

面向范畴类型数据的 sIB 算法

叶阳东¹, 何锡点¹, 贾利民²

(1. 郑州大学信息工程学院计算机科学系, 河南郑州 450052;

2. 北京交通大学轨道交通控制与安全国家重点实验室, 北京 100044)

摘 要: 本文针对 sIB 算法仅适用于共现数据的问题, 提出了一种能够自动进行范畴类型数据分析的 sIB 算法: CD-sIB. 该算法根据范畴类型数据的离散化表示、不同属性值有限的特征, 进行数据的属性的拓展和二元化处理, 基于属性值的出现进行 X, Y 的联合分布的计算, 使得 sIB 算法可有效应用于范畴类型数据的分析. 实验结果表明: CD-sIB 算法相对于现有的面向范畴类型数据聚类模式分析的算法 GAClust 和 K-modes 具有明显的优势; CD-sIB 算法在进行数据属性概化程度高、类数据分布相对平衡的范畴类型数据的分析中, 在效率和精确度方面均很突出.

关键词: IB 理论; sIB 算法; 范畴类型数据; 概化; 聚类

中图分类号: TP18

文献标识码: A

文章编号: 0372-2112 (2009) 10-2165-08

CD-sIB: A Kind of sIB Algorithm Orient to Categorical Data

YE Yang-dong¹, HE Xi-dian¹, JIA Li-min²

(1. School of Information Engineering, Zhengzhou University, Zhengzhou, Henan 450052, China;

2. State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, Beijing 100044, China)

Abstract: The sIB algorithm has previously been only applied to the analysis of co-occurrence data. Therefore, it cannot directly analyze categorical data that do not appear in the form of co-occurrence of two variables X, Y. Aiming to solve the problem, this paper proposes a CD-sIB algorithm for automatically analyzing categorical data based on the theory of sIB algorithm. According to the nature that categorical data is discrete and its distinct attribute value is finite, CD-sIB algorithm counts joint distribution of relevant variable X, Y based on the occurrence frequency of attribute value by extending the attributes of dataset and utilizing binarization to process the categorical data. Consequently, our algorithm can be effectively employed in analyzing the categorical data. As shown by our experimental results, CD-sIB outperforms the GAClust and the K-modes algorithm, and it achieves high precision and efficiency in analyzing categorical data, especially in the analysis of categorical data which is highly generalizable and comparatively balanced in the data distribution of each class.

Key words: IB theory; sIB algorithm; categorical data; generalization; clustering

1 引言

IB 方法 (Information Bottleneck Method) 是 Tishby, Pereira 等人于 1999 年提出^[1], 它起源于香农的率失真理论. 其基本思想: 用源变量和相关变量的联合概率分布对源变量进行压缩, 使压缩变量最大化地保存相关变量的信息. IB 方法在很多领域得到应用, 其中包括: 有监督的和无监督的文档分类^[2,3]、图像聚类^[4]、自然语言处理^[5]、视频图像检索^[6]等等. 2002 年, Slonim 将其命名为 IB 理论^[7].

sIB 算法是 Slonim 于 2002 年提出, 该算法将待分析的数据对象按照其与另一数据对象的相关性进行“硬”划分, 使得划分在一起的对象充分体现出源数据对象蕴

含的某个特征模式. 相对于 1999 年 Tishby 等提出的 iIB^[1]、dIB^[1]以及 Slonim 提出的 aIB^[8]等 IB 算法, 该算法具有较低的时间和空间的复杂度且可以保证得到问题的局部稳定优化解^[2], 这些优点使 sIB 算法得到广泛的应用. 但是 sIB 算法仍存在下述的三个问题: 第一, 随机选取的初始解易使算法陷入局部优化解; 第二, 压缩变量的参数需要用户指定; 第三, sIB 算法只能应用于共现数据 (Co-occurrence Data), 即 sIB 算法需要事先得到相关变量 X, Y 的先验联合分布, 从而限制了 sIB 算法的应用. 为了解决问题一, 文献[9,10]提出了各种方法对 sIB 算法得到的局部解进行优化, 取得了较好的效果. 针对问题二, 文献[11]根据 IB 算法对有限数据样本概率分布的依赖性, 推导出有限数据集的最大特征模式数目;

收稿日期: 2007-12-24; 修回日期: 2009-04-23

基金项目: 国家自然科学基金 (No. 60773048)

文献[12]将 sIB 算法应用于文本聚类时,设计了一种聚类目标函数能够发现文本数据集的特征子集和真实聚类数目;文献[13]基于最小描述长度原理提出了一种能够自动确定参数的 sIB 算法(AsIB)。针对问题三,2004 年 Periklis Andritsos 等人提出 LIMBO^[14],一种基于 aIB 的面向范畴类型数据的凝聚的层次聚类 IB 算法,其时间和空间的复杂度较高且不能保证得到问题的局部稳定优化解;2006 年 Yevgeny Seldin 等人提出了一种独立于模型的、一般性的方法^[15],以解决 IB 理论应用于特定的非共现数据,该方法中数据的行用 X 表示,列用 Y 表示,每行数据的元素是通过一个函数 $Z(X, Y)$ 的样本值来表示,而不是 X, Y 的统计值。在一般的范畴类型数据集中,函数 $Z(X, Y)$ 的值是很难确定的,所以文献[15]中的方法难以有效的应用于范畴类型数据。

