基于多变异粒子群优化算法的模糊关联规则挖掘

王 飞 缑 锦
(华侨大学计算机科学与技术学院 厦门 361021)

摘 要 针对事务数据库中连续型数值较难划分及粒子群优化算法易陷入局部最优的问题,提出一种用多变异粒子群优化算法进行模糊关联规则提取的框架,即先对连续型数值进行模糊区间划分,再通过多变异粒子群优化算法对划分结果进行模糊关联规则挖掘。分别对模糊划分方法和多变异粒子群优化算法的相关参数及框架等进行说明。在多组实验中进行比较分析,结果表明了该方法的准确性和有效性。

关键词 数据挖掘,粒子群优化,变异算子,多变异算子,关联规则,模糊规则
中图法分类号 TP181 **文献标识码** A

Mining Fuzzy Association Rules Based on Multi-mutation Particle Swarm Optimization Algorithm

WANG Fei GOU Jin
(College of Computer Science and Technology, Huaqiao University, Xiamen 361021, China)

Abstract To deal with the problem that continuous value in the transaction database are difficult to divide and particle swarm optimization algorithm is easy to be troubled with local optimal, this paper proposed a framework about multi-mutation particle swarm optimization algorithm for extracting fuzzy association rules. Firstly, the continuous values are divided into the fuzzy interval. Then using multi-mutation particle swarm optimization algorithm to mine the fuzzy association rules from the division results. This paper described the fuzzy division method and multi-mutation particle swarm optimization algorithm's parameters, framework and others. And it proved the accuracy and efficiency of this method by comparative analysis in several experiments.

Keywords Data mining, Particle swarm optimization, Mutation operator, Multi-mutation operator, Association rules, Fuzzy rules

1 引言

随着计算机技术的迅猛发展,1993年R. Agrawal等人在对市场购物篮问题的调查分析研究中,首次提出用关联规则表达得到的规则知识[1],并在1994年提出了关联规则挖掘算法中最为经典的Apriori算法[2]。关联规则提取的主要目标是发现数据项集之间内含的关联或依赖关系,即从大量积累的数据中找出隐藏的数据模式或知识。

关联规则自被提出以来,就被广泛深入的研究,已进入相对成熟的阶段。由于关联规则需从事务数据中提取,因此不同类型的数据样本规则提取方式和难度也不尽相同,目前研究较为关注的事务数据类型主要有:(1)多分布式大型数据类型[3-5];(2)强时间性数据类型[6,7];(3)高维度数据类型[8,9];(4)多表结构数据类型[10];(5)多层、多关系等数据类型[11-14]。虽然针对这些数据类型已提出较多的规则提取算法,但在规则提取过程中仍有两个问题:一是会产生大量的候选项集;二是要对数据样本进行多次读取操作。针对后一问题,有一些研究将群体智能技术应用于提取关联规则[16-19],以提高规则提取效率,参考这些方法的基本思想,本文提出基于一种改进的粒子群优化算法来提取所需规则。

粒子群优化算法(Particle Swarm Optimization,简称PSO)是对鸟群运动的模拟演化,通过对整个群体的信息共享来使得搜索向最优解靠拢。在1995年Eberhart和Kennedy首次将粒子群优化发展为一种带有启发式的搜索技术[20],PSO算法由于概念简单、易于实现、易于应用等特点,已被各领域广泛应用。粒子群中的各粒子都代表了问题的一个潜在解,可通过调整它们的位置将粒子趋向于最优解。粒子位置的调整依赖于自身的经验以及周围邻居的经验,PSO算法中速度向量包含了这些信息,驱动着整个优化进程。PSO算法中粒子运动更新的计算公式为:

$$V_i^{t+1}=wV_i^t+c_1r_1(Pbest_i^t-x_i^t)+c_2r_2(Gbest^t-x_i^t) \quad (1)$$

$$x_i^{t+1}=x_i^t+V_i^{t+1} \quad (2)$$

式中,x_i^t 和 V_i^{t+1} 表示第 i 个粒子的位置和速度分量,t 表示当前迭代次数,w 表示惯性权值,c_1、c_2 为加速因子,r_1、r_2 为属于(0,1)的随机数,$Pbest_i$ 是第 i 个粒子到目前为止发现的最优位置,即该粒子的认知信息,$Gbest$ 是种群到现在为止发现的最优位置,即社会信息。

