基于考试过程和知识结构的数据挖掘算法研究

代明竹 高嵩峰
(北京建筑大学机电与车辆工程学院 北京 100044)

摘 要 为了研究学生在不同阶段对知识点的掌握情况,基于对数据挖掘的理论研究,把知识结构与考试成绩相结合来进行数据研究。以教育测量学为基础,结合数据挖掘的决策树算法,针对原有的C4.5算法提出改进算法,应用试卷中涉及到的知识点的难易程度与知识点种类进行知识结构细化,以便确定单个学生或群体学生对知识点的掌握程度和试卷中各知识点之间的关系。结果显示,改进后算法的计算公式比原计算公式简单实用;根据决策树模型,使用剩余数据对计算公式进行验证,能够更快地得出对程序设计这个知识点的掌握是影响成绩相对重要的因素。使用测试数据对已创建的决策树进行验证,准确率为90%。最后对决策树进行可视化展示,为学生的学习安排、教师的教学方案及安排等提供有效的参考。

关键词 数据挖掘,决策树,C4.5,知识结构,试卷分析
中图法分类号 TP391 **文献标识码** A

Research on Data Mining Algorithm Based on Examination Process and Knowledge Structure

DAI Ming-zhu GAO Song-feng
(School of Mechanical-electronic and Vehicle Engineering, Beijing University of Civil Engineering and Architecture, Beijing 100044, China)

Abstract In order to study the mastery of knowledge points at different stages of student, based on the theory of data mining, knowledge structure was combined with examination results to study data. Based on the theory of educational measurement and the decision tree algorithm of data mining, an improved algorithm was proposed according to the original C4.5 algorithm, applying the difficulty level of the knowledge points involved in the test papers and the knowledge structure to refine the knowledge structure in order to determine the degree of knowledge of individual students or groups of students and the relationship between the knowledge points. The experimental results show that the efficiency of the improved algorithm is improved, whose formula is simple and practical compared with the original formula. According to the decision tree model, the remaining data is used to verify the improved formula, and it is faster to draw the conclusion that the effect of knowledge points on programming is relatively important. Test data is used to verify the decision tree, and the accuracy rate is 90%. Finally, a visual display of the decision tree can give an effective reference for students to learn the arrangements, teachers to develop teaching programs and arrangements.

Keywords Data mining, Decision tree, C4.5, Knowledge structure, Paper analysis

1 引言

大数据分析时代的到来,给教育领域带来了革命性的变化。大数据分析在试卷分析的基本成绩统计方面已经取得了明显的研究成果,在学生学习中发挥着重要的作用。

另外,由于近些年数据库的相关技术不断突破,数据挖掘技术迅速发展,其主要功能是从种类繁多的庞大数据仓库中获取人们感兴趣的潜在知识[1]。

2005年以来,相关国际会议中对数据挖掘在教育方面的应用开展了多次主题研讨会,促使研究者对数据挖掘在教育领域的应用给予了越来越多的关注[2]。在相关研究文献中对关联规则的使用频率最高,而分类算法中的决策树应用相对较少。刘志妩[3]应用C4.5算法,根据提出的相关属性发现了影响学生成绩的因素,为教师改进教学方法、提高学生成绩提供了帮助。王黎黎[4]通过使用C4.5算法对全国硕士研究生统一招生考试中的英语成绩进行分析,得到了影响成绩的主要规则,有利于指导学生的学习。胡庆[5]根据C4.5创建了知识点与成绩之间关系的决策树,有助于教师和学生了解不同难度的知识点对成绩的影响。但该算法应用到Web程序中时,随着数据集的不断增大,原始的C4.5算法公式的计算速度并不理想。综合以往学者的研究可以知道,C4.5算法是决策树技术中较有代表性的算法,通过对数据进行处理、归纳形成类似树形的模型结果,可以创建出更加精确的结果。但是,C4.5算法的改进与使用方面仍然存在一些不足。

为了检测学生的学习水平和教师的教学质量等,学校会对学生从入学到毕业过程中所学的知识进行大量的测试,从而产生了巨大的数据量。针对C4.5算法中计算公式的效率问题,本文使用改进的C4.5算法对高校考试过程中试卷涉及到的知识结构进行分析,能够更加客观并有效地就学生对

代明竹(1991—),女,硕士生,主要研究方向为管理信息系统;**高嵩峰**(1972—),男,博士,副教授,主要研究方向为管理信息系统、生物力学,E-mail:gaosongfeng@bucea.edu.cn。

