 3.2.1 MEKA的使用

1、由于WEKA不能处理多标签数据集，首先在http://sourceforge.net/projects/meka/files/ 下载MEKA: A Multi-label Extension to WEKA。

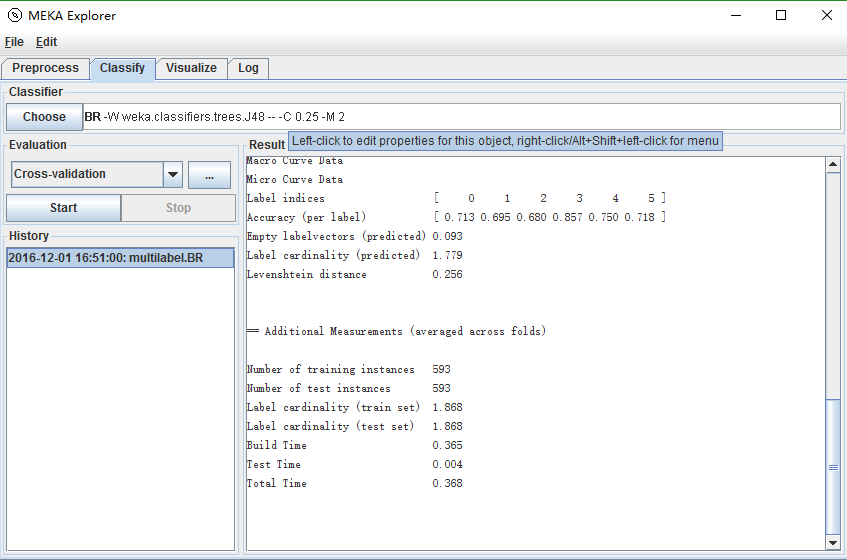
2、解压压缩包，运行，得到和WEKA很相似的MEKA的可视化界面：



3、打开Explorer，导入多标签数据集emotions，首先选择好数据集中的多标签（emotions中有六个），下图所示加粗为选择的class attributes：



4、然后选择多标签算法进行运行，和weka使用方法类似：



### 3.2.2 BR

首先选择多标签数据集emotions，选择MEKA Explorer->Classify->BR，选择五折交叉验证，设置参数classifiter=J48-C 0.25-M 以及classifiter=RandomForest-l 100-K 0-S 1,进行数据集测试，结果如下所示：

|  |  |  |
| --- | --- | --- |
|  | emotions数据集 | |
| J48基分类器 | RandomForest基分类器 |
| Hamming Loss | 0.264 | 0.181 |
| one-error | 0.447 | 0.248 |
| Ranking loss | 0.327 | 0.142 |
| Avg Precision | 0.475 | 0.47 |
| time | 0.347s | 2.47 |

对于emotions数据集，Avg Precision值在J48基分类器下的BR和RandForest基分类器下的BR算法结果上相差很小，但选择RandForest基分类器时其余指标均低于J48基分类器，但从时间效率上来说，J48远远优于RandomForest。

选择多标签数据集flags，选择MEKA Explorer->Classify->BR，选择五折交叉验证，设置参数classifiter=J48-C 0.25-M 以及classifiter=RandomForest-l 100-K 0-S 1,结果如下所示：

|  |  |  |
| --- | --- | --- |
|  | flags数据集 | |
| J48基分类器 | RandomForest基分类器 |
| Hamming Loss | 0.203 | 0.21 |
| one-error | 0.345 | 0.211 |
| Ranking loss | 0.206 | 0.16 |
| Avg Precision | 0.424 | 0.398 |
| time | 0.029 | 0.665 |

根据跑出来的实验结果来看，flags数据集和emotions结果特点类似，只是flags数据集下，各类评价指标相对来说更加接近。

### 2.2.3 ECC

在2.2中提到，由于BR算法将问题转换为q个二分类问题处理，使得它忽略了标签之间的关联关系。CC的提出是为了解决BR没考虑标记之间的关联性这一缺点。

首先选择多标签数据集emotions，选择MEKA Explorer->Classify->BaggingML，选择五折交叉验证，设置参数classifiter=J48-C 0.25-M 以及classifiter=RandomForest-l 100-K 0-S 1,进行数据集测试，结果如下所示：

|  |  |  |
| --- | --- | --- |
|  | emotions数据集 | |
| J48基分类器 | RandomForest基分类器 |
| Hamming Loss | 0.197 | 0.178 |
| one-error | 0.28 | 0.263 |
| Ranking loss | 0.163 | 0.176 |
| Avg Precision | 0.459 | 0.426 |
| time | 2.995 | 12.295 |

对于emotions数据集， J48基分类器下的BR和RandForest基分类器下的BR算法结果上相差很小，但是从时间效率上来说，在RandomForest基分类器下耗费太多时间。

首先选择多标签数据集emotions，选择MEKA Explorer->Classify->BaggingML，选择五折交叉验证，设置参数classifiter=J48-C 0.25-M 以及classifiter=RandomForest-l 100-K 0-S 1,进行数据集测试，结果如下所示：

|  |  |  |
| --- | --- | --- |
|  | flags数据集 | |
| J48基分类器 | RandomForest基分类器 |
| Hamming Loss | 0.2 | 0.204 |
| one-error | 0.186 | 0.175 |
| Ranking loss | 0.163 | 0.173 |
| Avg Precision | 0.411 | 0.429 |
| time | 0.251 | 4.113 |

对于flags数据集，结果上和emtions很相似，二者差距很小，时间效率上在J48基分类器下比RandomForest基分类器优越很多。

通过BR多标签分类算法和ECC多标签分类算法跑出来的结果对比来，ECC算法性能确实优于BR分类算法，也印证了前面理论知识介绍，ECC是BR方法的一种改进，针对BR未能考虑标签之间的联系,而导致一定程度上的信息损失,采用了CC(ClassfierChain)分类器链。对比数据评价指标结果，确实有了提升。

# 4.本周学习心得体会

本周学习了多标签分类算法的数据测试，由于weka中不能实现多标签分类，进行了meka的下载、学习与使用，由于meka目前只有英文文献介绍使用方法，学起来费劲一些。并通过下载论文：<多标签数据挖掘技术： 研究综述>来学习了多标签学习的理论知识。