关联规则提取通常是对数据样本本身进行操作,而PSO

到稿日期:2012-11-18 返修日期:2013-03-15 本文受国家自然科学基金项目(61103170),厦门市科技计划项目(3502Z20113022)资助。
王 飞(1988-),男,硕士生,主要研究方向为数据挖掘,E-mail:Alfred_wang@foxmail.com;缑 锦(1978-),男,副教授,硕士生导师,主要研究方向为人工智能、知识工程、数据融合等。

算法则是对给定样本数据 D 的随机样本 S 进行操作，以减少对给定样本的读取次数，提高算法的抽取效率。经典 PSO 算法中粒子通过全局最优位置 $Gbest$、自身最优位置 $Pbest_i$ 及惯性权值、加速因子等因素，往一个预期变好的方向移动，但这一特性易使求解过程陷入局部最优，导致丢失一些频繁项集，如图 1(a)所示。本文提出的多变异粒子群优化算法和经典 PSO 算法中一样，粒子会寻找一个期望更优的空间坐标，并往该方向移动，但算法中也存在一些粒子在满足变异条件的情况下，从空间的一个位置突变到其它几个随机位置，这些突变产生的新粒子运动更新过程与其它粒子相同，该方式的粒子运动如图 1(b)所示。

(a) 原粒子群优化算法中粒子运动　(b) 多变异粒子群优化算法中粒子运动

图 1　粒子运动示意图

本文第 2 节对相关概念给出问题描述；第 3 节定义数据处理的模糊划分方法；第 4 节提出一个新的多变异粒子群优化算法并给出具体描述；最后通过实验比较分析该算法的准确性与有效性。

2　问题描述

事务数据库由一系列事务项集构成，表示为 $T=\{t_1,t_2,\cdots,t_n\}$。事务项集 $t_i(i=1,2,\cdots,n)$ 由唯一标识(Tid)和一组事务项组成。假设事务项集 A 中的各事务都在 t_i 中出现，则 A 包含于 t_i，记为 $A\subseteq t_i$。关联规则是形如 $M\rightarrow N$ 的规则，其中 $M\subset t_i$，$N\subset t_i$，且 $M\cap N=\Phi$。而模糊规则是形如 if x is M then y is N 的规则。其中 M、N 用模糊语言值表示，模糊规则也可以简称为 $M\rightarrow N$。

定义 1(模糊关联规则，Fuzzy Association Rules，FAR)　指关联规则中事务项集内的值由模糊语言值表示的规则。

定义 2(支持度、置信度)　一条 $M\rightarrow N$ 规则中，$M\cup N$ 项集在样本数据中出现的次数除以样本总数就是该规则项集相对于该样本的支持度，记为 Support($M\cup N$)。规则的置信度则是 Support($M\cup N$)除以 Support(M)，记为 Confidence($M\rightarrow N$)。

定义 3(强模糊关联规则)　指满足给定的最小支持度(记为 MinSupport)和最小置信度(记为 MinConfidence)约束的模糊关联规则。

不同算法对支持度的定义不尽相同，何等将其定义为相对于所在表的支持度[11]，而本文定义为相对于所在样本的支持度。强模糊关联规则与强关联规则[21]定义相近，均为找出满足给定阈值的规则。

模糊关联规则挖掘(Fuzzy Association Rules Mining，FARM)任务是从事务数据库中找出所有满足支持度和置信度分别大于给定阈值 MinSupport 和 MinConfidence 的强模糊关联规则。

