文章编号：1007-1423(2014)15-0007-05 DOI：10.3969/j.issn.1007-1423.2014.15.002

基于决策树的 ID3 算法和 C4.5 算法的比较

苗红星[1]，余建坤[2]

（1. 云南财经大学信息学院计算机系，昆明 650000；2. 云南财经大学信息学院，昆明 650000）

摘要：

阐明决策树分类器在用于分类的数据挖掘技术中依然重要，论述基于决策树归纳分类的 ID3、C4.5 算法，并且对决策属性的选取法则进行说明。通过实例解析 ID3、C4.5 算法实现过程，结果表明 C4.5 算法相比较于 ID3 算法的优越性，尤其在处理具有多属性值的数据时的更加合理和正确。

关键词：

数据挖掘；决策树；ID3 算法；C4.5 算法

0 引言

决策树方法是挖掘分类规则的有效方法，通常包括两个部分：①树的生成，开始时所有的数据都在根节点，然后根据设定的标准选择测试属性，用不同的测试属性递归进行数据分割；②树的修剪，就是除去一些可能是噪音或异常的数据。基于信息熵的 ID3 算法、C4.5 算法都能有效地生成决策树，建决策树的关键在于建立分支时对记录字段不同取值的选择。选择不同的字段值使划分出来的记录子集不同，影响决策树生长的快慢及决策树的结构，从而可寻找到规则信息的优劣。可见，决策树算法的技术难点就是选择一个好的分支取值。利用好的取值产生分支可加快决策树的生长，更重要是产生好结构的决策树，并可得到较好的规则信息。相反，若根据一个差的取值产生分支，不但减慢决策树的生长速度，而且使产生的决策树分支过细、结构差，从而难以发现有用的规则信息。

随着训练样本集中样本个数的不断增多（即样本集规模不断扩大），训练样本集在主存中换进换出就耗费了大量的时间，严重影响了算法效率。因此使算法能有效处理大规模的训练样本集已成为决策树算法研究的一个重要问题，也是目前国内对决策树算法研究的热点。本文结合实例数据，介绍了 ID3 算法与 C4.5 算法的实现过程，并进行了比较分析。

1 ID3 与 C4.5 算法简介

1.1 ID3 算法

ID3 算法通过对一个训练例集进行学习生成一棵决策树，训练例集中的每一个例子都组织成属性-属性值对的形式。假设一个例子仅属于正例（符合被学习目标概念的例子）或反例（不符合目标概念的例子）两种分类之一，例子的所有属性都为离散属性。对于每个训练例集 E_s，如果正例的比例为 P_+，则反例比例就为 $P_-=1-P_+$，熵的公式为：

$$\text{Entropy}(E_s)=-P+\log_2 P_+-P_-\log_2 P_-$$

（这里约定 $\log_2 0=0$） （1）

若用属性 A 将训练例集 E_s 分组，$\text{Entropy}(E_s)$ 将会降低，新的期望信息量设为：

$$\text{New Entropy}(E_{s_i},A)=$$

$$\sum_{i\in Value(A)}(|E_{s_i}|/|E_s|)\text{Entropy}(E_{s_i}) \quad (2)$$

A 相对于 E_s 的信息赢取 Gain（E_s，A），即 Entropy（E_s）降低的数量，信息赢取越大的属性对训练例集越有利：

$$\text{Gain}(E_s,A)=\text{Entropy}(E_s)-\text{New Entropy}(E_s,A) \quad (3)$$

1.2 C4.5 算法

信息增益率度量偏向具有许多输出的测试。换句话说,它倾向于选择具有大量值的属性。这种划分对于分类没有用。

在 C4.5 的决策树中,每个节点都保存了可用于计算 E 值属性的信息,这些信息由属性的每个取值所对应的正例与反例计数组成。根据放在节点的信息就可判断哪个属性的训练例集 E_s 值最小,从而确定当前用某个属性进行划分。C4.5 算法属性选择基础是基于使生成的决策树中节点所含的信息熵最小。熵越小则记录集合的无序性越小,即记录集合内的属性越有顺序越有规律。集合 S 的熵计算公式为:

$$\text{Info}(S)=-\sum_{i=1}^{k}(freq(C_i,S)/|S|)\times\log_2(freq(C_i,S)/|S|) \quad (4)$$