知识点的掌握情况进行随时了解并加以测评，从而得到有效且有用的信息。

2 应用算法

2.1 经典测试理论的难易度算法

同一知识点在不同题型中的得分率不同，即掌握这个知识点的难易程度有所差异。这里的难易程度主要是学生得分的高低反映出的试题以及所含知识点的难易度。评价个人或集体对一个知识点的掌握程度，与包含这个知识点的试题是直接联系的，因此将其纳入评价指标中。

主观题的难度系数按如下公式计算：

$$q=\frac{x}{x_n} \tag{1}$$

其中，q 代表难度系数，x 表示一道测试试题的平均得分值，x_n 代表这道题的分值。当试题的平均得分值较高时，表示试题的难度系数较小，反之难度也就较大[5]。

可选客观题的难度系数按如下公式计算：

$$q=\frac{t\times\frac{x}{x_m}-1}{t-1} \tag{2}$$

其中，q 表示难度系数，t 表示选项的数量，x 表示答对试题的人数，x_m 表示被试者的数量[5]。

一般情况下，试卷的难易度系数值在 0.3～0.7 之间，考试过程中的试题难度系数尽量不要超过 0.7 或低于 0.3。

2.2 数据挖掘算法

2.2.1 基本概念

数据挖掘(Data Mining)是发现暗藏的未被人们所发现的知识的行为，是从巨大的、有脏数据的、随机的、不完整的现实数据中获取隐藏的有价值的信息和知识的过程[6]。普遍采用的技术方法主要有分类、聚类、关联规则、回归分析、偏差分析等，它们基于不同的角度与侧重点对数据进行挖掘，其中分类算法主要有决策树、贝叶斯、人工神经网络等[7]。

决策树由主要的节点构成，如根、叶和非叶节点，是从根节点开始进行循环递归操作形成的树形结构。它是一种用于分类的方法，首先分析已经存在的数据来创建多属性之间的模型关系，然后从根节点开始以递归的方式进行创建。本文使用决策树的 C4.5 算法对学生掌握知识点的情况进行分析。

2.2.2 C4.5 算法的原理

1993 年，Ross 对 ID3 算法进行改进，推出了 C4.5 算法，该算法将对信息增益进一步计算为信息增益率，并将其作为衡量变量的标准[8]，是目前最流行的决策树算法。它支持离散属性和连续属性，对未知的属性值和其他可选特征进行处理，并对其进行后剪枝修理。

用 S 表示 s 个数据样本的集合，其中某属性有 n 个类别，类别的集合为 $\{C_1,C_2,\cdots,C_n\}$，用 p 表示一个类别发生的概率。在训练集 S 中随机选择类别，则该类别发生的概率按如下公式计算：

$$p=\frac{freq(C_i,S)}{|S|} \tag{3}$$

应用每一个类别属性的加权平均值得到 S 训练集中的属性分类的期望信息(也称信息熵)，如下：

$$info(s_1,s_2,\cdots,s_n)=-\sum_{i=1}^{n}\left(\frac{freq(C_i,S)}{|S|}\log_2\left(\frac{freq(C_i,S)}{|S|}\right)\right) \tag{4}$$

假设在训练集 S 中，属性 A 有 v 个不同的取值 $\{a_1,a_1,\cdots,a_v\}$，可以把集合 S 划分为 v 个子集 $\{S_1,S_2,\cdots,S_v\}$。如果属性 A 是测试属性，则集合 S 中某一节点产生的分支对应该属性的子集。设 s_{ij} 为子集 S_j 中属于类别 C_i 的样本数量，那么利用属性 A 来划分当前样本集合所需的信息熵[6]，公式如下：

$$E(A)=-\sum_{j=1}^{v}\frac{s_{1j}+\cdots+s_{nj}}{s}info(s_{1j},\cdots,s_{nj}) \tag{5}$$

其中，$\frac{s_{1j}+\cdots+s_{nj}}{s}$ 被视为第 j 个子集的权，并且其值为子集 a_j 中样本的数量和再除以大集合 S 中的样本总量。熵值越小，子集划分的纯度就越高[9]。根据式(4)的期望信息计算方式，对于给定的子集 S_j，其信息熵的计算公式如下：

$$info(s_{1j},s_{2j},\cdots,s_{nj})=-\sum_{i=1}^{n}p_{ij}\log_2(p_{ij}) \tag{6}$$

