第19卷第2期 广西科学院学报 Vol. 19, No. 2
2003年5月 Journal of Guangxi Academy of Sciences May 2003

判定树归纳分类法在毕业生就业预测中的应用
Application of the Decision Tree Induce Classification in the Forecast of Employment of Graduates

聂永红
Nie Yonghong

(广西工学院计算机工程系 柳州 545006)
(Dept. of Comp. Engi., Guangxi Institute of Tech., Liuzhou, 545006)

摘要 采用数据挖掘中的判定树归纳分类法预测毕业生就业情况，给出预测模型、数据采集过程和相应的实现算法及判定树的算法，对判定树归纳分类法进行准确性评估，并给出一个实例。该预测可以用来统计历届毕业生就业情况和指导下届毕业生就业。
关键词 毕业生 就业 判定树归纳分类法 数据挖掘 预测
中图法分类号 TP311.132.3
Abstract The basic process of the decision tree induce classification of data mining calculation is introduced to forecast the employment situations of graduates. The related arithmetics of realization and decision tree induce classification are given. The accuracy of the decision tree induce classification is evaluated with a sample. It can be used for counting the employment situations of the preterit graduates, and guiding the employment works of the graduates upcoming.
Key words graduate, employment, decision tree induce classification, data mining, forecast

随着数据库技术的迅速发展以及数据库管理系统的广泛应用，人们积累的数据越来越多，在大量的数据背后隐藏着许多重要的信息。只有拥有了先进的数据库技术，才能有效地管理好浩如烟海的数据，并从中提取出对自己有用的信息。数据挖掘是从数据库或数据仓库中发现并提取隐藏在其中的信息的一种新技术，它建立在数据库、数据仓库之上，面向非专业用户，定位于桌面，支持即兴的随机查询。数据挖掘技术能自动分析数据，对它们进行归纳性推理和联想，寻找数据间某些内在的关联，从中发掘出潜在的、对信息预测和决策行为起着十分重要作用的模式，从而建立新的模型，为人们的正确决策提供了很大的帮助。本文就毕业生的就业情况，采用数据挖掘中的判定树归纳分类法进行预测，并给出相应的实现算法。

1 数据采集

1.1 数据挖掘的模型设计

数据挖掘算法的工作方法是通过分析已知分类信息的数据总结出一个预测模型。用于建

2002-10-14 收稿。

立模型的数据称为训练集，用于测试所建模型的准确率称为测试集，通常是已经掌握的数据。训练和测试数据挖掘模型把已知数据分成两部分：一个用于模型训练，占整个数据的 2/3；另一个用于模型测试，占整个数据的 1/3。注意一定要保证数据选择的随机性，这样才能使分开的两部分数据的性质是一致的。用训练集把模型建立出来之后，可以先在测试集数据上进行测试，此模型在测试集上的预测准确率就是一个很好的指导数字，若准确率大于 90%，则说明在该训练集上所建立的模型是可行的，可用此模型预测其它的数据，并给出其预测结果为正确的百分比。

1.2　数据采集过程

关系数据库是数据挖掘最丰富、最流行的数据源，它是数据挖掘的主要数据形式，它是表的集合，每个表有唯一的名字，包含一组属性（字段或列），存放大量元组（记录或行）。关系中的每个元组代表一个被唯一标识的对象——关键字，并被一组属性描述[1]。可用关系数据库构造 E-R 模型。如：某校计算机专业有 280 名毕业生，对其在毕业 2 个月内是否就业的情况进行抽样调查，按学号、性别、综合成绩、毕业论文、党员、学生干部和就业情况等 7 种属性进行数据收集，形成应届毕业生就业情况分析表，即分类预测所需的关系数据库，我们将获取的部分数据分为两部分，一部分是应届毕业生情况数据库训练数据元组（即训练集），见表 1，占收集数据的 2/3，它的功能是完成根据已有的数据对各项指标的分类计算并建立相应的分类模型；另一部分是应届毕业生测试数据库测试数据元组（即测试集），占收集数据的 1/3，它的功能是完成对毕业生就业情况分类预测结果的评估。