式中,$freq(C_i,S)$为集合 S 中属于类 C_i (k 个可能类中的一个) 的样本数量;$|S|$ 为集合 S 中的样本数量。子集进行熵的加权和的计算公式为:

$$\text{Info}_x(T)=-\sum((|T_i|/|T|)\text{Info}(T_i)) \quad (5)$$

式中,T 为按照属性 x 进行分区的集合。计算分区前的集合的熵和分区后的熵的差(增益),增益大的就是要选取的节点。

C4.5 算法将分类范围从分类的属性扩展到数字属性。如果数据集中存在连续型的描述性属性(数字属性),C4.5 算法首先将这些连续型属性的值分成不同的区间,即"离散化"。通常将连续型属性值"离散化"的方法为:①寻找该连续型属性的最小值,并将它赋值给 min,寻找该连续型属性的最大值,并将它赋值给 max;②设置区间[min,max]中的 N 个等分断点 A_i,其中,i=1,2,…,N;③分别计算把[min,A_i]和(A_i,max)(i=1,2,…,N)作为区间值时的 Gain 值,并进行比较;④选取 Gain 值最大的 A_k 作为该连续型属性的断点,将属性值设置为[min,A_k]和(A_k,max)两个区间值。

2 实例解析

2.1 ID3 算法实现

通过某个动物园的数据实例说明 ID3 算法的实现过程以及结果分析。其样本数据见表 1。通过各种属性来判断每种动物属于哪种类型的动物,有 17 个属性和一个类标号属性,例如是否有羽毛、是否哺乳、是否有牙齿、是否有鳍、是否有尾巴等属性来判断,类标号属性有哺乳动物、鸟类、爬行动物、鱼类、两栖动物、昆虫、无脊椎动物。总共 101 条数据,数据如表 1。

一开始全部包含在根节点中,为找当前的最佳划分属性,先必须根据式(1)计算训练集 E_s 的熵值。节点的熵值为:

$$\text{Entropy}(E_s)=2.3906$$

下一步,需要计算每个属性的期望信息需求。从属性 animal name 开始。使用 animal name 属性划分的信息增益是:

$$\text{Gain}(E_s,animal\ name)=2.3906$$

类似的,可以计算 $\text{Gain}(E_s,hair)=0.7907$, $\text{Gain}(E_s,feathers)=0.7179$, $\text{Gain}(E_s,eggs)=0.8301$, $\text{Gain}(E_s,milk)$

表1

animal	hair	feathers	eggs	milk	airborn	aquatic	predator
aardva	true	false	fals	true	false	false	true
antelop	true	false	fals	true	false	false	false
...	.						
...	.						
worm	fals	false	true	false	false	false	false
wren	fals	true	true	false	true	false	false
toothed	backbone	breathes	venomous	fins	legs	tail	domestic
true	true	true	false	fals	4	fals	false
true	true	true	false	fals	4	true	false
false	false	true	false	fals	0	fals	false
false	true	true	false	fals	2	true	false

表 2 各种属性的划分增益

Atrribute	animal	hair	feathers	eggs	milk	airborne	aquatic	predator	toothed
Gain	2.3906	0.7807	0.7179	0.8301	0.9743	0.4697	0.3895	0.0934	0.8657

Atrribute	backbone	breathes	venomous	fins	legs	tail	domestic	catsize
Gain	0.6762	0.6145	0.1331	0.4666	1.3630	0.5005	0.0507	0.3085

= 0.9743,等等。如表 2。

由表 1 可以得到 Gain(E_s,*animal name*)=2.3906,是最高的信息增益，所以选择 *animal name* 为分裂属性。但是可以看到 Gain (E_s,*animal name*)=2.3906 相等于 Entropy(E_s)=2.3906,说明 New Entropy(E_{s_i},*animal name*)=0,对 *animal name* 的分裂将导致大量划分(与值一样多),每个只含有一个元组。由于每个划分都是纯的,因此给予该划分对数据集 E_s 分类所需要的信息 New Entropy(E_{s_i},*animal name*)=0。这样一来,对该属性划分得到的信息量最大,显然这种划分对于分类是没有用的。

2.2 C4.5 算法的实现

通过 2.1 已经知道 Entropy(E_s)=2.3906,下面来计算每个属性用 C4.5 算法计算 GainRatio(E_s,*Atrribute*),然后找出最大的增益率当做分裂属性。