其中，$p_{ij}=\frac{s_{ij}}{|S_j|}$ 表示子集 S_j 中某一随机属性 C_i 数据样本的概率。

假设属性 A 为目前的一个分支节点，则对应的样本集合划分所得到的信息增益如下：

$$Gain(A)=info(s_1,s_2,\cdots,s_n)-E(A) \tag{7}$$

用属性 A 作为一个基准样本，它的信息增益率如下：

$$GainRatio(A)=\frac{Gain(A)}{SplitInfo(A)} \tag{8}$$

其中，$SplitInfo(A)$ 为分裂信息，表示根据属性 A 对样本集合 S 进行分裂的广度和均匀性，其计算式如下：

$$SplitInfo(A)=-\sum_{j=1}^{v}p_j\log_2(p_j) \tag{9}$$

2.2.3 C4.5 的改进算法

C4.5 拥有产生的分类规则更易于理解且准确率相对较高等优点[10-12]，但因在构造决策树的过程中需要对数据集进行多次扫描和排序，因此算法公式的计算效率尤为重要。由于该算法的信息增益率等公式中涉及对数函数，而对数函数的计算效率较低，因此本文针对算法公式中的对数函数进行改进，以缩短计算时间，提高效率。

基于上述 C4.5 算法原理，设 S_j 中包含的正情况样本数量为 e_j，负情况样本数量为 f_j，因此，子集 S_j 的信息熵为 $info(e_j,f_j)$，使用属性 A 划分当前样本训练集集合所需的信息熵，如下：

$$E(A)=\sum_{j=1}^{v}\frac{e_j+f_j}{e+f}info(e_j,f_j) \tag{10}$$

应用对数函数的换底公式 $\log_a b=\frac{\log_c b}{\log_c a}$，结合属性分类的期望信息的式(10)可以得到：

$$E(A)=\frac{1}{(e+f)\ln 2}\sum_{j=1}^{v}\left(-e_j\ln\frac{e_j}{e_j+f_j}-f_j\ln\frac{f_j}{e_j+f_j}\right) \tag{11}$$

根据泰勒级数和等价无穷小原理，当 x 的值趋近于无穷小时，可以得到 $\ln(1+x)\approx x$ [13]。因此可以得出：

$$\ln\frac{e_j}{e_j+f_j}=\ln\left(1-\frac{f_j}{e_j+f_j}\right)\approx-\frac{f_j}{e_j+f_j}$$

同理：

$$\ln\frac{f_j}{e_j+f_j}=\ln\left(1-\frac{e_j}{e_j+f_j}\right)\approx-\frac{e_j}{e_j+f_j}$$

根据式(11)，改进后的信息熵的计算公式如下：

$$E(A)=\frac{1}{(e+f)\ln 2}\sum_{j=1}^{v}\left(-e_j\ln\frac{e_j}{e_j+f_j}-f_j\ln\frac{f_j}{e_j+f_j}\right)$$
$$\approx\frac{1}{(e+f)\ln 2}\sum_{j=1}^{v}\frac{2e_jf_j}{e_j+f_j} \tag{12}$$

同理，分裂信息公式如下：

$$SplitInfo(A)\approx\sum_{j=1}^{v}\frac{e_j}{e_j+f_j} \tag{13}$$

改进的计算公式中只涉及了简单的加减乘除这种基本运算，比对数函数消耗的运算时间要少很多，因此大幅提高了算法的执行效率。黄秀霞等在对 C4.5 算法的优化中也使用了等价无穷小原理，并引入了 gini 指数的概念[13]，一定程度地提升了计算效率，将该算法的计算公式应用到软件程序开发中时本文中的改进公式的计算效率更高。

应用 C4.5 算法创建决策树并解决问题的主要步骤如下[14-15]：

根据要解决的实际问题选取数据样本集 S，确定相关的属性，构造候选属性集 A。

1)对每一个分类都给出与其名字对应的类编号。

2)读取候选属性集中的属性信息，并将属性信息划分为离散型属性和连续型属性，确定每一个连续型属性的分支阈值；把所有属性放到一个 attributeList 集合中。

