基于JEP平均长度的分类方法

于大东[1] 刘东波[2,3] 罗 睿[3] 王建新[1,3]

(国防科学技术大学电子科学与工程学院 长沙 410073)[1]

(华中科技大学计算机科学与技术学院 武汉 430000)[2] (中国电子设备工程公司 北京 100039)[3]

摘 要 本文研究了JEP——一种在不同数据集之间支持度从零到非零跳跃性变化的项集在数据分类中存在的问题,提出了项集独立支持度的概念。相对于传统的项集支持度来说,独立支持度能够更加全面地描述数据的分布特征,为更加准确的分类提供依据。进而,在独立支持度的基础上提出了JEP平均长度的概念,并提出了一种以测试样本所覆盖JEP的平均长度作为分类特征的分类方法,该方法可以更加有效地区分类边界上的数据,能够为数据提供更为准确的分类。

关键词 项集,关联规则,跳跃式显露模式,分类

Classification Based on Average Length of JEPs

YU Da-Dong[1] LIU Dong-Bo[2,3] LUO Rui[3] WANG Jian-Xin[1,3]

(School of Electronic Science & Engineering, National University of Defense Technology, Changsha 410073)[1]

(College of Computer Science & Technology, Huazhong University of Science & Technology, Wuhan 430000)[2]

(Institute of China Electronic System Engineering, Beijing100039)[3]

Abstract JEP is a kind of itemset which support in one dataset is zero but in another dataset not zero. After studying the problems of JEPs in the classification procedure, we presents the concept of independent supports of itemsets. Compared with traditional supports, independent supports provide more detailed distribution information of the dataset and more powerful classifiers can be built on them. Then we propose the concept of average length based on independent supports of JEPs. Finally, we present a classification algorithm using the average length of JEPs as the classification feature. This algorithm can give a more precise classification to data on the border of the datasets.

Keywords Itemset, Association rules, Jumping emerging patterns, Classification

1 引言

近几十年来,由于信息存储技术和计算机处理能力的长足进步,计算机信息管理系统在各行各业得到广泛应用,并积累了大量的数据。随着数据资源的不断积累,如何有效地分析和利用这些数据,为决策人员提供有意义的关于这些数据的知识成为人们面临的新课题。于是,数据挖掘应运而生。

数据挖掘是一种从大量数据中获取潜在的、有用的、新颖的、有利于决策的知识的技术,它的研究方向主要包括特征概念描述、关联规则挖掘、分类聚类、孤立点分析和演变分析等。关于分类方法的研究是一个重要的研究课题,它所采用的技术包括判定树归纳、贝叶斯分类、k最近邻分类、贝叶斯网络、神经网络、模糊逻辑和粗糙集等[1]。这些技术的共同特点是,它们都要首先建立一个分类模型,然后通过训练集对该模型进行训练。它们都面临着一个严峻的问题,即分类模型对训练数据的过度适应问题,这是由这些方法的本质所决定的。关联规则挖掘[1,3,4]通常用来发现数据库中普遍存在的关联关系,它也可以用来发现数据之间普遍存在的相似之处,因此也可用于分类。Liu、Hsu和Ma[5]以及Li和Dong[6,7]提出了基于关联规则进行分类的思想,这两种方法都是首先从数据的总体特征——关联规则着手来建立分类规则,在一定程度上克服了上述方法对于训练数据过度适应的问题,同时还具有较高的分类准确率,其分类准确率高于Quinlan的C4.5分类算法[5]。

跳跃式显现模式(Jumping Emerging Patterns,缩写为JEP)是一种特殊的关联规则,它是在一类数据中支持度为零,而在另一类数据中支持度非零的特殊关联模式,它对于异类数据具有很强的区分能力[6]。Li等人提出的JEP分类方法首先发现两个数据集中所有的JEP,然后利用那些对分类贡献最大的JEP来构建分类器,分类器计算测试数据中包含的JEP,分别计算两个类的综合影响因子并进行比较,最后给出划分结果。然而,这种JEP分类方法在影响分类结果的综合影响因子的计算上,仅仅考虑了测试数据所覆盖的JEP的支持度,没有考虑到这些JEP覆盖了多少训练数据,也没有考虑这些JEP的相对长度对分类的影响,下面的例子说明了这一点。