先计算分裂信息 SplitInfo(E_s,*Atrribute*)计算以后得到：

SplitInfo(E_s,*animal*)=6.6384

下一步需要计算属性分裂 *animal* 的信息增益率 GainRatio(E_s,*animal*),计算结果如下：

GainRatio(E_s,*animal*)=Gain(E_s,*animal*)/SplitInfo(E_s,*animal*)=0.3601

类似的也可以计算出其他属性的 GainRatio(E_s,*hair*)=0.8035，GainRatio(E_s,*feathers*)=1,等等。具体如表 3。

根据在 MatLab 中的代码实现。分类以后并且使用 IF THEN 规则的结果如以下程序所示。

```
>> C45(data)
if backbone=true and
if feathers=false and
if milk=true then type=mammal
if milk=false and
if fins=true then type=fish
if fins=false and
if tail=false then type=amphibian
if tail=true and
if aquatic=true and
if legs=4 then type=amphibian
if legs=0 then type=reptile
if aquatic=false then type=reptile
if feather=true then type=bird
if backbone=false and
if airborne=false and
if predator=true then type=invertebrate
if predator=false and
if legs=6 then type=insect
if legs=0 then type=invertebrate
if airborne=true then type=insect
```

由以上程序以及 GainRatio(E_s,Atrribute)可以得出第一个分裂属性是 backbone，然后依次得出决策树如图 1。

由 C4.5 的分类结果可以看到,C4.5 的分类更加准确,有效地避免了 ID3 偏向具有许多输出的测试。而且 C4.5 算法也可以解决了 ID3 算法无法描述属性连续型的情况,在实现过程中利用 C4.5 算法建立决策树的速度较 ID3 算法迅速，而且决策树结构也较 ID3 算法合理,同时也找到较好的规则信息。

然而，随着分裂信息趋向于 0，该比例变得不稳定。为了避免这种情况,增加一个约束,选取测试的信

表3

Atrribute	animal	hair	feathers	eggs	milk	airborne	aquatic	predator	toothed
GainRatio	0.3601	0.8035	1	0.8475	1	0.5938	0.4145	0.0943	0.8938

Atrribute	backbone	breathes	venomous	fins	legs	tail	domestic	catsize	Atrribute
GainRatio	1.0000	0.8332	0.3332	0.7137	0.6702	0.6082	0.0915	0.3122	GainRatio

息增益必须较大，至少与所考察的所有测试的平均增益一样大。

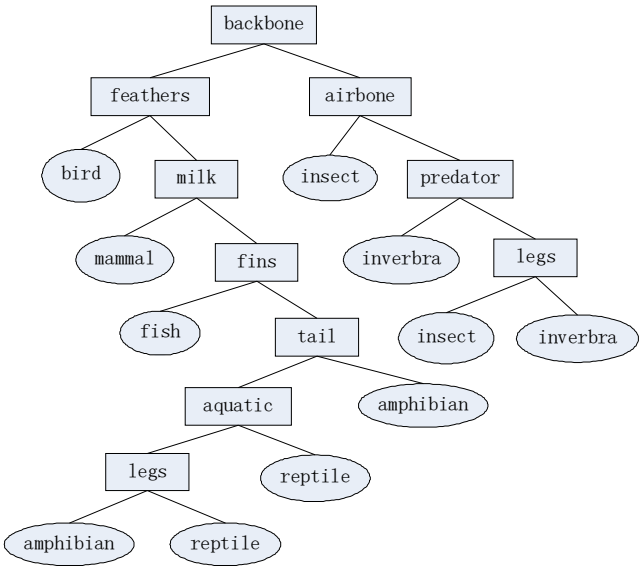

图1

3 结语

ID3 通过循环处理，逐步求精，直至找到完全正确的决策树。ID3 算法不存在无解的危险;全盘使用训练数据，而不是像侯选剪除算法逐个地考虑训练例，从而抵抗噪音。ID3 倾向于选择取值较多的属性，却不是最优的属性，这样就有可能得到局部最优解而失去全局最优解；在搜索过程中无回溯;ID3 不能增量地接受训练集，每增加一次实例就抛弃原有的决策树，重新构造新的决策树，开销很大。