本文旨在解决 sIB 算法不能直接用于范畴类型数据的问题,提出一种能够自动进行范畴类型数据模式提取的 CD-sIB 算法。该算法采用一种有效的特征构造方法,对范畴类型数据进行属性的拓展和二元化处理,基于属性值的出现进行 X, Y 的联合分布的计算,使 CD-sIB 算法能够有效应用于范畴类型数据的分析。实验表明:CD-sIB 在范畴类型数据模式分析中相对于常用的范畴类型数据的算法 K-modes^[16,17]和 GAClust^[18]具有明显的优势;CD-sIB 算法在进行数据属性概化程度高、类数据分布相对平衡的范畴类型数据的分析中,在效率和精确度方面均突出。

本文提出的算法相对于已有的工作有以下不同之处:

- (1) 提出了一种能够自动进行范畴类型数据模式分析的 sIB 算法,拓展了 IB 算法的应用范围;
- (2) 将特征构造和二元化方法引入到 sIB 的研究中;
- (3) 将 sIB 算法应用于范畴类型数据分析,发现 sIB 中的互信息损失能够有效的用于范畴类型数据之间的距离度量。

2 背景知识

2.1 IB 理论

香农的率失真理论是 IB 方法的思想起源,它寻求源随机变量 $X \sim p(x)$ 压缩到压缩变量 T 的编码方案并通过失真度量函数 $d(x, t)$ 度量该过程中产生的失真,使该编码所产生的期望失真小于预先指定的值 D 的条件下压缩信息 $I(X; T)$ 达到最小。该理论可形式化表示为: $R(D) = \min_{\{p(t|x): E(d(x, t)) \leq D\}} I(T; X)$, 其中 $E(d(x, t))$ 是以 $p(t|x)$ 为自变量的失真度量函数 $d(x, t)$ 的数学期望。

率失真理论的数据分析方法的问题是难以寻找合适的失真度量函数 $d(x, t)$ 去准确度量数据分析结果的误差。为了有效地解决该问题,IB 理论引入了变量 X 的相关变量 Y ,并推导出一个合适的失真度量函数。Tishby 在文献[1]中详细介绍了 IB 方法的思想:假设具有相关性的随机变量对 (X, Y) 满足 $(X, Y) \sim p(x, y)$, 寻找变量 X 的压缩表示 T ,使得变量 X 和压缩变量 T 之间的互信息 $I(T; X)$ 最小化,即尽可能的压缩数据,同时,在这个过程中使相关变量 Y 和 T 之间的互信息 $I(T; Y)$ 最大化,即尽可能的保存相关结构。在压缩数据和保存相关结构的过程中随机变量 T 就相当于数据 X 和相关信息 Y 之间的瓶颈。该问题的形式化描述和相关目标函数的形式解可见文献[1]。

2.2 范畴类型数据的相关定义

范畴类型数据是离散、无序、属性的类型不一致,值与值之间没有自然的序关系,属性的值域有限,以元组的形式出现。具体定义和相关符号见定义 1,表 1 所示的是一个范畴类型数据集的例子。

表 1 范畴类型数据集例子:个人爱好数据集

	Sports	Drinks	Movie . genre
x_1 (Tom)	Football	Coke	Crime
x_2 (John)	Basketball	Pepsi	Crime
x_3 (Mary)	Fence-play	Mocca	Thriller
x_4 (Jane)	Skating	Nescafe	Thriller
x_5 (Helen)	Football	Pepsi	Comedy
x_6 (Andy)	Basketball	Nescafe	Comedy

定义 1 设 $\{A_1, A_2, \dots, A_m\}$ 为范畴类型属性集合,其中属性 $A_i (i = 1, \dots, m)$ 中的不同属性值是有限的, $|A_i|$ 是其不同属性值的数量, A_i 属性的值域 $Dom(A_i)$ 是一个有限集合。范畴类型数据对象定义为: $O = (c_1, c_2, \dots, c_m)$, 其中 $c_i \in Dom(A_i), 1 \leq i \leq m$, 范畴类型数据集定义为: $D = \{O_j | j = 1, \dots, n\}$ 。

范畴类型数据的特性决定了寻求度量数据对象之间的相似度方法的困难。IB 算法在相关模式的分析过程中,把数据集中的每个数据对象 $x \in X$ 当成一个簇,然后选择其中一个簇和其余的簇逐个进行合并,并计算每一次合并过程中的信息损失 (Information Loss), 选择一个信息损失最小的簇进行合并,其相应的数据对象之间的相似度度量是用信息损失表示。其相关的应用也验证了其有效性。本文中的算法是基于 sIB 算法的,在模式分析时仍采用信息损失进行范畴类型数据对象之间的相似度度量。相关的说明可见定义 2、3。

定义 2 $x_1, x_2 \in D$, 既 x_1, x_2 是范畴数据集集中的数据对象,那么 x_1, x_2 之间的距离定义为:

$$Cost(x_1, x_2) = [p(x_1) + p(x_2)] \times JS_{-1, -2}(p(y|x_1), p(y|x_2)) \quad (1)$$