例1 两个简单的数据集如表1所示,从A到B的所有JEP为$\{a,b,ab\}$,而从B到A的所有JEP为$\{c,d,ce,cd,de,cde\}$,这两个JEP集合都可以以项集边界[7]的方式表达:$Border_{AB}=\langle\{a,b\},\{ab\}\rangle$和$Border_{BA}=\langle\{c,d\},\{cde\}\rangle$。根据Li的分类方法,利用JEP集合的左边界计算数据样本分类的综合影响因子(Collective Impact Factor,缩写为Cif),测试数据将被划

于大东 博士研究生,主要研究方向为数据挖掘、数据融合等。刘东波 教授,主要研究方向为模糊逻辑、模糊数据库、数据挖掘和数据融合等。罗 睿 博士,主要研究方向为数据挖掘、数据融合等。王建新 研究员,博士生导师,主要研究方向为软件体系结构、数据挖掘和数据融合等。

分给综合影响因子最大的那一类。对于 abcde 这样的测试数据来说，它们的综合影响因子 $Cif_{AB}=supp_A(a)+supp_A(b)=1.6>Cif_{BA}=supp_B(c)+supp_B(d)=1.2$，因此这个测试数据将被划分给 A 类数据集，显然这种分类是错误的。

表 1 两类项集数据

ID	数据集 A	数据集 B
1	ab	ce
2	ab	cde
3	ab	de
4	ab	de
5	e	de

从这个示例中我们可以看出 Li 的 JEP 分类算法(以下简称 JEPC)主要存在以下不足：

1. JEPC 没有考虑用来计算综合影响因子的 JEP 的长度对分类的影响。我们认为在同等支持度的情况下，长度大的 JEP 对于分类的贡献要大于长度小的 JEP。

2. 覆盖率越大的 JEP 越能够反映数据集的整体特征，然而，JEPC 在使用 JEP 计算分类的综合影响因子时没有考虑 JEP 的覆盖率对分类的影响。

3. 只使用 JEP 边界来构建分类器，在很多情况下不能很好地反映数据样本实际的类属情况，容易造成错误的分类结果。

本文的内容是这样安排的，第 2 节给出 JEP 分类问题的形式化描述；第 3 节介绍基于 JEP 平均长度的分类算法；第 4 节介绍相关的研究工作；最后对本文进行总结。

2 问题描述

假设训练数据集 D 是一个标准的关系表，共有 m 个互不相同的属性，表中共有 N 个记录分别属于 q 个不同的类。根据记录所属类别的不同，数据集 D 被划分为 q 个子数据集：$D_1,D_2,\cdots,D_q$。我们的目标是通过对训练数据集 D 的学习，建立一个分类器，使它能够对类似 D 中数据的样本 T 进行正确分类。数据样本的属性既可以是离散的，也可以是连续的。对于离散的属性，我们直接将其各种可能的取值映射为不同的项，而对于连续的属性，我们首先采用等间隔方式将其取值区间划分为互不相交的若干子区间，将连续的属性值映射到这些区间，从而将连续的属性离散化为项。设表中所有项的集合为 I，数据集中所有的记录都是若干个项的集合，简称为项集。

定义 1[6] 数据集 D_i 到 D_j 的 JEP 是指那些在 D_i 中支持度非零，而在 D_j 中支持度为零的项集，它们的集合记为 $JEP(D_i,D_j)$。类似地，数据集 D_j 到 D_i 的 JEP 是指那些在 D_j 中支持度非零，而在 D_i 中支持度为零的项集，它们的集合记为 $JEP(D_j,D_i)$。

例 2 对于表 1 中的数据，通过计算各个项集出现的次数，我们可以发现：$JEP(A,B)$ 为 $\{a,b,ab\}$，而 JEP 为 $\{c,d,cd,de,ce,cde\}$。