C4.5 算法为 ID3 算法的扩展，有如下改进:①用信息增益率选择属性，克服了用信息增益选择属性时偏向选择值多的属性的不足;②可处理连续数值型属性;③为避免树的高度无节制的增长和过度拟合数据，采用了从“规则后修剪”方法演变而来的后剪枝方法。该法使用训练样本集本身估计剪枝前后的误差，从而决定是否真正剪枝;④对于缺失值的处理。利用 C4.5 算法可提高决策树生长速度，优化决策树结构，挖掘较好的规则信息。挖掘的数据越多，算法的效率和性能越好，算法的优越性越明显。

决策树算法有广泛的应用，ID3 算法与 C4.5 算法是决策树中基础、经典的算法.C4.5 算法在 ID3 算法基础上进行了改进，但是 C4.5 算法还是不够稳定，精度也不是最高，后续发展为 C5.0 算法.C5.0 算法使用推进技术，有效改善了 C4.5 算法预测精度的缺点以及决策树的性能。

在长时间来看，如何发掘更好的数据预处理方法来支持决策树;如何提高决策树预测精度，获得更好的分类效果;如何更好地简化决策树方法，都需要更深入的研究。

参考文献:

[1]Goebel M，Gruenwald L. A Survey of Data Mining and Knowledge Discovery Software Tools[J]. SIKDD Explorations，1999，1(6):22~33

[2]Quinlan J R. Induction of Decision Trees[J]. Machine Learning，1986:84~101

[3]Quinlan J R. C4.5:Programs for Machine Learning[M].San Mateo，Calif:Morgan Kaufmann，1993:35~55

[4]Han Jiawei，Micheline K. 数据挖掘:概念与技术[M]. 范明，孟小峰译. 北京:机械工业出版社，2001:70~218

[5]Xingdong Wu，Vipin Kumar，J.Ross Quinlan.Top 10 Algorithms in Data Mining[M]. London:Springer-Verlag London Limited，2007

[6]刘小虎，李生. 决策树的优化算法[J]. 软件学报，1998，9(10):797~800

[7]王晓国，黄韶坤，朱炜等. 应用 C4.5 算法构造客户分类决策树的方法[J]. 计算机工程，2003，29(14):89~91

[8]Kantardzic Mehmed. 数据挖掘——概念，模型，方法和算法[M]. 闪四清，陈茵，程雁译. 北京:清华大学出版社，2004

[9]陈文伟，黄金才，赵新显. 数据挖掘技术[M]. 北京:北京工业大学出版社，2004

[10]Agosta Lou. 数据仓库技术指南[M].潇湘工作室译. 北京:人民邮电出版社，2001

[11]丁华，张少中，王秀坤. 基于改进 ID3 算法的轨迹化决策研究[J]. 计算机工程与设计，2004，25(10):1721~1 723

作者简介:

苗红星(1986-)，男，河南郑州人，在读硕士研究生，研究方向为数据挖掘

余建坤(1962-)，男，云南昆明人，研究方向为人工智能以及数据挖掘

收稿日期:2014-04-08　　修稿日期:2014-05-06

(下转第 14 页)

Research on the Performance Test of Cluster System Load Balancing

ZHANG Jian-dong, YANG Jin

(Department of Computer Science, Leshan Normal University, Leshan 614004)

Abstract:

Analyzes the key factors affecting the performance of the cluster, and proposes to set up load balancing system under Linux, gets the key performance parameters of the system and calculate the real servers' load. The load balancer dynamically distributes connection requests to real servers based on servers' load to achieve load balancing. Different clients can be used for testing, and uses multiple kinds of testing software to provide support for improving the performance of load balancing.

Keywords:

Cluster; Load Balancing; Performance Test

(上接第 10 页)

The Comparisons Between ID3 and C4.5 Algorithm Based on Decision Tree

MIAO Hong-xing[1], YU Jian-kun[2]

(1. Institute of Computer, Infomation College, Yunnan University of Finance and Economicas, Kunming 650000;
2. Infomation College, Yunnan University of Finance and Economicas, Kunming 650000)

Abstract:

Illustrates the decision tree classifier which is still important in data mining techniques for classifying, and discusses the algorithm of ID3 and C4.5 in decision tree inductive classification, and states that how to choose the decision attributes. Analyzes the algorithm implementation by instance, results that the algorithm of C4.5 has many advantages compared with the algorithm of ID3, especially it is more correct and reasonable on dealing with the data that have many attribute.

Keywords:

Data Mining; Decision Tree; ID3 Algorithm; C4.5 Algorithm