3)使用数据集创建决策树，需要输入训练样本 samples、测试样本 testSamples 和候选属性 attributeList。

4)创建根节点 root。

5)若 S 中的元素都属于同一类数据 C，则返回 root 为叶节点并将其标记为 C 类。

6)如果 attributeList＝null 或者 S 中的样本数量小于某给定值，则返回 root 为一个叶节点。

7)循环遍历 attributeList 集合中的每一个属性，并计算每个属性的信息增益率 *GainRatio*。把 testAttribute 记为 root 的测试属性，获取 attributeList 中信息增益率最大的属性[9]。

8)将 root 作为一个新的叶节点继续循环，若该叶节点所对应的数据样本集 S'＝null，则对此叶节点进行分支计算每个分支对应的决策结果；否则，在该叶节点上返回步骤 7)执行递归操作继续分裂，直至给定节点属于同一类、可以进行划分的属性为空或 attributeList 中没有符合条件的 testAttribute 为止。

9)生成决策时，根据树结构从中抽取判断规则。

10)使用 testSamples 数据样本对所抽取的规则进行测试，即测试生成的决策树。

11)根据测试结果，对该决策树进行判定并进一步优化。

3　C4.5 算法在知识结构分析中的应用

3.1　学生成绩数据的获取

本研究数据来源于北京建筑大学 2015 级工业工程专业 42 名学生对《管理信息系统》这本教材第八章“管理信息系统的系统实施”共 25 道测试题多次考试后的最后一次考试成绩。为了对创建完成的决策树进行评估，剔掉其中一条学生缺考成绩和一条成绩不完整的脏数据，取 1/4 的数据记录作为测试信息，3/4 的数据记录作为创建决策树模型的训练集。训练集部分的成绩如表 1 所列。

表 1　学生成绩

学生编号	题号																								总分
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	
1	2	2	0	2	0	2	2	0	0	2	0	0	0	0	0	0	4	2	0	0	8	2	0	2	8	38
2	2	0	2	2	2	0	2	2	0	0	2	4	4	0	4	0	0	2	4	0	8	8	6	2	4	60
3	2	2	2	2	2	2	2	2	2	2	4	4	4	4	4	4	4	4	4	4	8	8	6	8	8	98
4	2	2	0	2	2	0	2	2	2	2	2	4	4	4	4	4	4	4	4	4	6	8	4	0	6	78
5	2	2	2	0	2	2	2	0	2	2	2	4	0	4	4	4	4	4	4	4	8	8	2	4	8	80
6	2	2	2	0	0	2	0	2	2	0	0	4	4	4	4	4	4	2	0	4	4	6	8	0	8	68
…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…
30	2	0	2	2	2	2	2	0	2	0	0	4	4	4	4	0	4	4	4	4	6	0	4	0	8	62

3.2　数据处理

为了更合理地划分学生对试题中所包含知识点的掌握情况，利用经典测试理论对每道试题的难度系数进行计算。试卷中 1－10 题是选择题，为客观题，应使用式(2)进行计算。例如第 1 题中答对的学生有 18 人，则难度系数 q＝(4×18/20－1)/(4－1)≈0.87(取小数点后两位)。11－20 题是填空题，21－25 题是简答题，都为主观题应使用式(1)进行计算。例如第 11 题的满分为 4 分，平均得分为 2.1，则难度系数 q＝2.1/4≈0.5。该文中，使用 JavaWeb 编程对难度系数进行计算，表 2 和表 3 分别给出了主观题和客观题中每题的难度系数，其中第 1 行为试题号，第 2 行为该题的难度系数。

表 2　主观题的难度系数

1	2	3	4	5	6	7	8	9	10
0.87	0.8	0.8	0.8	0.4	0.6	0.73	0.27	0.53	0.33

表 3　客观题的难度系数

11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
0.53	0.75	0.85	0.65	0.8	0.7	0.0.86	0.65	0.75	0.8	0.83	0.6	0.6	0.46	0.93

根据本文中第 2 节提到的经典测试理论，试题的难度系数在 0.3～0.7 范围内的为中，小于 0.3 的为难，大于 0.7 的为易。

为了创建关于知识结构与学生总分数之间相互关系的决策树，首先要获取每题所包含的知识点。根据测试章节的知识点确定决策属性有：物理系统的实施、程序设计、软件开发工具、程序和系统调试以及系统切换、运行及维护，结合每个知识点的难易度将每一个属性类别取值设为 A 易、A 难、B 易、B 中、C 易、C 中、D 易、D 中、E 易、E 中。将知识点与难易度结合后，得出表 4。