$Cost(x_1, x_2)$ 也是 x_1, x_2 作为单独簇合并成一个新簇的信息损失, 其中 $\frac{1}{2}$ 为权值, 具体值为:

$$\frac{1}{2} = \frac{P(x_1)}{P(x_1) + P(x_2)}, \quad \frac{1}{2} = \frac{P(x_2)}{P(x_1) + P(x_2)} \quad (2)$$

定义 3 $x_1, x_2, x_3 \in D$, 若 $Cost(x_1, x_2) > Cost(x_1, x_3)$, 称 x_1, x_2 之间的距离大于 x_1, x_3 之间的距离。

3 面向范畴类型数据的 CD-sIB 算法

3.1 sIB 算法

Slonim 等人在文献[2]中将 IB 目标函数变换为等价的式(5)且取平衡参数 $\alpha = 1$, 得到 sIB 算法的目标函数 $F = I(T; Y)$ 。针对联合概率分布 $p(x, y)$ 和用户输入的压缩变量参数 k , 将数据集 X 随机地划分为 k 个子集, 在此基础上迭代地将 X 中的每个元素 x 从其所在的 t 中取出, 做为一个单独的新簇, 然后将 x 重新分配到满足 $t^{new} = \arg\min_t I(T; Cost(\{x\}, t))$ 的 t^{new} 中, 这里 $Cost(\{x\}, t)$ 表示将 x 指派到 t 引起的互信息 $I(T; Y)$ 值的减小, 其值可以由定义 2 计算。迭代结束后, sIB 算法得到了将数据 X 划分为 k 个子集的数据分析结果 T 。

$$F_{max}[p(t|x)] = I(T; Y) - \alpha^{-1} I(X; T) \quad (3)$$

sIB 算法的时间和空间复杂度分别为 $O(|X| |T| |Y|)$ 和 $O(|X|^2)^{[2]}$, 其中 l 是一个正的常数。相对于其它 IB 算法, 这种具有较低时空开销的算法更宜于应用。此外, 该算法在迭代结束时, 总是保证可以得到稳定局部优解。但是, sIB 算法只适用于共现数据, 需要事先得到相关变量 X, Y 的先验联合分布, 这在一定程度上限制 sIB 算法的应用。

为了拓展 sIB 算法的应用范围, 使 sIB 算法能够用于范畴类型数据模式分析, 需要解决三个问题: (1) 找到 IB 理论中相关变量 X, Y 在范畴类型数据中的合适含义; (2) 要寻求一种适合 IB 算法的范畴类型数据对象之间的相似度的度量方法; (3) 求出相关变量 X, Y 的联合分布 $p(x, y)$ 。

3.2 范畴类型数据中相关变量

IB 方法把从数据中提取结构看成是压缩数据的同时保存相关信息, 数据用随机变量 X 表示, 相关信息就用第二个随机变量 Y 表示, 随机变量 X 的压缩表示为 T 。例如: 把 IB 算法应用于文档聚类模式分析时, X, Y 都有具体的含义, X 代表文章, Y 代表文章中的单词, 于是, 原文章就转化成由最具有代表性的单词的统计值来描述, 在文章 X 和单词 Y 的共现矩阵中的属性值是以 (X, Y) 的共现统计值出现的。那么在范畴类型的数据中 X 和 Y 应该要代表什么呢? 根据 IB 算法在文档聚类模式分析中相关变量的含义, 在本文中让 X 表示元组, 即数据集中的数据对象 (数据集中的每一行数据), Y 表示描述元组的不同属性值, 即把数据集中所

有不同的属性值看成一个变量 Y , 因此, 在进行范畴类型数据模式分析过程中的寻求变量 X, Y 的共现统计值的问题就和 sIB 算法应用于文档模式分析中方法一致。

3.3 特征构造和二元化转化

范畴类型数据不是以共现数据对 (X, Y) 的形式出现, 其形式不适合 sIB 算法, 因此, 必须对原数据集进行重新构造, 在对范畴类型数据进行模式分析时, 采用特征构造对原数据集进行重新构造: X 代表元组, Y 代表描述元组的不同属性值。这种重新构造后, 新数据集中元组就变成了由不同属性值和原元组构成的共现统计量的一种表示, 即原数据转化成由原元组 X 和不同属性值 Y 的共现矩阵且矩阵中的元素值是以 X, Y 共现统计值的形式出现。

具体的特征构造和二元化方法如下: 在对范畴类型数据进行聚类模式提取时, 输入的数据集 X 是由 n 条记录 x 组成的。根据定义 1, 设 $Y = A_1 \cup A_2 \cup \dots \cup A_m$ 代表数据集中所有不同的属性值集合, 数据集中所有不同的属性值的个数为 $|Y| = d = |A_1| + |A_2| + \dots + |A_m|$, 把这 d 个不同的属性值作为新数据集的特征, 那么新的数据集为 $n \times d$ 的矩阵 M , 在此矩阵中每行数据对象 $x \in X$ 是一个 d 维的行向量。如果每行数据对象 $x \in X$ 包含属性值 $y \in Y$, 那么在矩阵 M 中相应的 $M[x, y] = 1$, 否则 $M[x, y] = 0$ 。

