基于关联规则的分类模型系统①

Classification Modeling System Based on Association Rule

晁玉宁　许孝元　(广东工业大学 计算机学院 广东 广州 510006)

摘　要： 关联分类是数据挖掘及机器学习领域的一个研究热点。利用原子关联分类算法（CAAR）建立了数据模型的机器学习系统，详细说明了 CAAR 算法的分类步骤并给出了算法的伪代码表示。在 UCI 提供的标准数据集上进行测试，实验验证了在大规模数据集中，在不同的抽样率情况下，原子关联分类算法的分类准确度，用数据的方式与其他分类算法做了比较。对数据集记录次序的依赖性进行的 10-折交叉验证实验表明，原子关联分类算法的分类准确度要高于 CBA 算法。

关键词： 数据挖掘 原子关联规则 分类 机器学习

数据挖掘就是从大量数据中提取或“挖掘”有用的知识，因此数据挖掘又被称为数据库中的知识发现。关联分类是数据挖掘及机器学习领域内的一个很重要的研究热点。许多实验也都证明了关联分类具有良好的分类准确性。

利用简易规则构造分类模型，可以追溯到 1993 年出现的 1R(或 OneR)分类算法[1] 。该算法的基本思想是对数据集上的各个属性创建单层的决策树，然后计算训练集上达到的错误率，最终选择错误率最低的单一属性产生的单层决策数作为分类策略。在 1R 算法的改进算法中[2]，引入了加权机制，如果有两个或两个以上的规则有相同的最小错误率，则选择权重较大的规则作为分类规则。1998 年，Liu 提出了类关联规则分类算法 CBA[3]。该算法使用类 Apriori 算法产生分类规则，当遇到高维数据库时，会遗失部分长模式的优质规则，降低模型的分类准确度。此后，陆续有人针对 CBA 的不足提出了各种关联分类算法，典型的有 CMAR[4]、CPAR[5]、基于概念格的关联规则分类法[6]等。但这些关联分类算法并没有解决分类模型准确度低，执行效率低，系统资源消耗大的问题。2004 年，提出了采用多步原子关联规则的分类新技术 CAAR[7](Classification based on Atomic Association Rules)，消除了关联规则挖掘时的“组合爆炸效应”。就算法的性能来看 CAAR 有不俗的业绩，其分类的准确度和模型的可理解性都优于决策树与 CBA。本文首先介绍 CAAR 算法的分类原理及系统设计，再从实验的角度，在 UCI 提供的标准数据集上进行测试，比较 CAAR 与其他算法的分类准确度。

1　CAAR算法理论(原子关联规则分类算法)

1.1 原子关联分类的相关定义

定义 1.　关联规则：它表现为项集之间的关联关系。关联规则 r 表示为：<LHS, RHS>，其中 LHS 是一个项集，称为规则的前件或条件，RHS 是一个类标签，成为规则的后件或结论，可以表示为 LHSRHS。

定义 2.　支持度(sup)：指数据集中包含 LHS 及类标签 RHS 的实例数与数据集中总的实例数 D 的比值。

定义 3.　置信度(conf)：指数据集中包含 LHS 及类标签 RHS 的实例数与只包含 LHS 的实例数的比值。

定义 4.　原子型分类关联规则：指规则的前件只有一个项的分类关联规则，即|LHS|=1。

定义 5.　强关联规则：指支持度大于 minsup 且置信度大于 minconf 的关联规则。

在 CAAR 算法中，minconf=k×maxconf(一般 k=0.98)，其中 maxconf 是候选原子关联规则的最高置信度。我们只选择具有最高置信度和接近最高置信度的强关联规则用于分类器的构建。

① 收稿时间:2008-11-11

© 中国科学院软件研究所 http://www.c-s-a.org.cn

1.2 CAAR 算法描述

CAAR 算法的出现不是偶然的，它是在发现知识要点的背景下提出的。CAAR 算法的分类思想是，利用人类在区分事物时采取“利用突出特征和先易后难策略”的分类原理。对数据集进行几次部分分类，直到全部实例分类完毕。CAAR 算法重要的第一步是产生强原子规则，用于原子规则产生的数据结构是一个 Counter 类，它用一个三维数组去存储与原子规则有关的 2-项集(r.LHS^r.RHS)的计数。这个类用 Java 语言定义如下：

```
public class Counter {
  public int [ ][ ][ ] count = new int [m][nV][nC];
  public void counting (DataSet d) {...}
  public void counting (Instance inst) {...}
  public int getCount (int X, int Vx, int c) {...}
  public int getCount (int X, int Vx) {... }
}
```