表 4 知识点难易度对应的试题号

A 易	A 难	B 易	B 中	C 易	C 中	D 易	D 中	E 易	E 中
12	8	4,13,15,21	6,9,23,24	1,20	16	7,19	10,11,22	2,3,17,25	5,14,18

在表 4 中,表示 A 知识点较容易的有第 12 题,较难的有第 8 题;B 知识点较容易的有第 4,13,15,21 题;其余同理。为了便于了解学生对知识点的掌握程度,在表 4 的基础上结合试题的难易程度,根据难度系数规定其所占权重,易的权重值为 0.3,中的权重值为 0.7,难的权重值为 1,从而计算出不同学生的知识点的部分得分情况,如表 5 所列。

表 5 不同学生对各知识点的得分情况

学生编号	不同知识点成绩				
	A	B	C	D	E
1	0	5.8	0.6	3.4	5.6
2	3.2	11	0.6	8.8	4.6
3	3.2	18	4.6	11.6	11.8
4	3.2	9	4.6	10.2	10.6
5	1.2	10.6	4.6	10.2	11.8
6	3.2	12	4.6	4.2	9
…	…	…	…	…	…
30	1.2	9	1.8	1.8	11.2
满分	3.2	19.4	4.6	11.6	11.8

为了解学生对知识点的掌握程度,根据知识点的得分计算每个知识点的得分率。通过查找各个知识点得分率的个数对决策属性取值进行等级划分,例如全部学生对知识点 A 的得分与该知识点的分值的比为两个不同值,则取值为掌握和未掌握,3 个值为掌握、了解、未掌握,若数值比个数大于 3,则取得分率最大值与最小值用于获取 3 个区间的值如式(14)、式(15)所示,对知识点的掌握程度进行划分。

$$\omega=\frac{\max(course)-\min(course)}{3} \tag{14}$$

其中,ω 为区间宽度,$\max(course)$ 为数据中的最大值,$\min(course)$ 为数据中的最小值;由于把数据划分为 3 个区间,因此除以 3。

$$\begin{aligned} a_i&=\min(course)+(i-1)\times\omega\\ b_i&=a_i+\omega \end{aligned} \tag{15}$$

其中,i 取 1,2,3,为 3 个区间,a_i 为区间的起始值,b_i 为区间的终止值,区间形式为左闭右开,具体区间为($[a_1,b_1)$,$[a_2,b_2)$,$[a_3,b_3)$)。

根据以上公式对所有训练集学生的知识点得分进行掌握度的划分,再对学生试卷的总成绩进行划分。根据学校对成绩的常规划分,进行不同区间的设置,取 4 个区间分别为:优[85,100]、良[75,85)、中[60,75)、差[0,60)。部分学生知识点的具体掌握情况如表 6 所列。

表 6 学生对知识点的掌握情况

A	B	C	D	E	试卷
未掌握	未掌握	未掌握	未掌握	未掌握	差
掌握	了解	未掌握	掌握	未掌握	中
掌握	掌握	掌握	掌握	掌握	优
掌握	未掌握	掌握	掌握	掌握	良
了解	了解	掌握	掌握	掌握	良
掌握	了解	掌握	未掌握	了解	中
…	…	…	…	…	…
了解	未掌握	未掌握	掌握	掌握	中

3.3 构造决策树

表 5 显示了样本训练集的部分数据,包括基于知识结构的 5 个分类;在每个分类中,根据知识点的得分率将知识点属性分为 3 个子集。为了创建知识点与试卷成绩关系的决策树,选取试卷成绩作为类别标识属性,其余各知识点为决策属性。

训练集数据中包含了 30 个元组。为了计算每个决策属性的信息增益率,首先要计算试卷表标识属性试卷的信息熵。元组中试卷类所对应的子集中的元组数分别为:优 $s_1=8$,良 $s_2=4$,中 $s_3=12$,差 $s_4=6$。使用未改进算法计算试卷的信息熵:

$$\begin{aligned} info(s_1,s_2,s_3,s_4)=&-\frac{8}{30}\log_2\frac{8}{30}-\frac{4}{30}\log_2\frac{4}{30}-\\ &\frac{12}{30}\log_2\frac{12}{30}-\frac{6}{30}\log_2\frac{6}{30}\\ =&1.8797 \end{aligned}$$