定义 4(多变异算子，Multi-mutation operator，MmO)　指满足一定变异条件的粒子需按两种方式变异：首先其每个方向按一定概率发生变异，生成多个单方向变异粒子；再延多个随机方向对该粒子进行变异，生成一个多方向变异粒子。即满足变异条件的粒子 $r^n=(r_1^n,r_2^n,\cdots,r_i^n,\cdots,r_m^n)$，给定变异概率为 p_0，各维随机变异概率为 $p_k(k=1,2,\cdots,m)$，按两种方式变异：

(1)对 $j\in\{1,2,\cdots,m\}$，若 $p_j>p_0$，则 $r_i^n\xrightarrow{\text{变异}}(r_i^n)'$；

(2) $r^n\xrightarrow{\text{变异}}(r^n)'$。

不同算法定义的变异算子和变异过程各不相同，例如钟等用启发性的变异算子 HMO[22]来加速 PSO 求解，赫等用一种简化的确定变异操作让粒子逃逸[23]，而本文提出多变异粒子群优化算法(Multi-mutation Particle Swarm Optimization，MmPSO)是将经典 PSO 算法与 MmO 相结合，使其具有较好的搜索性能。

3　模糊划分

数据库中往往存储了大量的连续型数据，为挖掘出其中的关联规则需将其进行数据划分。而划分过程中常会遇到边界值划分过硬的问题，可通过隶属度函数来进行区间划分[24]。模糊概念的划分经常使用一组语言变量，如“高”、“中”、“低”，“长”、“中”、“短”等，模糊规则正是用这样的语言变量来描述。语言变量的划分可通过对各维变量计算它们的最小值、最大值以及平均值来确定。

上述语言变量划分过程可通过一个 $k-1$ 输入，单输出的连续属性值变量来说明。假设有 m 组样本采样数据：$(x_1,\cdots,x_{k-1};x_k)_1,(x_1,\cdots,x_{k-1};x_k)_2,\cdots,(x_1,\cdots,x_{k-1};x_k)_m$，其中 $x_1,\cdots,x_{k-1}$ 为输入，x_k 为输出。所有的输入、输出变量均为正值。下面以输入变量 x_j 为例给出计算隶属函数的步骤：

Step 1　计算所有样本 $x_{ij}(i=1,\cdots,m)$ 中最小值

$$a_j=\min(x_{ij}) \tag{3}$$

Step 2　计算所有样本 $x_{ij}(i=1,\cdots,m)$ 中平均值

$$b_j=mean(x_{ij}) \tag{4}$$

Step 3　计算所有样本 $x_{ij}(i=1,\cdots,m)$ 中最大值

$$c_j=\max(x_{ij}) \tag{5}$$

其中，$j=1,2,\cdots,k$ 表示粒子维度，进而确定各维度上的隶属函数形式如下：

$$S=trimf(x,[a_j,a_j,b_j])=\begin{cases}0, & x\leqslant a_j \text{ 或 } b_j\leqslant x\\ \dfrac{x-a_j}{b_j-a_j}, & a_j\leqslant x\leqslant b_j\end{cases} \tag{6}$$

$$M=trimf(x,[a_j,b_j,c_j])=\begin{cases}\dfrac{x-a_j}{b_j-a_j}, & a_j\leqslant x\leqslant b_j\\ 0, & x\leqslant a_j \text{ 或 } c_j\leqslant x\\ \dfrac{x-b_j}{c_j-b_j}, & b_j\leqslant x\leqslant c_j\end{cases} \tag{7}$$

$$B=trimf(x,[b_j,c_j,c_j])=\begin{cases}0, & x\leqslant b_j \text{ 或 } c_j\leqslant x\\ \dfrac{x-b_j}{c_j-b_j}, & b_j\leqslant x\leqslant c_j\end{cases} \tag{8}$$