这里，m=属性个数，nC=类标签个数。nV 表示最大的属性值域空间大小。重载的 Counter 类的 getCount 方法既能返回候选原子规则前件项(r.LHS)的计数，也能返回原子规则 r 对应 2-项集(r.LHS^r.RHS)的计数，通过三个下标值存取原子规则的计数。

CAAR 算法产生分类器的步骤如下：

输入：数据集 D，最小支持度 minsup 和最小置信度 minconf。

输出：分类规则集。

(1) 挖掘原子型分类关联规则，选择具有最高置信度和接近最高置信度的强原子规则用于部分分类。

(2) 为了与其他关联分类算法进行比较，对强原子关联规则按置信度为第一关键字、支持度为第二关键字进行降序排序(更好的排序方法是文献[8]中指出的)；接着，在数据集上测试强原子关联规则集，删除被规则覆盖的实例；当数据集扫描完毕时，删除冗余规则。

(3) 对余下的数据，继续上述部分分类过程，直到数据集的实例数为零。最后顺序组合各阶段的分类规则，得到分类模型。

在每次部分分类中，在测试强关联规则集选择分类规则时，同时进行下一次部分分类有效实例的计数，因此，减少了对数据集的扫描次数。

算法的伪代码表示如下：

```
Gen_Classifier (DataSet D0)
①RuleSet classifier ← φ ; Dx= D0    // Dx 为部分分类的数据集
②Counter counter ← new Counter ()
③counter.counting (D0)
④WHILE | Dx |>0 AND NumOfClass (Dx) >1 DO
⑤RuleSet ← sarGen_SAR (Dx, counter) // sar 为强原子规则集
⑥counter.clear()    //  为下一次计数做准备
⑦RuleSet  cr ← PruneRules(sar, Dx,counter) //在 PruneRules 中更新 Dx
⑧classifier ← classifier ⋃ cr    // 将部分分类的分类规则集加入分类器
⑨END DO
⑩IF  |Dx|>0  THEN    // 产生一个指示缺省类的规则
⑪classifier ← classifier ⋃ {<指示缺省类的规则>}
⑫END IF          // 在预测时总是取最后一个规则的结果作为缺省类
⑬return classifier
```

首先调用 Counter 类的 counting 方法扫描数据集 D (第 3 行)，对所有出现的项以及与原子规则对应的 2-项集计数。 NumOfClass 函数获得数据集中实例的种类数 (第 4 行)。Gen_SAR 函数通过计数器产生强原子规则 (第 5 行)。PruneRules 函数对冗余的规则剪枝，并删除数据集中被规则覆盖的实例；同时，对没有被规则覆盖的记录进行计数，为下一次的部分分类做准备。分类器由每一步部分分类产生的分类规则集顺序组合而成。当数据集中实例数为 0 时，则完成分类器的构造。或者，数据集中的实例数不等于 0 且其中的所有实例都属于一个类时，产生一个指示缺省类的规则(第 10-12 行)，完成分类器的构造。

CAAR 分类算法与文献[1]中的 1R 算法提取的规则都属于简单规则，得到的分类模型简单易于理解，并且模型不易出现过学习现象。但它们也存在以下的不同点：

(1) 分类机理不同：1R算法基于简单的统计计数，规则局限于某一个属性，是一个粗糙的单步分类过程。CAAR 算法基于人工智能原理，模仿人类区分事物时利用突出特征与先易后难策略进行分类，是精细的多

© 中国科学院软件研究所 http://www.c-s-a.org.cn

步分类过程。

(2) 规则的度量方法不同：1R 算法仅采用基于计数的误差值选择分类属性。而 CAAR 使用多项数据挖掘技术，利用具有最高置信度和接近最高置信度的优质原子规则构建分类器。