因为 JEP 是不同类别之间特有的数据特征，所以它可以作为分类的重要因素。对于两个数据集的所有 JEP 来说，每个 JEP 所反映类的差别的能力也是不同的，它们受另外两个因素的影响：JEP 的支持度和 JEP 所包含项的多少。同等条件下，支持度越大、包含项数越多的 JEP 所能反映类之间差别的能力就越大。例如表 1 中所示数据，对于测试数据 abc 而言，ab 较 a 和 b 更能说明 abc 是属于 A 类而不是 B 类。

分类问题的本质就是比较测试样本和训练样本集之间的类似程度，进而作出分类预测。换句话说，分类问题就是比较测试样本有多少训练样本所具有的特征。因此，对于上述问题，我们自然需要考虑测试数据覆盖了多少训练样本所共同包含的 JEP。为此，我们给出下面的 JEP 集对于数据集的覆盖度和 JEP 集中 JEP 平均长度的定义。

定义 2 集合 $J=\{J_1,J_2,\cdots,J_n\}$，其中 $J_i(1\leqslant i\leqslant n)$ 是数据集 D_1 到 D_2 的 JEP，$J\subseteq JEP(D_1,D_2)$。J 对于数据集 D_1 的覆盖度是指 D_1 中包含这些 J_i 的所有数据占全体数据的百分比，记为 $Cover_{D_1}(J)$ 或者 $Cover_{D_1}(\{J_1,J_2,\cdots,J_n\})$。对于表 1 中数据集 B 而言，$Cover_B(ce,de\})=100\%$。

定义 3 项集集合 $I=\{I_1,I_2,\cdots,I_m\}$，且 I 满足以下条件：$\forall I_i,I_j\in I$，都有 $I_i\cup I_j\in I$ 或者 $support(I_i\cup I_j)=0$ 成立。I 中项集 $I_j(1\leqslant j\leqslant m)$ 的独立支持度是指那些包含 I_j 且不包含任何 $I_k(k\neq j,I_k\subset I_j)$ 的数据在数据集中所占的百分比，I_j 的独立支持度记为 $supp_{in}(I_j)$。

项集独立支持度的概念是相对于一个项集集合来讲的，同一个项集在不同的项集集合中独立支持度可以是不同的。它在计数方法上有别于传统的项集支持度：它消除了传统支持度的计数方法对于包含的项集重复计数，不利于更加有效地反映数据分布的特点。对于每一个参与计数的数据，仅仅对它所包含的项集集合内最大的项集计数，而忽略其所有子集的计数。例如，表 1 中项集集合 $\{de,cde\}$ 的支持度为 $\{80\%,20\%\}$，而其独立支持度则为 $\{60\%,20\%\}$。

有了项集独立支持度，我们可以方便地得到项集对于数据集的覆盖度。下面关于 JEP 的平均长度的定义也基于独立支持度这个概念。独立支持度可以由项集支持度直接计算得到，下一节我们将给出它的计算方法。

定义 4 集合 $J=\{J_1,J_2,\cdots,J_n\}$，其中 $J_i(1\leqslant i\leqslant n)$ 是数据集 D_1 到 D_2 的 JEP，$J\subseteq JEP(D_1,D_2)$。J 中 JEP 的平均长度是指各个 J_i 长度乘以其独立支持度的加权平均，记为 $\overline{Length}(J)$。