4　MmPSO 算法

目前国内研究粒子群优化算法挖掘关联规则的相关工作相对较少[16,25]，李等提出了一种 PSO-WMAR 算法挖掘关联规则[17]，程等将粒子群优化算法与遗传算法结合挖掘关联规则中的频繁项集[18]。本文将变异算子与粒子群优化算法结

合设计 MmPSO 算法，以提高关联规则的挖掘效率。

1. 粒子编码

粒子编码中最为直接的编码方式是二进制编码，其优点是易处理、易操作等，但二进制编码导致整个粒子编码很长，降低了算法的效率。PSO 中各粒子位置由不同的维度组成，粒子间的移动通过交换维度等信息实现。有学者采用实数编码[19]，但在转换成模糊规则时仍需按照一定规则取整。MmPSO 算法用正整数编码，取值范围为 $\{0,1,2,\cdots,l_j\}$，$\{0\}$ 表示该属性与其它属性无关联，$\{1,2,\cdots,l_j\}$ 分别对应属性的模糊值。粒子编码如表 1 所列。

表 1　粒子编码

x_1	x_2	…	x_j	…	x_k
$\{0,1,2,\cdots,l_1\}$	$\{0,1,2,\cdots,l_2\}$	…	$\{0,1,2,\cdots,l_j\}$	…	$\{1,2,\cdots,l_k\}$

2. 适应度函数

适应度函数是 PSO 算法与关联规则之间的桥梁，好的适应度函数可以帮助算法更好地挖掘所需规则。PSO 算法中，适应度函数用来评价粒子位置的优劣程度，即适应度越高则位置越好。在关联规则中，置信度显示的是规则的可信程度，而支持度对应规则所在数据集对规则的支持程度。适应度定义如下：

$$fitness_i=\frac{Support_i+Confidence_i}{MinSupport+MinConfidence} \tag{9}$$

3. 粒子更新

根据上述编码方式，对式(2)计算得到的 x_i 中的每一维进行取整，即 $x_{ij}=round(x_{ij})$。但仅采用取整策略，粒子在更新位置后很可能会飞出有效的搜索空间。鉴于此情况，可采用 Robinson 等人提出的吸收墙(absorbing wall)和反射墙(reflecting wall)的策略[26]及陈等人提出的循环墙(cyclic wall)策略[27]进行改进，具体描述如下：

(1)吸收墙指当粒子飞出某个维的边界时，将其拉回搜索空间的边界上，粒子移动后按式(10)进行处理。

$$x_{ij}=\begin{cases}0, & x_{ij}<0,j\neq k\\ l_j, & x_{ij}>l_j\\ 1, & x_{ij}<1,j=k\\ x_{ij}, & \text{其他}\end{cases} \tag{10}$$

(2)反射墙指当粒子飞出某个维的边界时，将其反弹回搜索空间的边界内，粒子移动后按式(11)进行处理。

$$x_{ij}=\begin{cases}(-1)\times x_{ij}, & x_{ij}<0,j\neq k\\ 2\times l_j-x_{ij}+1, & x_{ij}>l_j\\ (-1)\times x_{ij}+1, & x_{ij}<1,j=k\\ x_{ij}, & \text{其他}\end{cases} \tag{11}$$

(3)循环墙指当粒子飞出某个维的边界时，将其从该维度的另一面进入边界内，粒子移动后按式(12)进行处理。

$$x_{ij}=\begin{cases}(x_{ij}+l_j)+1, & x_{ij}<0,j\neq k\\ x_{ij}-l_j, & x_{ij}>l_j\\ x_{ij}+l_j, & x_{ij}<1,j=k\\ x_{ij}, & \text{其他}\end{cases} \tag{12}$$