(3) 分类准确度不同：1R 算法及其改进算法在平均分类准确度上都不及决策树，而 CAAR 算法的平均分类准确度显著高于决策树及关联分类基准算法 CBA[7]。

2 系统设计与实现

原子关联分类算法机器学习系统的功能就是利用 CAAR 分类算法实现对装入的数据集进行分类关联规则的提取，构建分类器能够正确的预测未知样本的类标号。此系统还可以显示，所测试数据集进行部分分类的次数，及每次分类所提取的原子规则，预测模型的分类准确度和建模时间。机器学习系统主要由以下类函数实现：

App：是对算法的应用也是学习模型的主函数。在此函数中选择要测试的数据集。

CAAR：是对原子关联规则算法的实现。用于产生原子规则，构建分类器，并计算建立分类模型的时间。

Global：是用于对数据集进行抽样的类函数，然后对抽样到的数据进行训练和测试。第 4 部分将列出在大型数据集进行抽样，不同抽样率下，不同分类算法的分类准确度的比较。

Counter：是对候选原子规则前件和原子规则 r 对应 2—项集计数的类函数。该函数中实现按置信度是第一关键字、支持度是第二关键字对强原子规则进行排序。

Partition：是一个对数据集进行分区的类函数。将数据集一部分作为训练集，剩余的做测试集。通常用到的是 10-折交叉验证，是把数据集分为大小相同的 10 份，选择其中一份作测试集，而其余的 9 份全作训练集。该过程重复 10 次，使得每份数据用于测试恰好一次。

Attributeset：是用于存储数据集中属性集的类函数，其用哈希表结构实现。

本文所做的实验都是在微机上进行的，处理器为：Pentiuml1.60GHZ，内存为 512M。操作系统：Microsoft XP Professional，CAAR 算法采用 Java 语言编写，实现平台为 JBuilder 9.0。

3 实验结果与分析

文献[4]中给出了 CAAR 算法在 26 个不同类型数据集上的实验测试结果。这里利用抽样技术对大规模数据集在不同采样点的分类准确度进行测试。目前，对于大规模数据库知识发现，抽样技术是一种行之有效的方法。这里对一个实际大规模数据集 Census-Income 进行不同采样点的测试。该数据集取自 UCI(http://kdd.ics.uci.edu/databases/census-income /census-income.html)提供的标准大规模机器学习测试数据库。数据集包含 199523 个实例，有 41 个属性，调查的对象分为两类：一类是高于 50000，另一类是低于 50000。实验中，数据集的连续属性均采用基于熵的方法进行离散化。

表 1　算法在不同采样点下的分类准确度

	10	100	500	1000	2000	3000	4000	5000
记录数	19952	1995	399	199	99	66	49	39
CAAR	95.31	97.24	98.75	98.99	98.99	100.00	100.00	100.00
CBA	93.71	98.45	100.00	100.00	100.00	100.00	100.00	100.00
J48	95.04	93.99	89.44	89.44	89.90	95.45	87.75	94.87

本文利用 C++实现的 CBA 算法及决策树中的 J48 算法进行对比测试，在不同抽样率情况下应用 CAAR、CBA 及 J48 算法对所采集到的数据进行训练，并在该数据集上进行测试。这里 CAAR 算法的动态支持度为 0.01，其他算法的实验参数取缺省值(例如，CBA算法中规则前件最大长度为 6)。以下是实验结果：

由表 1 可以得出，在大规模数据集中 CBA 算法的分类准确度不及 CAAR 算法，因为 CBA 算法规定了挖掘的规则不能超过 80000 个，规则左部的属性个数最大为 6 个，当遇到高维数据库时，当属性个数达到 3 个或 4 个时规则数过多就停止了，不能挖掘出完整的规则集，导致分类准确度的降低。当数据集中实例个