原数据集中的每行数据对象都有 m 个属性, 所以在矩阵 M 中每行向量中 1 出现的次数的总和为 m 。为了使矩阵 M 中的每行数据之和为 1, 可以对矩阵 M 进行规范化 (Normalize) 处理。对于数据集中某一条记录 $x \in X$, 在规范化处理后矩阵 M 中的对应行中的条件概率分布为 $p(Y|x)$ 。由于数据集中的每行记录都包含 m 个属性, 所以对于某一个属性值 $y \in Y$ 如果在记录 x 中出现, 则 $p(y|x) = 1/m$, 否则为 $p(y|x) = 0$ 。表 2 是表 1 数据集中所有不同属性值和数据对象 (数据集中的每一行数据) 联合分布经过二元化及规范化处理后矩阵。

表 2 规范化后的联合条件分布矩阵

	S.F	S.B	S.Fp	S.S	D.C	D.P	D.M	D.N	M.Gr	Mg.T	M.C	$p(x)$
x_1	1/3	0	0	0	1/3	0	0	0	1/3	0	0	1/6
x_2	0	1/3	0	0	0	1/3	0	0	1/3	0	0	1/6
x_3	0	0	1/3	0	0	0	1/3	0	0	1/3	0	1/6
x_4	0	0	0	1/3	0	0	0	1/3	0	1/3	0	1/6
x_5	1/3	0	0	0	0	1/3	0	0	0	0	1/3	1/6
x_6	0	1/3	0	0	0	0	0	1/3	0	0	1/3	1/6

在某一分析结果 T 中, 对数据集中的每一条记录 $x \in X$ 都赋予同样的权重, 那么 $p(x) = 1/n$, 对于结果 T 中的一个元素 $t \in T, x \in X, y \in Y$ 有以下关系:

$$p(t) = \sum_{x \in X} p(x) = \frac{|t|}{n}$$

和 $p(y|t) = \frac{1}{p(t)} p(x) p(y|x)$.

3.4 CD-sIB 算法描述

针对 sIB 算法不能直接进行范畴类型数据分析的问题, CD-sIB 算法引入了合适的相关变量 Y , 对范畴类型数据构建新的特征并进行二值化处理, 求得了 X, Y 的联合分布, 从而扩展了 sIB 算法应用范围, 同时也解决了在 IB 理论框架中范畴类型数据对象之间的相似度的度量问题. CD-sIB 算法描述如下.

输入:

数据集 X (X 是 $n \times m$ 的矩阵).

平衡参数 λ .

基数值 k .

输出:

X 到 k 簇的划分 T .

步骤 1: 求出 X 和 Y 的联合分布 $p(x, y)$.

1. 特征构造: Y 为数据集 X 中所有不同的属性值集合, 数据集中所有不同的属性值的个数为 $|Y| = d$, 把这 d 个不同的属性值作为新数据集的特征, 那么新的数据集为 $n \times d$ 的矩阵 M .

2. 二值化处理并得到 X 和 Y 的联合分布 $p(x, y)$

步骤 2: 初始化.

3. T 为 X 到 k 簇随机的划分;

步骤 3: 主循环.

4. While not Done

5. Done = TRUE;

6. For every $x \in X$:

7. 从当前簇 $t(x)$ 中移除 x , 形成单独的簇 $\{x\}$;

8. $t^{new} = \operatorname{argmin}_t \tau Cost(\{x\}, t)$;

9. If $t^{new}(x) \neq t(x)$ then

10. Done = FALSE;

11. End if

12. 把 x 合并到 $t^{new}(x)$ 中;

13. End for

14. End while

3.5 CD-sIB 算法分析

CD-sIB 算法中步骤 1 第 1 步的时间复杂度为 $O(|X| |Y|)$, 第 2 步的时间复杂度为 $O(|X| |Y|)$, 所以算法在步骤 1 中的时间复杂度为 $O(|X| |Y|)$; 步骤 2 中时间复杂度为常数; 步骤 3 中是主循环, 第 7 - 12 步将一个数据对象合并到某个簇中, 其时间复杂度为 $O(|X| |Y|)$, 当第 4 - 14 步算法迭代结束时, 其时间复杂度为 $O(l |X| |Y| |T|)$, 其中 l 是算法找到局部最优解的循环次数, $|T|$ 是簇的个数, $|Y|$ 是所有不同属性值的个数, $|X|$ 是数据集中数据对象的个数. 综合考虑整个算法, 可以得到 CD-sIB 算法的时间复杂度为 $O(l |X| |Y| |T|)$. 同理可以分析出 CD-sIB 算法的空间复杂度为 $O(|T|^2)$.

CD-sIB 算法继承了 sIB 算法特性, 其和面向范畴类型数据的凝聚层次聚类 IB 算法——LIMBO 算法^[14]相比, CD-sIB 有以下的优势: (1) CD-sIB 算法可以保证得到

一个局部稳定最优解, 而 LIMBO 无法保证得到局部最优解; (2) CD-sIB 算法时间复杂度比 LIMBO 算法的时间复杂度低; (3) LIMBO 包含三个步骤: (1) 建立 DCF 树; (2) 对数据进行聚类模式分析; (3) 把簇中的 DCF 树节点和数据对象联系起来; 而 CD-sIB 算法在求得联合分布后直接进行数据模式分析, 过程较简洁.