表 1　应届毕业生 2 月内就业情况

学号	性别	综合成绩	毕业论文	党员	学生干部	就业情况
971002	女	70～79	良	是	是	已
971003	男	70～79	中	否	否	未
971006	女	80～89	良	否	否	未
971018	男	60～69	及格	否	否	未
971022	男	70～79	良	否	是	已
971037	女	80～89	中	否	否	未
971045	女	80～89	优	否	是	已
971051	男	80～89	良	是	否	已
971062	女	90 以上	良	否	否	已
971081	男	90 以上	良	否	否	已
971093	女	60～69	良	否	是	未
971113	男	80～89	中	否	否	已
971125	男	70～79	及格	否	否	已
971133	男	70～79	中	否	否	未
971235	男	80～89	中	是	否	未
971246	女	80～89	中	否	是	已
971276	男	70～79	良	否	否	已
971284	男	90 以上	中	是	是	已
971295	男	70～79	良	否	否	已
971298	男	70～79	中	是	否	已

表 1（stud 表）有一组属性，包括毕业生的唯一标号（学号）、性别、综合成绩、毕业论文、党员、学生干部和就业情况等，我们通过对该表现有数据进行分析，得到所需分类规则，通过另外 1/3 的数据对所建模型进行正确性评估后（正确率为 90%以上），就可以预测另外的毕业生就业情况，现介绍用判定树归纳分类法预测就业情况及其算法实现。

2 数据分类模型

2.1 判定树归纳

判定树归纳的基本算法是贪心算法，它以自顶向下递归的各个击破方式构造判定树，算法的基本策略[1,2]如下：

①判定树以代表训练样本的单个节点开始；

②如果样本都在同一个类，则该节点成为树叶，并用该类标记；

③否则，算法使用称为增益的基于熵的度量作为启发信息，选择能够最好地将样本分类的属性。该属性成为该节点的“测试”或“判定”属性；

④对测试属性的每个已知的值，创建一个分枝，并以此为根据划分样本；

⑤使用同样的过程，递归地形成每个划分上的样本判定树；

⑥递归划分步骤仅当下列条件之一成立时停止：

a. 给定节点的所有样本属于同一类；

b. 没有剩余属性可以用来进一步划分样本，在此情况下，使用多数表决。这涉及将给定的节点转换成树叶，并用训练集中的多数所在的类标记它；

c. 分枝没有样本，此时，以训练集中的多数类创建一个树叶。

2.2 属性选择度量

在树的每个节点上使用信息增益度量选择测试属性[1,3]，选择具有最高信息增益（或最大熵压缩）的属性作为当前节点的测试属性，该属性使得对结果划分中的样本分类所需的信息量最小，并反映划分的最小随机性。这种信息理论方法使得对一个对象分类所需的期望测试数目达到最小，并确保找到一棵简单的树。

设S是s个数据样本的集合。假定类标号$C_i(i=1,\cdots,m)$具有m个不同值，设s_i是类C_i中的样本数。对一个给定的样本分类所需的期望信息由下式给出：

$$I(s_1,\cdots,s_m)=-\sum_{i=1}^{m}p_i\log_2(p_i), \tag{1}$$

其中p_i是任意样本属于C_i的概率，并用s_i/s估计。

设属性A具有v个不同值$\{a_1,\cdots,a_v\}$。可以用属性A将S划分为v个子集$S_1,\cdots,S_v$；其中，S_j包含S中这样一些样本，它们在A上具有值a_j。如果A选作测试属性，则这些子集对应于由包含集合S的节点生长出来的分枝。设s_{ij}是子集S_j中类C_i的样本数。根据由A划分成子集的熵由下式给出：

$$E(A)=\sum_{i=1}^{m}\frac{(s_{1j},\cdots,s_{mj})}{s}I(s_{1j},\cdots,s_{mj}), \tag{2}$$