$$\overline{Length}(J)=\sum_{i=1}^{n}supp_{in}(J_i)*Length(J_i) \quad (1)$$

3 基于 JEP 平均长度的分类方法

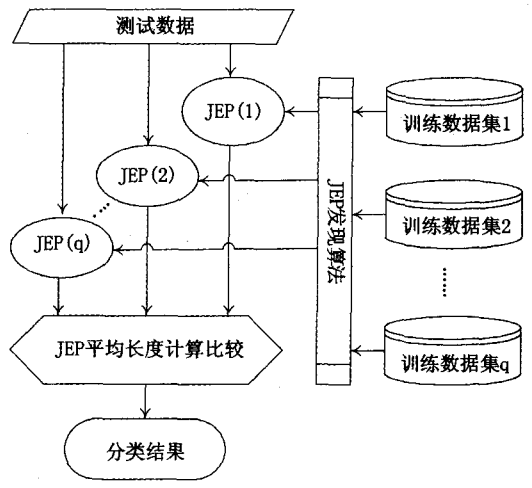

图 1 基于 JEP 平均长度的分类流程

相对于 JEP 边界来说，数据样本中所包含 JEP 的平均长度能够更好地反映该数据样本的类属特性，因此我们采用

JEP的平均长度作为分类的评价标准，通过比较数据样本所覆盖的JEP的平均长度来构建分类器，分类的流程如图1所示。

3.1 JEP发现算法

由JEP的定义可知，最为直接的JEP发现方法就是首先分别发现两个数据集的所有项集及其相应的支持度，然后通过逐项对比找出所有的JEP，显然这种方法的时间开销很大。Dong和Li[7]给出了一种基于JEP边界的发现算法，极大地提高了JEP的发现效率，这里简单描述如下：

JEP的边界是一个序偶$\langle \mathcal{L}, \mathcal{R}\rangle$，其中$\mathcal{L}$和$\mathcal{R}$都是项集的集合，分别称为左边界和右边界，并且其中的各个元素两两之间互不包含。$\mathcal{L}$中的每个元素都是$\mathcal{R}$中某个元素的子集，$\mathcal{R}$中每个元素都是$\mathcal{L}$中某些元素的超集。$\langle \mathcal{L}, \mathcal{R}\rangle$界定了一个项集的集合，它是由所有满足以下条件的项集构成的集合：它既是$\mathcal{L}$中某个元素的超集，又是$\mathcal{R}$中某个元素的子集。若$\mathcal{L}$为空，则称这个序偶为水平边界。

Dong和Li[7]提出的发现算法首先分别发现两个数据集$\mathcal{D}_1$和$\mathcal{D}_2$支持度不为零的所有项集的集合，这两个集合可以分别由水平边界$\langle\{\phi\}, \mathcal{R}_1\rangle$和$\langle\{\phi\}, \mathcal{R}_2\rangle$来表示，Dong将发现所有项集的问题转化为发现两个集合的右边界$\mathcal{R}_1$和$\mathcal{R}_2$的问题，从而避免了传统频繁候选项集生成步骤。最后，通过集合边界的减法分别得到$\mathcal{D}_1$到$\mathcal{D}_2$以及$\mathcal{D}_2$到$\mathcal{D}_1$的所有JEP的集合$(\mathcal{D}_1, \mathcal{D}_2)$和$(\mathcal{D}_2, \mathcal{D}_1)$。算法$FindJEP(\mathcal{D}_1, \mathcal{D}_2, \cdots, \mathcal{D}_q)$给出了发现$q$类数据之间所有JEP的方法，如下所示。

输入：$\mathcal{D}_1, \mathcal{D}_2, \cdots, \mathcal{D}_q$；
输出：$JEP(\mathcal{D}_i, \bigcup_{1\leqslant j\leqslant q, j\neq i} \mathcal{D}_j)(1\leqslant i\leqslant q)$，简记为$JEP(i)$

```
for(i=1;i≤q;i++)
    R_i=φ;
for all itemset t in Δ_i do
    R_i=R_i∪{t};
for(i=1;i≤q;i++)
    JEP(i)=R_i−∪_{1≤j≤n,j≠i} R_j;
return JEP(1),JEP(2),…,JEP(q);
```

3.2 项集独立支持度计算方法

正如前面所提到的，将传统的项集支持度直接应用于分类器的构建存在许多不足之处，而项集独立支持度能够更加全面地描述数据的分布特征，因此更适合分类器的构建。项集独立支持度可以直接由项集支持度计算得到，图2和图3说明了这一过程，其中项集abcde的支持度为0。