4 实验与性能分析

4.1 实验评估方法

算法的实验均是基于范畴类型数据, 将 CD-sIB 算法压缩的参数看成是所要分析的聚类模式, 最终的分析结果完全可以具体为聚类模式分析. 为了评估 CD-sIB 的性能, 本文的实验采用聚类结果的正确率 (AC)、精确率 (PR) 和召回率 (RE) 来评估聚类算法的优劣. 其衡量方法的定义如下:

$$AC = \frac{\sum_{c=1}^k a_c}{n}, \quad PR = \frac{\sum_{c=1}^k \left(\frac{a_c}{a_c + b_c} \right)}{k},$$

$$RE = \frac{\sum_{c=1}^k \left(\frac{a_c}{a_c + c_c} \right)}{k}$$

其中 a_c 是正确归到簇 c 的数据对象的个数; b_c 是不应该归到簇 c , 但却归到簇 c 的数据对象的个数; c_c 是错误的拒绝了原本属于簇 c 的数据对象的个数; k 是数据集中簇的个数; n 是数据对象的总数.

4.2 实验数据集

实验所涉及的 18 个数据集来自于 UCI Machine Learning Repository^[19]. 这些数据集在 K-modes, GACust, ROCK, CACTUS, COOLCAT 等面向范畴类型数据的算法研究中使用.

4.3 实验设计和结果分析

4.3.1 实验设计

为了验证 CD-sIB 算法的有效性, 本文针对不同的数据集进行了 5 组实验. 其中, 前两组实验分别从数据集类别规模和数据集数据量的规模来说明 CD-sIB 算法的有效性和鲁棒性; 后三组实验主要是分析 CD-sIB 算法的性能, 探索 CD-sIB 算法的适用范围, 其中的第 3、4 组实验是说明相关变量 Y 的降维对 CD-sIB 算法性能的影响, 第 5 组实验是数据集中各类数据分布不平衡对 CD-sIB 算法的影响.

为评估 CD-sIB 算法的性能, 本文将该算法和 K-modes、GACust 进行对比. K-modes 算法实际上是 K-means 算法面向范畴类型数据模式分析的算法; GACust 算法是基于信息论的基因遗传变异算法, 在聚类模式分析过程中, 簇之间的距离是用条件熵来度量. 实验所采用的 K-modes 算法、GACust 算法, 其对应实施工具是 C++、Ja-

va,而 CD-sIB 算法是基于 MatLab,全部实验的物理运行时间表明,CD-sIB 算法运行时间优于 GAClust 算法,不如 K-modes 算法,由于不同的实施语言对算法的时间参数影响很大,我们认为这种物理运行时间对比不一定公正,因此,本文不进行不同算法的运行时间对比分析。

实验的相关说明:(1) CD-sIB 算法实验中的随机初始参数 Restarts 均设置为 10,IB 函数式(3)中的平衡参数 为 ;(2) GAClust 算法的参数 population 设为 50,其它参数为算法设定的默认值^[18];(3) K-modes 算法要求设定数据集中数据对象的数量。

4.3.2 类别数不同的数据集对比实验

为了验证 CD-sIB 算法在类别数不同的数据集上的性能,本组实验中选定了 5 个范畴类型的数据集,其对应的类别数分别是 2 类、3 类、4 类、7 类和 19 类,分别调

用 CD-sIB、K-modes 和 GAClust 算法来对这 5 个数据集进行聚类模式分析,实验结果见表 3 和图 1。从中可见:CD-sIB 算法的总体性能优于 K-modes 和 GAClust,CD-sIB 的平均正确率为 76.2%,比 GAClust 高 20.2%,比 K-modes 高 11.8%。

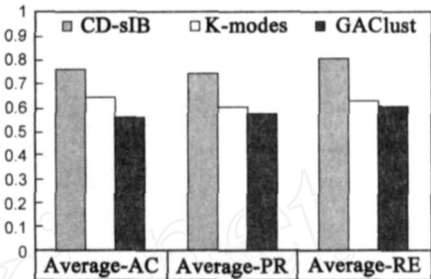


图1 算法在不同数据量的数据集上平均性能比较

表 3 算法在类别数不同的数据集的实验结果

数据集	类别数	CD-sIB			K-modes			GAClust (p = 50)			对比结果
		AC	PR	RE	AC	PR	RE	AC	PR	RE	
votes	2	0.869	0.864	0.883	0.864	0.857	0.873	0.802	0.810	0.826	优
iris	3	0.840	0.840	0.840	0.887	0.894	0.887	0.640	0.702	0.640	相当
Lymphography	4	0.527	0.496	0.756	0.358	0.275	0.310	0.473	0.496	0.669	优
zoo	7	0.871	0.757	0.793	0.594	0.467	0.520	0.703	0.689	0.701	优
Soybean	19	0.701	0.753	0.777	0.517	0.543	0.518	0.182	0.183	0.185	优
平均值	-	0.762	0.742	0.810	0.644	0.607	0.622	0.560	0.576	0.604	优

该组实验说明:CD-sIB 在类别数不同的数据集上均能取得较高的正确率、精确率和召回率,总体性能好于 K-modes 和 GAClust。

4.3.3 不同数据量的数据集对比实验

为验证 CD-sIB 算法在不同数据量的数据集上的性能,本组实验中选定了 5 个范畴类型的数据集,其对应的记录个数分别是 683、690、3190、3196 和 8124,所对应的实验结果见表 4 和图 2。从结果可见,CD-sIB 的平均正确率为 74.2%,比 GAClust 高 14.2%,比 K-modes 高 15.3%。说明:CD-sIB 在数据规模不同的数据集上总体性能优于 K-modes 和 GAClust。