熵值越小，子集划分的纯度越高。对于给定的子集S_j，

$$I(s_{1j},\cdots,s_{mj})=-\sum_{i=1}^{m}p_{ij}\log_2(p_{ij}), \tag{3}$$

其中，$p_{ij}=\frac{s_{ij}}{|S_j|}$是$S_j$中的样本属于类$C_i$的概率。

在A上分枝将获得的编码信息是

$$Gain(A)=I(s_1,\cdots,s_m)-E(A), \tag{4}$$

计算每个属性的信息增益。具有最高信息增益的属性选作给定集合S的测试属性。创建一

个节点，并以该属性标记，对属性的每个值创建分枝，并据此划分样本。

2.3　具体实施过程

表 1 是应届毕业生数据库数据元组训练集，类标号属性“就业情况”有 2 个不同的值（“已”，“未”），因此有 2 个不同的类。设类 C_1 对应于“已”，类 C_2 对应于“未”。类 C_1 有 13 个样本，类 C_2 有 7 个样本。计算对给定样本分类所需的期望信息：

$$I(s_1,s_2)=I(13,7)=-\frac{13}{20}\log_2\frac{13}{20}-\frac{7}{20}\log_2\frac{7}{20}=0.934\ 068\ 054.$$

接着，需要计算每个属性的熵。我们先计算属性为“综合成绩”的熵。观察综合成绩的每个样本值为“已”和“未”的分布，对每个分布计算期望信息。

对于综合成绩 =“60 ～ 69”，$s_{11}=0$，$s_{21}=2$，由(3) 式得：$I(s_{11},s_{21})=0$；

对于综合成绩 =“70 ～ 79”，$s_{12}=6$，$s_{22}=2$，由(3) 式得：

$I(s_{12},s_{22})=0.811\ 278\ 124$；

对于综合成绩 =“80 ～ 89”，$s_{13}=4$，$s_{23}=4$，由(3) 式得：

$I(s_{13},s_{23})=0.985\ 228\ 136$；

对于综合成绩 =“90 以上”，$s_{14}=3$，$s_{24}=0$，由(3) 式得：$I(s_{14},s_{24})=0$.

由(2) 式，如果样本按“综合成绩”划分，对一个给定样本分类所需的熵为：

$$E(\text{综合成绩})=\frac{2}{20}I(s_{11},s_{21})+\frac{8}{20}I(s_{12},s_{22})+\frac{7}{20}I(s_{13},s_{23})+\frac{3}{20}I(s_{14},s_{24})$$
$$=0.669\ 341\ 096.$$

由(4) 式，可求出这种划分的信息增益是：

$Gain(\text{综合成绩})=I(s_1,s_2)-E(\text{综合成绩})=0.264\ 726\ 957$，

类似地，可计算出：

$Gain(\text{性别})=I(s_1,s_2)-E(\text{性别})=0.177\ 361\ 421\ 2$，

$Gain(\text{毕业论文})=I(s_1,s_2)-E(\text{毕业论文})=0.090\ 176\ 027$，

$Gain(\text{党员})=I(s_1,s_2)-E(\text{党员})=0.124\ 982\ 085$，

$Gain(\text{学生干部})=I(s_1,s_2)-E(\text{学生干部})=0.166\ 723\ 636$.