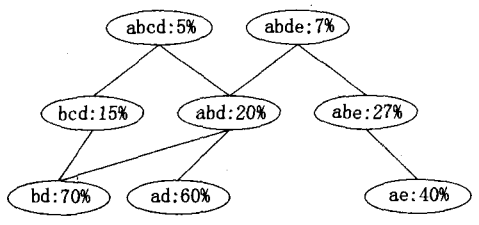

图2 项集支持度

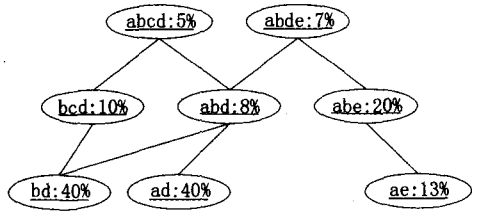

图3 项集独立支持度

首先，根据项集之间的包含关系，将项集排序形成如图2所示的层次结构，图2中的线表示上层项集与下层项集之间的包含关系。为简单起见，项集X的所有父项集的集合记为$P(X)$。随后，根据项集的包含关系，从最底层的项集入手，逐层计算其独立支持度，其独立支持度可以由其本身和父项集以及父项集的父项集的支持度计算得到，计算公式如下：

$$supp_{in}(X)=supp(X)-\sum_{h\in p(X)} supp(h)+\sum_{g\in p(p(X))} supp(g) \tag{2}$$

3.3 分类方法

在分类特征的计算过程中，与Li[6]的JEPC相比，我们充分考虑了测试数据与训练数据集整体的相似程度。本文3.2节定义的JEP平均长度的概念可以较为全面地衡量一个JEP集合对数据集的描述程度，即平均长度越长的JEP集描述得越全面，包含这个JEP集的测试数据更倾向于属于该数据集。对于多个类别的分类问题，选择测试样本所覆盖的JEP集中平均长度最大的所属的类别作为该样本的类别。对于前面所述的q类数据分类问题，分类过程如下伪代码所示：

```
1.  for(i=1;i≤q;i++)
2.      for all itemset t in D_i do
3.          for all itemset d in JEP(i) do
4.              if t contains d
5.                  d.support++;
6.  for(i=1;i≤q;i++;)
7.      JEP_T(i)=φ;
8.      for all itemset t in JEP(i) do
9.          if T contains t
10.             JEP_T(i)=JEP_T(i) ∪{t};
11.     for all itemset d in JEP_T(i) do
12.         d.in_support=supp_in(d);
13.     Let Length(i)=Σ_{d∈JEP_T(i)} d.in_support * d.length;
14.     Let Coverage(i)=Σ_{d∈JEP_T(i)} d.in_supp;
15. Let class={k|Length(k)=max_{1≤i≤q}{Length(i)}};
16. if |class|=1
17.     T∈D_class[1];
18. else
19.     T∈D_class[k](Cover_{D_class[k]}(JEP(k))=max_{1≤i≤|class|}{Cover_{Dclass[i]}
        (JEP_T(i))});
```

算法中首先找出各类数据相对于所有其它类数据的所有JEP的集合$JEP(\mathcal{D}_i, \{t\mid t\in(\mathcal{D}_j, j\neq i\})(1\leqslant i\leqslant q)$简记为$JEP(i)$。然后，扫描各个数据集，计算所有JEP的支持度。继而在各个$JEP(i)$中分别计算出测试样本T所覆盖的全部JEP，记为$JEP_T(i)$。根据公式(2)所给出的方法计算$JEP_T(i)$中各个JEP的独立支持度。最后，由公式(1)给出各个$JEP_T(i)$的JEP平均长度，将T划分给平均长度最大的数据集，若T在各个数据集中覆盖JEP的平均长度相等，则将T划分为对测试数据覆盖率最大的类。