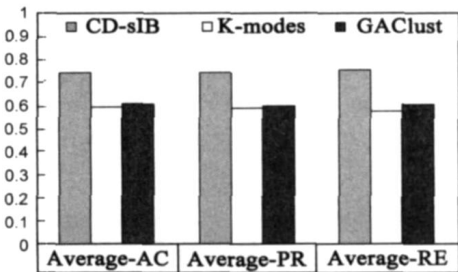


图2 算法在不同数据量的数据集上平均性能比较

表 4 算法在不同数据量的数据集的实验结果

数据集	记录数	CD-sIB			K-modes			GAClust (p = 50)			对比结果
		AC	PR	RE	AC	PR	RE	AC	PR	RE	
Cancer-683	683	0.978	0.972	0.981	0.960	0.961	0.952	0.814	0.802	0.829	优
Credit-a	690	0.733	0.730	0.728	0.710	0.738	0.686	0.707	0.705	0.706	相当
Splice	3190	0.729	0.747	0.769	0.299	0.288	0.268	0.339	0.340	0.339	优
Kr. vs. kp	3196	0.512	0.511	0.511	0.410	0.386	0.423	0.537	0.538	0.538	相当
Mushroom	8124	0.760	0.761	0.761	0.568	0.567	0.564	0.605	0.605	0.605	优
平均值	-	0.742	0.744	0.750	0.589	0.588	0.579	0.600	0.598	0.603	优

4.3.4 祛除和概化值量大属性的实验

该实验选定了8个数据集,其属性值的数量分布不平衡,部分属性的属性值个数明显过大,这种属性在此称之为值量大属性,这类属性对算法的性能影响比较大,在祛除这类属性后,CD-sIB 算法的运行时间明显地减少,但性能提高不多.所以本组实验采用 LUCS-KDD

DN^[20]概化这类属性以达到对数据进行降维的目的.表5是 CD-sIB 算法在祛除和概化相同的值量大属性的数据集上的对比实验结果,从中看出对值量大的属性采用概化的降维方法在算法的正确率、精确率、召回率方面均优于单一的祛除值量大属性的降维方法.

表5 祛除和概化相同的值量大属性的对比

数据集	Del. Column					Generalized. Column					对比结果
	AC	PR	RE	Time (s)	Y	AC	PR	RE	Time (s)	Y	
Hepatitis	0.710	0.680	0.771	1.1	195	0.735	0.693	0.787	1.26	213	优
Diabetes	0.638	0.645	0.659	203	737	0.642	0.647	0.662	164	755	优
Hypothyroid	0.359	0.255	0.354	1634	494	0.379	0.273	0.242	1647	514	略优
Ionosphere	0.729	0.748	0.766	137	7802	0.729	0.748	0.766	136	7809	相当
Credit-a	0.733	0.730	0.728	59	586	0.733	0.730	0.728	82.3	841	相当
Iris	0.780	0.780	0.780	1	80	0.840	0.840	0.840	0.8	95	优
Zoo	0.861	0.742	0.765	0.6	30	0.871	0.757	0.793	0.7	35	优
Lymphography	0.500	0.469	0.624	1.28	43	0.500	0.477	0.743	1	57	优
平均值	0.664	0.631	0.681	-	-	0.679	0.646	0.695	-	-	优

4.3.5 数据属性概化实验

从前组实验可知概化处理对 CD-sIB 算法的性能影响很大,为了进一步验证概化的降维方法对 CD-sIB 性能的影响,本组实验选定了一些数据集,其不同属性值的数量大,也就是原数据集中的相关变量 Y 的值域大.用 LUCS-KDD DN 对这些数量值大的属性进行概化使得原数据降维的同时又保持了相关信息.用 CD-sIB 算法分别对原数据集、概化后的数据集进行实验,以及概化后的数据集在 K-modes、GAClust 算法的实验,具体结果见表6. CD-sIB 算法分别对原数据集、概化后的数据集时间参数见图3.

由表6和图3可见:(1)采用概化的降维方法可以明显提高算法的效率,概化后的数据集上的运行时间

大大的减少,概化前后的运行时间比值最高为40倍,最小的为3倍;(2) CD-sIB 算法在概化后的数据集上平均正确率、平均精确率、平均召回率比在原数据集上分别提高了7.1%、6.3%、2.8%;(3)在概化后的数据集上的结果表明 CD-sIB 性能也优于 K-modes、GAClust 算法.