4. MmO

Ratnaweera 等首先将变异引入到 PSO 算法中[28]，并在多峰函数求解中进行了较多应用[22,29]。不同的算法对变异的处理过程不同，MmO 是对较差粒子进行多次单维变异和一次随机变异的方法，在经典 PSO 算法基础上，增强了全局搜索能力，并避免了粒子由于困于局部最优而导致粒子群萎缩的情况。其具体思路是对所有 $fitness_i<1$ 的粒子，进行如下变异操作。

Step 1　将粒子 x_{ij} 需按如下 2 种方式变异：

第 1 种　多次单维变异

```
for j=1,…,k{
  if Pj< P {// Pj∈(0,1)随机生成,P∈(0,1)用户给定
    随机生成 b∈{0,1,2,…, lj},
    if xj=b then b=b+1;
    根据循环墙策略公式(12),将 b 拉回边界;
    令 x′=x,xj′ =b;
    S′= S′∪x′;
  }
}
```

第 2 种　一次随机多维变异

随机变异生成 x″，$S'=S'\cup x''$；

Step 2 为尽可能避免粒子的重复搜索，对生成的变异粒子集合进行如下处理：

1)if $S'\cap S\neq\Phi$, then $S'=S'-S'\cap S$;

2)if $S'\cap Pbest\neq\Phi$, then $S'=S'-S'\cap Pbest$ 。

上述变异思想可由图 2 的示例说明。假设一个待变异粒子 $a_1a_2a_3\cdots a_n$ 的适应度值为 $fitness=0.5$，各维的变异概率分别为 $p_1=0.04$，$p_2=0.8$，$p_3=0.03$，…，$p_n=0.5$。则该粒子的变异过程如下：首先粒子满足变异条件 $fitness=0.5<1$，且第一维上满足 $p_1=0.04<0.05$，故将 a_1 变异到 a_1'，生成新粒子 $a_1'a_2a_3\cdots a_n$，又第三维满足 $p_3=0.03<0.05$，故生成新粒子 $a_1a_2a_3'\cdots a_n$，而第二维不满足变异条件，故不能产生新的粒子，按此方法得到一组变异粒子，并按变异操作中第 2 种方式生成一个随机多维变异粒子 $a_1''a_2''a_3\cdots a_n''$。两种变异方式生成的变异粒子构成新变异集合 S'。

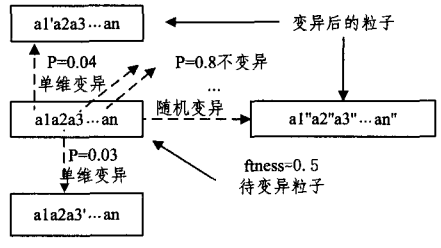


图 2　MmO 实例

通常变异算子对满足一定条件的粒子仅进行单次变异[29]，为提高算法的搜索效率，本文在第 2 步中进行多次单维变异和一次随机多维变异，从而提高粒子逃出局部最优解的几率。

5. MmPSO 算法框架

MmPSO 是基于 PSO 的改进算法，其提取关联规则的基本框架如下：

a. 随机生成粒子群 S，并初始化各参数；

b. 计算各粒子的 $Support_i$、$fitness_i$、$Confidence_i$；

c. 更新种群中各粒子的 $Pbest_i$ 及整个种群的 Gbest；

d. 将 $fitness_i<1$ 的粒子按 MmO 变异操作生成粒子群 S'；

e. 计算 S' 中各粒子的 $Support_i'$、$Confidence_i'$、$fitness_i'$；

f. 更新 S' 粒子群的 $Pbest_i'$ 和 Gbest；

g. $S=S\cup S'$；

h. 将 Pbest 中满足条件的规则加入到规则集中；

i. 消除 S 中的重复粒子，并消除规则集中的重复规则；

j. 判断是否满足终止条件。如满足条件，则算法终止；

k. 根据式(1)和式(2)更新各粒子的速度和位置，再按式(10)将飞

出搜索空间的粒子拉回。跳转到 b。

5 实验

本文通过 3 组实验验证 MmPSO 算法的准确性和有效性。首先，在一般维度和数量的样本中进行验证；然后，验证较高维度下算法的有效性；最后，将其与同类算法在较高维度和较大样本数量环境中进行比较。