4 相关工作

虽然分类一直是数据挖掘领域的热点研究问题，然而，利用关联规则进行分类的研究却相对较少[6~10]。Li等[6]认为边界JEP可以表达两个数据集之间足够的分类特征，仅仅使用两个数据集的边界JEP即可很好地建立起分类器。然而，这种分类方法中JEP的分类能力没有得到充分的利用。因此，相对来说，该方法对于类边界上的数据的区分能力不强。本文提出的JEP平均长度的概念更充分地利用了类间JEP的分类能力，对于类边界上的数据具有更好的区分能力。

Dong和Li提出的基于EP(Emerging Pattern)的分类方法[7]，使用EP作为分类的基本元素，EP是两个数据集中支持度相差比较大的那些项集，它包含了JEP。因为EP中既包含了类间独有的特征，也包含了类间共有的特征(只是程度不同)，这就可能导致分类时的干扰，因此JEP方法和EP方

(下转第185页)

从实验结果可以看出，随着博弈规模的增大，算法收敛到合理 ε 值的近似解。

4.2 算法的运行效率

为了测试算法在较大规模 Agent 交互模型上的运行效率，我们在一组不同规模的 Road 博弈上进行了实验。实验中，Road 博弈采用图 3 所示的效用矩阵，ε 阈值为 0.3（即算法收敛到 ε≤0.3 时停止迭代），针对 Agent 数目分别为 10，20，30，40，…，100 的 Road 博弈，我们测试了本文提出的 Nash 均衡求解算法，其执行时间如图 5 所示。

从实验结果可以看出，算法的效率并不随 Agents 的增多而显著下降，并且对于大多数情况有较合理的收敛结果。因此，我们提出的算法对于求解连续图型博弈的 Nash 均衡具有一定的可行性和高效性。

总结与展望 博弈论在多 Agent 系统交互问题的研究中具有重要的理论价值和应用背景。图型博弈是近年来提出的一种重要的基于博弈论的多 Agent 交互模型。Nash 均衡是多个理性 Agent 交互的预期结局，求解 Nash 均衡是图型博弈的关键问题。与已有的基于 Agent 策略离散化的求解方法不同，本文把求解图型博弈的 Nash 均衡看作是一个函数优化问题，定义了目标函数，并把目标函数最优解的求解归结为一组线性规划，进而提出了一个求解连续策略空间中图型博弈 Nash 均衡的新型算法。并且，为了验证我们提出算法的可行性、收敛性，以及分析算法的运行效率，我们对在一些经典的博弈模型及复杂图型博弈上对算法进行了实验分析。实验结果表明我们提出的算法具有一定的可行性和高效性。本文研究的内容和提出的方法也引发了另外一些有价值的研究课题，例如在处理较大规模多 Agent 交互模型时，进一步提高算法的求解精度、求解非完全信息情况下多 Agent 图型博弈的 Nash 均衡，这些也是我们正在进行的研究工作。

参 考 文 献

1 Kearns M, Littman M L, Singh S. Graphical models for game theory [A]. In: Proceedings of the 17th Conference on Uncertainty in Artificial Intelligence [C]. Seattle, USA, 2001. 253～260

2 Koller D, Milch B. Multi-agent influence diagram for representing and solving games [A]. In: Proceedings of the 17th International Joint Conferences on Artificial Intelligence [C]. Seattle, USA, 2001. 1027～1034

3 La Mura P. Game networks [A]. In: Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence [C]. Stanford, CA, USA, 2000. 335～342

4 Fudenberg D, Tirole J. Game Theory [M]. Cambridge, MA, MIT Press, 1991. 4～42

5 Papadimitriou C H. Algorithms, Games, and the Internet [A]. In: Proceedings of the Thirty-Third Annual ACM Symposium on Theory of Computing [C]. Crete, Greece, 2001. 749～753

6 Conitzer V, Sandholm T. Complexity Results about Nash Equilibrium [A]. In: Proceedings of the 18th International Joint Conferences on Artificial Intelligence [C]. Acapulco, Mexico, 2003. 765～771