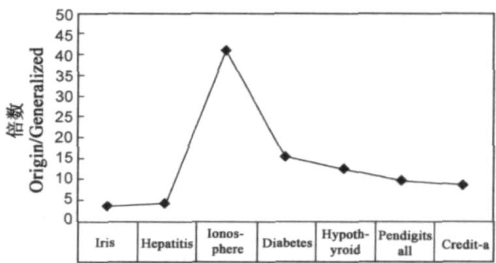


图3 CD-sIB算法在数据概化前后的运行时间比值

表6 CD-sIB、K-modes、GAClust 算法在数据属性概化的实验结果

数据集	CD-sIB (Origin)				CD-sIB Generalized				对比	K-modes Generalized			GAClust Generalized			对比
	AC	PR	RE	Y	AC	PR	RE	Y		AC	PR	RE	AC	PR	RE	
Iris	0.840	0.840	0.840	123	0.953	0.953	0.953	16	优	0.927	0.928	0.927	0.913	0.913	0.913	优
Hepatitis	0.684	0.675	0.766	364	0.735	0.706	0.810	86	优	0.716	0.671	0.752	0.703	0.690	0.790	优
Ionosphere	0.729	0.748	0.766	8082	0.895	0.894	0.874	155	优	0.379	0.320	0.322	0.821	0.807	0.829	优
Diabetes	0.607	0.620	0.631	1254	0.652	0.652	0.667	119	优	0.508	0.454	0.454	0.574	0.525	0.527	优
Hypothyroid	0.351	0.269	0.398	1020	0.398	0.315	0.223	164	略优	0.555	0.245	0.211	0.262	0.248	0.197	略优
Pendigitsall	0.648	0.648	0.647	1608	0.689	0.678	0.683	76	优	0.585	0.568	0.581	0.138	0.138	0.138	优
Credit-a	0.733	0.730	0.728	1175	0.770	0.772	0.759	134	优	0.728	0.755	0.705	0.716	0.718	0.720	优
平均值	0.656	0.647	0.682	-	0.727	0.710	0.710	-	优	0.628	0.563	0.565	0.590	0.577	0.588	优

4.3.6 数据集的类数据分布不平衡的实验

本组实验是为了验证数据集的各类数据分布不平衡对算法的影响.实验选定了6个数据集,其均含有少量的稀有类数据.数据集中的各类数据分布较为不平衡.表7是算法在含有/删除稀有类的数据集上的性能

对比.实验结果表明:CD-sIB 算法对稀有类数据敏感,但是和 K-modes、GAClust 对比,算法的总体性能仍然是相当的,CD-sIB 在类数据分布相对平衡的范畴类型数据的分析性能尤其突出.

表 7 算法在含有稀有类的原数据集/删除稀有类后数据集上的实验结果

数据集	CD-sIB (Origin)			K-modes (Origin)			GAClust (Origin)			对比	CD-sIB (Del)			K-modes (Del)			GAClust (Dels)			对比
	AC	PR	RE	AC	PR	RE	AC	PR	RE		AC	PR	RE	AC	PR	RE	AC	PR	RE	
Zoo	0.871	0.757	0.793	0.594	0.467	0.520	0.703	0.689	0.701	优	0.976	0.967	0.988	0.452	0.366	0.360	0.905	0.871	0.850	优
Lymphography	0.527	0.496	0.756	0.358	0.275	0.310	0.473	0.496	0.669	优	0.690	0.688	0.692	0.479	0.466	0.466	0.648	0.648	0.651	优
Primary-tumor	0.292	0.254	0.400	0.327	0.258	0.347	0.198	0.164	0.216	略差	0.408	0.425	0.473	0.435	0.378	0.437	0.349	0.350	0.380	略优
Car	0.309	0.309	0.366	0.322	0.217	0.163	0.289	0.294	0.371	相当	0.528	0.534	0.546	0.648	0.571	0.587	0.546	0.547	0.564	略差
Hypothyroid	0.351	0.269	0.398	0.696	0.253	0.245	0.270	0.251	0.228	差	0.443	0.351	0.482	0.732	0.332	0.334	0.358	0.338	0.335	略优
Nursery	0.318	0.261	0.233	0.590	0.442	0.452	0.215	0.215	0.171	差	0.426	0.426	0.428	0.467	0.447	0.469	0.367	0.366	0.366	略差
平均值	0.445	0.391	0.491	0.481	0.319	0.340	0.358	0.352	0.393	略优	0.579	0.565	0.602	0.536	0.427	0.442	0.529	0.520	0.524	优

4.3.7 实验结果分析

综合上述的实验结果可以得出如下结论:

(1) CD-sIB 在类别数不同、数据规模不同的数据集上均能取得较高的正确率、精确率和召回率,总体性能优于 K-modes、GAClust,CD-sIB 算法具有很好的鲁棒性和有效性.

(2) 在 CD-sIB 算法中采用概化方法的降维比单一祛除值量大属性的方法优越;采用概化方法可以明显提高 CD-sIB 算法的效率.

(3) CD-sIB 算法在进行数据属性概化程度高、类数据量分布相对平衡的范畴类型数据的分析,在效率和精度方面均优越.

(4) 基于 IB 方法的 CD-sIB 算法是一种可有效进行范畴类型数据模式分析的算法,进一步验证了 sIB 算法是一种优秀的模式分析算法.

5 结论和进一步工作

范畴类型数据不是以共现数据的形式出现,使得 sIB 算法不能用于这种类型的数据分析.本文基于 IB 方法,采用特征构造方法对范畴类型数据构建新的特征,再对新特征数据集进行二元化处理,自动发现 X、Y 的联合分布,使得 sIB 算法能够有效进行范畴类型数据模式分析,拓展了 IB 算法的应用范围.实验结果表明:所提出的 CD-sIB 算法有效、性能优越.

如何拓展 CD-sIB 算法在范畴类型、连续类型等多种属性混合的数据方面的应用,是我们下一阶段的任务之一.