对于标识属性试卷来说,对其他决策属性进行计算。属性 A 类别为未掌握的有 7 个样本,且试卷为优(s_{11})的有 1 人,良(s_{21})有 0 人,中(s_{31})有 3 人,差(s_{41})有 3 人;类别为了解的有 9 个样本,且试卷为优(s_{12})的有 0 人,良(s_{22})有 3 人,中(s_{32})有 4 人,差(s_{42})有 2 人;类别为掌握的有 14 个样本,且试卷为优(s_{13})的有 7 人,良(s_{23})1 人,中(s_{33})5 人,差(s_{43})1 人。根据改进式(6)和式(10),可以直接得出 A 为决策属性的条件熵和分裂熵为:

$$\begin{aligned} E(A)=&\frac{7}{30}I(s_{11},s_{21},s_{31},s_{41})+\frac{9}{30}I(s_{12},s_{22},s_{32},s_{42})+\\ &\frac{14}{30}I(s_{13},s_{23},s_{33},s_{43})\\ =&1.7145 \end{aligned}$$

$$\begin{aligned} SplittInfo(A)=&-\frac{7}{30}\log_2\frac{7}{30}-\frac{9}{30}\log_2\frac{9}{30}-\frac{14}{30}\log_2\frac{14}{30}\\ =&1.5099 \end{aligned}$$

根据式(7)和式(8)计算属性 A 的信息增益率为:

$$Gain(A)=1.8797-1.7145=0.1652$$

$$GainRatio(A)=\frac{0.1652}{1.5099}=0.1094$$

使用同样的方法依次计算出其他决策属性的信息增益率分别为:

$$GainRatio(B)=\frac{1.07}{1.5999}=0.6688$$

$$GainRatio(C)=\frac{0.2702}{0.8726}=0.3096$$

$$GainRatio(D)=\frac{0.6199}{1.4125}=0.4389$$

$$GainRatio(E)=\frac{0.3483}{1.5220}=0.2288$$

根据以上计算结果可以得出:$GainRatio(B)>GainRatio(D)>GainRatio(C)>GainRatio(E)>GainRatio(A)$。由此可知,决策属性 B(程序设计)的信息增益率最大,则该属性被选为决策树的根节点;再根据该属性有 3 个取值,因此从该节点可以分裂出掌握、了解、未掌握 3 个分支。为避免不必要的分裂或产生多余的分支,分别计算属性 B 中每个值占试卷各取值的比例,用 Q 表示,结果如下:

Q(未掌握,差)=6/9=0.667;Q(了解,中)=9/12=0.75;Q(掌握,优)=8/9=0.889。

其中，属性 B 取值为掌握的情况且试卷成绩为优的概率为 88.9%，因为我们设定的估计准确率为 80%以上时不用继续分裂，则取值为掌握的分支符合要求；其余两条分支不符合要求，因此需要继续生成各自的子树。

对属性 B 取值为了解的分支，用改进算法来计算其余决策属性的信息增益率。使用式(12)和式(13)来计算属性 A 的条件熵和分裂熵，由于改进公式中 $1/[(e+f)\ln 2]$ 在计算时为常数，因此省去，不参与计算，结果如下：

$$E'(A)=\frac{0\times0\times0\times3}{0+0+0+3}+\frac{1\times2\times2\times0}{1+2+2+0}+\frac{0\times0\times4\times0}{0+0+4+0}=0$$

$$SplitInfo'(A)=\frac{3\times5\times4}{3+5+4}=5$$

$$GainRatio'(A)=\frac{5-0}{5}=1$$

在此子树中，继续使用改进公式计算其他属性的信息增益率，$GainRatio'(C)=0.4375$，$GainRatio'(D)=0.8333$，$GainRatio'(E)=0.84$。因为属性 A 的信息增益率最大，所以选择 A 为节点，根据计算公式可以得出：当 A 取值为未掌握和试卷为中的估计概率均为 100%时不需要继续分支，只有取值为了解时需要继续分支。然后按照同样的方法对其他节点进行递归计算并分析和剪枝，可以得到学生对知识结构的掌握与试卷成绩的关系决策树，如图 1 所示。