7 McKelvey R, McLennan A. Computation of equilibria in finite games [A]. In: Handbook of computational Economics, Elsevier Science, 1996, 1:87～142

8 Littman M L, Kearns M, Singh S. An efficient exact algorithm for singly connected graphical games [A]. In: Proceedings of the 14th Neural Information Processing Systems [C]. Vancouver, British Columbia, Canada, 2001. 817～823

9 Ortiz L E, Kearns M. Nash propagation for loopy graphical games [A]. In: Proceedings of the 15th Neural Information Processing Systems [C]. Vancouver, British Columbia, Canada, 2002. 793～800

10 Parsons S, Gmytrasiewicz P, Wooldridge M. Game Theory and Decision Theory in Agent-based Systems [M]. Kluwer Academic Publishers, 2002,5

11 Wooldridge M. An Introduction to Multiagent Systems [M]. John Wiley & Sons, 2002

（上接第 167 页）

法是可以互补的方法，它们分别适用于不同类型的数据集。

Liu 等[8]将类标志作为一个属性参与关联规则的挖掘过程，然后逐条利用得到的关联规则，特别是带有类标志的关联规则来训练分类器，将那些无助于分类，甚至对分类产生副作用的关联规则去除，最后用剩余的关联规则来构建分类器。但 Liu 对于分类器的训练方法也存在着对于训练集过度适应的问题。

总结 本文研究了 JEP——一种在不同数据集之间支持度从零到非零跳跃性变化的项集在数据分类中存在的问题，提出了项集独立支持度的概念。相对于传统的项集支持度来说，独立支持度能够更全面地描述数据的分布特征，为更加准确的分类提供依据。进而，在独立支持度的基础上提出了 JEP 平均长度的概念，并提出了一种以测试样本所覆盖 JEP 的平均长度作为分类特征的分类方法，该方法可以更加有效地区分类边界上的数据，能够为数据提供更为准确的分类。我们还将进一步研究 JEP 分类与概念树的结合以及 JEP 在序列分类中的应用等问题。

参 考 文 献

1 Han Jiawei, Kamber M. Data Mining: Concepts and Techniques. Morgan Kaufman Publishers, Inc. 2001

2 Agrawal R, Imielinski T, Srikant R. Mining association rules between sets of items in large databases. In: Proceedings of 1993 ACM SIGMOD International Conference on Management of Data, Washington, D.C., May 1993. 207～216

3 Han J, Pei J, Yin Y. Mining Frequent Patterns without Candidate Generation. In: Proceedings of the 2000 ACM SIGMOD international conference on Management of data, Dallas, Texas, United States, May 2000. 1～12

4 Zaki M J, Hsiao C. CHARM: An Efficient Algorithm for Closed Itemset Mining: [Technical Report 99-10]. Computer Science Dept., Rensselaer Polytechnic Inst., Oct. 1999

5 Quinlan J. R C4.5: Programs for Machine Learning. San Mateo, CA: Morgan Kaufmann, 1993

6 Li J. Dong G, Ramamohanarao K. Making Use of the Most Expressive Jumping Emerging Patterns for Classification. In: Proceedings of the Fourth Pacific-Asia Conference on Knowledge Discovery and Data Mining, Kyoto, Japan, 2000. 220～232

7 Dong G, Li J. Efficient Mining of Emerging Patterns: Discovering Trends and Differences. In: Proceedings of the fifth ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining, San Diego, CA, ACM Press, New York, August 1999. 43～52

8 Liu B, Hsu W, Ma Y. Integrating Classification and Association Rule Mining. In: Proceedings of the fourth Int'l Conf. on Knowledge Discovery and Data Mining, New York, AAAI Press, August 1998. 80～86

9 Lent B, Swami A, Widom J. Clustering Association Rules. In: Proc. 1997 Int. Conf. Data Engineering (ICDE'97), Birmingham, England, April 1997. 220～231

10 Wang Y, Wang K C. From Association to Classification: Inference Using Weight of Evidence. IEEE Transactions on Knowledge and Data Engineering, 2003, 15(3): 764～767