© 中国科学院软件研究所 http://www.c-s-a.org.cn

数减少时，CAAR 和 CBA 的分类准确度相差不多，对于较小数据集，两者的分类准确度相等，但是 CBA 算法分类模型学习时间却比 CAAR 长的多[7]。由表 1 还可以看出，抽样率对 J48 分类的影响较大，它的分类效果对抽样率的变化比较敏感。对任何抽样率，J48 的分类准确度都不如 CAAR。这是因为决策树 J48 采用基于熵的技术，在类分布严重不均衡的情况下，J48 有忽略少数类的缺点，从而降级分类准确度。在测试中，CAAR 算法采用动态支持度，在类分布严重倾斜，抽样率较低的情况下，仍然能提取有效的分类模型，达到 100%的分类准确度。

这里再对 UCI 提供的 zoo 数据集进行 10－折交叉验证测试，对比 CBA 算法与 CAAR 算法的分类准确度。该数据集有 101 个实例，无属性值的缺失。先对 zoo 数据集进行随机洗牌，使实例随机的变换顺序，再对该数据集进行 10－折交叉验证测试，计算测试的平均分类准确度。表 2 是对上述过程重复 21 次的实验结果：

表 2　Zoo 数据集随机洗牌后 10-折交叉验证测试的分类准确度

算法	数据集随机洗牌后 10-折交差验证测试的分类准确度										
	1	2	3	4	5	6	7	8	9	10	11
CAAR	97.09	93.00	96.00	95.98	95.00	95.98	94.00	95.89	94.89	96.00	96.00
CBA	92.09	95.00	92.00	95.00	92.18	95.27	92.00	95.09	94.00	94.09	96.00
算法	数据集随机洗牌后 10-折交差验证测试的分类准确度										
	12	13	14	15	16	17	18	19	20	21	
CAAR	92.87	97.00	95.89	94.09	97.98	93.89	95.00	95.89	95.89	96.09	
CBA	93.00	94.00	92.00	96.09	94.09	93.09	94.00	92.09	95.00	93.09	

由表 2 可以看出，CAAR 算法赢 17 局,负 3 局，平 1 局。21 次 10-折交叉验证的平均预测准确度：CAAR 为 95.45%，CBA 为 93.77%，CAAR 最高的分类准确度已高达 98%。可见，CAAR 算法的分类准确度是优于 CBA 算法的。

4　结论

关联分类的分类准确度和分类模型的稳定性一直是研究人员关注的问题。本文给出了 CAAR 算法构建分类器的步骤，主要利用抽样技术，在大规模数据集中进行测试，验证了 CAAR 分类算法的有效性。对 zoo 数据集进行的 10-折交叉验证也表明 CAAR 的分类准确度明显高于 CBA 算法。原子关联分类算法的分类模型简单，易于理解，算法本身具有内在加速特性，执行速度快且分类准确度较高，在数据挖掘中具有重要的应用价值。

参考文献

1 Holte CR. Very simple classification rules perform well on most commonly used datasets. Machine Learning, 1993,(11):63－90.

2 Buddhinata G, Derry D. A Simple Enhancement to One Rule Classification. http: //www.buddhinath.net/Other-Links/Documents/Improved%20OneR%20Algorithm.pdf.

3 Liu B, Hsu W, Ma Y. Integrating classification and association rule mining. Proceedings of 4th ACM Int. Conf. on KDD, New York, 1998:80－86.

4 Li WM, Han JW, Pei J. Cmar:Accurate and efficient classification based on multiple class-association rules. Proceedings of the IEEE Int. Conf. on Data Mining, California, 2001:369－376.

5 Yin XX, Han JW. Cpar:classification based on predict-tive association rules. Proceedings of the 2003 SIAM International Conference on Data Mining (SDM'03),San Francisco, 2003:34－42.

6 胡可云,陆玉昌,石纯一.基于概念格的分类和关联规则的集成挖掘方法.软件学报,2000,11(11):1478－1484.

7 Xu XY, Han GQ, Min HQ. Construct concise and accurate classifier by atomic associa- tion rules.Proc of the Third Into Conf on Machine Learning and Cybernetics, New York: IEEE Press, 2004,8:1604－1609.

8 许孝元,韩国强,闵华清.多步原子规则的大规模关联分类.控制理论与应用, 2007,(6):471－474.

© 中国科学院软件研究所 http://www.c-s-a.org.cn