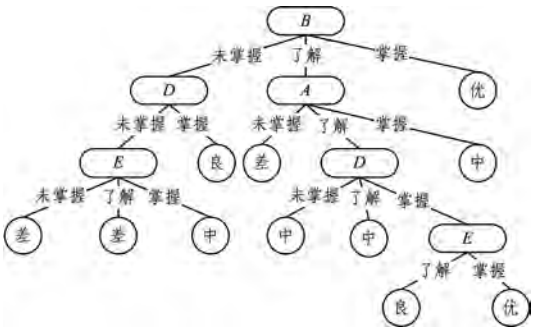

图 1　学生对知识结构的掌握与成绩的关系决策树

利用测试集中与训练集中具有相同测试环境的学生的成绩对该决策树结果进行测试。在测试集中共有 10 人，其中与决策树分类结果相符的有 9 人，准确率为 90%，则该准确率有效。

3.4　实验结果

根据图 1 中各个分支的情况，能够得出试卷成绩的优与差主要由程序设计(B)、程序和系统调试(D)、物理系统的实施(A)和系统切换、运行及维护(E)这几个知识点决定，与软件开发工具(C)是否掌握没有直接关系。其中，学生对程序设计这个知识点的掌握情况对试卷成绩好坏最为重要，若对程序设计这个知识点没有掌握，则要把握对程序和系统调试知识点的学习。在对程序设计和物理系统的实施有一定了解的基础上，要加强对程序和系统调试与系统切换、运行及维护这两个知识点的掌握。

结束语　本文在论述了决策树 C4.5 算法的基础上，对其进行了计算公式效率上的改进，并采用决策树技术对学生掌握知识结构与试卷成绩之间的关系进行了分析。在创建决策树的过程中选择了试卷中涉及到的相关知识点作为决策属性，使用未改进与改进的 C4.5 算法公式进行对比计算。最终实验结果表明，在《管理信息系统》这本教材的第八章知识结构中程序设计这一知识点对该章学习的影响最大。不足之处在于，该决策树模型的决策属性考虑得不够细致，只是按照章中小节划分，而且没有考虑试题是否有包含多个知识点的情况；训练集与测试集的样本数量都较少；对于决策树的后剪枝技术需要更深入的学习。未来可以对知识结构进行完善，并把改进后的算法应用到 Web 开发中，使结果能更快速、清晰地进行展示，达到信息的真正可视化。

参 考 文 献

[1] 白彦辉. 关联规则挖掘在试卷分析系统中的应用[J]. 内蒙古民族大学学报(自然科学版)，2012，27(2)：159-161.

[2] 牛瑞敏. 数据挖掘在国内教育领域应用的研究综述[J]. 中山大学研究生学刊(人文社会科学版)，2016，37(2)：193-200.

[3] 刘志妩. 基于决策树算法的学生成绩的预测分析[J]. 计算机应用与软件，2012，29(11)：312-314，330.

[4] 王黎黎，刘学军. 决策树 C4.5 算法在成绩分析中的应用[J]. 河南工程学院学报(自然科学版)，2014，26(4)：69-73.

[5] 胡庆. 基于决策树的试卷知识点掌握程度分析研究[D]. 南昌：江西财经大学，2014.

[6] 段薇，马丽，路向阳. 基于信息增益和最小距离分类的决策树改进算法[J]. 科学技术与工程，2013，13(6)：1643-16552.

[7] NOH C H，CHO K C，MA Y B，et al. Grid resource selection system using decision tree method [J]. Korea Soc Comput Inf，2009，13(1)：1-10.

[8] 阮晓宏，黄小猛，袁鼎荣，等. 基于异构代价敏感决策树的分类器算法[J]. 计算机科学，2013，40(11)：140-142，146.

[9] 毛国君，段立娟. 数据挖掘原理与算法[M]. 北京：清华大学出版社，2016：128-137.

[10] 于孝美，陈贞翔，彭立志. 基于决策树的网络流量分类方法[J]. 济南大学学报(自然科学版)，2012，26(3)：291-295.

[11] 王领，胡扬. 基于 C4.5 决策树的股票数据挖掘[J]. 计算机与现代化，2015(10)：20-24.

[12] JIANG W L. Research and Application of Credit Score Based on Decision Tree Model[M]// Applied Informatics and Communication. Springer Berlin Heidelberg，2011：493-501.

[13] 黄秀霞，孙力. C4.5 算法的优化[J]. 计算机工程与设计，2016，67(5)：1267-1271.

[14] 宋万洋，李国和，洪云峰，等. 基于平衡准确率和规模的决策树剪枝算法[J]. 科学技术与工程，2016，16(16)：79-82.

[15] KANTARDZIE M. Data mining：concepts models methods and algorithms[M]. John Wiley & Sons，Inc.，2004.