由计算结果，可知“综合成绩”在属性中具有最高信息增益，它被选作测试属性。创建一个节点，用综合成绩标记，并对于每个属性值，引出一个分枝。样本按此划分。对每个分枝，再用判定树归纳分类法进行分类，再引出分枝，最后，算法返回的最终判定树如图 1 所示。

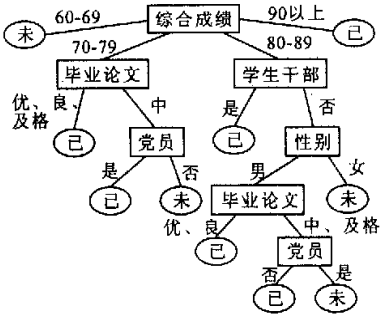

图 1　应届毕业生就业情况判定树

注：每个内部节点表示一个属性上的测试，每个树叶节点代表一个类（就业情况 =“已”，就业情况 =“未”）。

3　预测毕业生就业情况

利用上述所构造的判定树归纳分类法对测试集数据进行测试，可知其准确率为 90% 以上。因此，利用该判定树归纳分类法可对未知样本 X 进行分类预测，若分类预测准确率为 90% 以上，说明对所选测试集而建立的分类模型是可行

的。

对于已给的任意样本 X，只要知道其性别、综合成绩、毕业论文、党员及学生干部，就可预测其是否就业。

例 1 $X=\{971057,$"女","70～79","良","否","是"$\}$.

由图 1，可知其就业情况为"已"。

该预测模型在学生管理工作统计历届毕业生的近期就业情况，及对下届毕业生就业指导等方面有着现实的意义。

参考文献

1 Jiawei H, Kamber M. 数据挖掘概念与技术. 范 明，孟小峰译. 北京：机械工业出版社，2001. 185～220.
2 Matheus C J, Piatetsky-Shapiro G, McNeil D. Selecting and reporting what is interesting: The KEFIR application to healthcare data. In: Fayyak U M, Piatetsky-Shapiro G, Smyth P, Uthurusamy R. Advances in Knowledge Discovery and Data Mining, Cambridge, MA: AAAI/MIT Press, 1996. 495～516.
3 盛 骤，谢式千，潘承毅编. 概率论与数理统计. 北京：高等教育出版社，1999.

（责任编辑：黎贞崇）

（上接第 71 页）

(d)如果第(c)步递归调用算法后返回的是失败标志，则退出算法并返回失败标志，否则递归调用算法 CreateBTree 生成 T 的右子树并返回标志。该层递归算法结束。递归调用的参数：$T=T$ 的右子树指针，prestr _ startpos＝prestr _ startpos＋1＋(pos-midstr _ startpos)，midstr _ startpos＝pos＋1，N＝midstr _ startpos＋N－1－pos。（即划分前序遍历子序列 prestr[prestr _ startpos＋1＋(pos-midstr _ startpos). . prestr _ startpos＋N－1]为根 T 的右子树的前序遍历序列，而划分中序遍历子序列 midstr[pos＋1. . midstr _ startpos＋N－1]为根 T 的右子树的中序遍历序列，然后由这 2 个划分出的序列递归生成 T 的右子树）

3 结束语

本文改进算法只适用于结点数据为单个字符的二叉树，若希望生成任意数据类型(包括记录类型)结点的二叉树，则可以先对每个结点赋予一个惟一的 ID 号，由该 ID 号代替结点数据，然后选择一个好的散列函数对该 ID 号进行地址散列，也可在最差情况下达到 $O(N)$时间复杂度。

参考文献

1 Corman T H, Leisern C E. Introduction to algorithms. The MIT Press, 1995.
2 Pieprzyh J, Sadeghiyan. Design B of hashing algorithms. Berlin: Springer-verlag, 1993.
3 Baase A, Gelder A V. Computer algorithms: Introduction to design and analysis. Higher Education Press, 2001.
4 娄定俊. 算法分析与设计. 广州：中山大学出版社，1998.
5 严蔚敏，吴伟民. 数据结构(C 语言版). 北京：清华大学出版社，1997.
6 克努特 D E. 计算机程序设计技巧(第一卷 基本算法). 管纪文，苏运霖译. 北京：国防工业出版社，1980. 277.

（责任编辑：黎贞崇）