致谢:在 IB 理论的研究过程中,美国 Princeton 大学的 Noam Slonim 博士和澳大利亚 Deakin 大学的 Gang Li 博士为我们提

供了很多帮助,实验中用的 K-modes 算法的源代码是由香港大学的 Joshua Zhexue Huang 博士提供,在此对他们表示衷心的感谢.

参考文献:

[1] N Tishby, F Pereira, W Bialek. The information bottleneck method[A]. Proceedings of 37th Allerton Conference on Communication, Control and Computing[C]. 1999. 368 - 377.

[2] N Slonim, N Friedman, N Tishby. Unsupervised document classification using sequential information maximization[A]. Proceedings of the 25th Ann. Int. ACM SIGIR Conf. on Research and Development in Information Retrieval [C]. 2002. 129 - 136.

[3] N Slonim, N Tishby. Document clustering using word clusters via the information bottleneck method[A]. Proceedings of 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval [C]. Athens, Greece, 2000. 208 - 215.

[4] J Goldberger, S Gordon, H Greenspan. Unsupervised image-set clustering using an information theoretic framework[J]. IEEE Transactions on Image Processing, 2006, 5(2): 449 - 458.

[5] M Gorodetsky. Methods for discovering semantic relations between words based on co-occurrence patterns in corpora[D]. School of Computer Science and Engineering, Hebrew university, Jerusalem, 2002.

[6] Winston H Hsu, Lyndon S Kennedy, Shih-Fu Chang. Video search reranking via information bottleneck principle[A]. Proceedings of ACM International Conference on Multimedia[C]. Santa Barbara, CA, USA, 2006. 35 - 44.

[7] N Slonim. The information bottleneck: Theory and Application [D]. The Hebrew University of Jerusalem, Jerusalem, Israel,

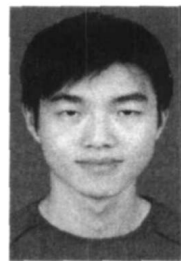
- 2002.
- [8] N Slonim, N Tishby. Agglomerative information bottleneck [A]. Proceedings of Advances in Neural Information Processing Systems (NIPS-2000) [C]. 1999, vol. 12. 617 - 623.
- [9] J Peltonen, J Sinkkonen, S Kaski. Sequential information bottleneck for finite data [A]. Proceedings of 21st International Conference on Machine Learning [C]. Madison, USA, 2004. 647 - 654.
- [10] 朱真峰, 叶阳东, Gang Li. 基于变异的迭代 sIB 算法 [J]. 计算机研究与发展, 2007, 44(11): 1832 - 1838.
Zhu Zhenfeng, Ye Yangdong, Gang Li. Iterative sIB Algorithm Based on Mutation [J]. Journal of Computer Research and Development, 2007, 44(11): 1832 - 1838. (in Chinese)
- [11] S Still, W Bialek. How many clusters? an information-theoretic perspective [J]. Neural Computation, 2004, 16(12): 2483 - 2506.
- [12] Z Y Niu, D H Ji, C L Tan. Document clustering based on cluster validation [A]. Proceedings of 11st International Conference on Information and Knowledge Management [C]. Virginia, USA, 2004. 501 - 506.
- [13] 叶阳东, 刘东, 贾利民, Li Gang. 一种自动确定参数的 sIB 算法 [J]. 计算机学报, 2007, 30(6): 969 - 978.
Ye Yang-Dong, Liu Dong, Jia Li-Min, Li Gang. An sIB Algorithm for Automatically Determining Parameter [J]. Chinese Journal of Computers, 2007, 30(6): 969 - 978. (in Chinese)
- [14] Periklis Andritsos, Panayiotis Tsaparas, René J Miller, Kenneth C Sevcik. LIMBO: Scalable Clustering of Categorical Data [A]. Proceedings of 9th International Conference on Extending Database Technology (EDBT) [C]. March 2004.
- [15] Y Seldin, N Slonim, N Tishby. Information bottleneck for non co-occurrence data [A]. Proceedings of Advances in Neural Information Processing Systems (NIPS-19) [C]. 2006.
- [16] Z Huang. A fast clustering algorithm to cluster very large categorical data sets in data mining [A]. SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery [C]. 1997. 1 - 8.
- [17] Z Huang. Extensions to the k-means algorithm for clustering large data sets with categorical values [J]. Data Mining and Knowledge Discovery, 1998, 2(3): 283 - 304.
- [18] D Cristofor, D Simovici. Finding median partitions using information-theoretical-based genetic algorithms [J]. Journal of Universal Computer Science, 2002, 8(2): 153 - 172.
- [19] C L Blake, C J Merz. UCI repository of machine learning databases [OL], 1998. <http://www.ics.uci.edu/mllearn/MLRepository.html>
- [20] Coenen, F. The LUCS- KDD Discretised/ normalised ARM and CARM Data Library [OL], Department of Computer Science, The University of Liverpool, UK. 2003, <http://www.csc.liv.ac.uk/~frans/KDD/Software/LUCS-KDD-DN/>.

作者简介:



叶阳东 男, 1962 年生于河南潢川. 工学博士、郑州大学信息工程学院教授、博士生导师, 研究方向为知识工程、机器学习、数据库.

E-mail: yeyd@zzu.edu.cn, yeyangd@gmail.com



何锡点 男, 1981 年生于福建永泰. 工学硕士, 研究方向为机器学习.