5.1 实验 1

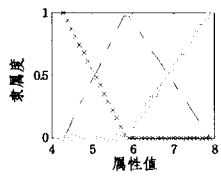
1. 数据编码

本实验通过对鸢尾属植物数据进行规则提取，验证算法准确性和有效性，该数据集来源于 UCI 数据集：archive. icsuci. edu/ml/datasets/Iris，共有 150 组样本数据，其中 Setosa、Versicolor、Verginica 3 类各 50 个样本。样本的各字段及其对应的数据类型如表 2 所列。

表 2 鸢尾属植物数据字段及数据类型

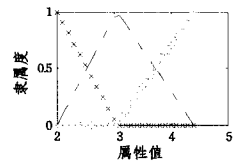
字段	Sepal length	Sepal width	Petal length	Petal width	Class
类型	Float	Float	Float	Float	Char

由于 Class 属性为 Char，且共 3 类，可用 1、2、3 分别编码 Setosa、Versicolor、Verginica。根据本文第 3 节所述的模糊划分方式，对 Sepal length、Sepal width、Petal length、Petal width 这 4 个属性进行模糊划分，S、M、B 分别对应 1、2、3 编码。剩余 4 个属性对应的隶属函数图如图 3 所示。



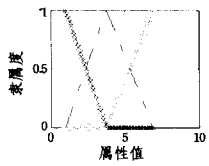
a=4.3，b=5.8367，c=7.9

(a) Sepal length 隶属函数图



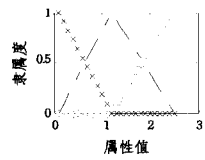
a=2，b=3.054，c=4.4

(b) Sepal width 隶属函数图



a=1，b=3.7587，c=6.9

(c)Petal length 隶属函数图



a=0.1，b=1.1978，c=2.5

(d) Petall width 隶属函数图

图 3 剩余属性的隶属函数图

将样本按上述方式模糊编码后，可得到如表 3 所列的结果。每个样本均被表示成 5 位编码串，每串编码对应一个样本的模糊表示，例如："23111"编码表示为"萼片长度中，萼片宽度大，花瓣长度小，花瓣宽度小的鸢尾花是 Setosa"。

表 3 鸢尾属植物数据模糊划分结果

Sepal length	Sepal width	Petal length	Petal width	Class
2	2	1	1	1
1	2	1	1	1
2	3	1	1	1
		.....		
1	1	2	2	2
2	2	2	2	2
2	1	2	2	2
		.....		
3	2	3	3	3
3	2	2	3	3
2	2	3	3	3

2. 实验参数设置

实验中 MmPSO 算法的相关参数设置如下：

(1)惯性权重 w：控制的是粒子之前的飞行状态对于下一速度的影响程度。w 影响全局和局部的搜索能力，较大则倾向于全局搜索。w 一般定义域为(0,1)，实验中 w 取 0.9。

(2)最大速度 V_{max}：表示粒子在一次位移中的最大速度，该值一般不超过粒子的最大宽度。如果粒子速度过大，会错过全局最优解；如果粒子速度过小，易使粒子陷入局部最优解。这里取粒子的最大宽度 $V_{max}=4$。

(3)加速因子 c_1、c_2：通常取正常数，一般取值在 0～2 之间，两因子间的比值依赖于具体问题。本文中取 $c_1=c_2=2$。

(4)初始种群数 N：50。

(5)迭代次数：30。

3. 实验结果及分析

根据上述参数设定，分别取 MinSupport 为 0.01、0.03、0.05、0.07、0.09、0.11，取 MinConfidence 为 0.5、0.6、0.7、0.8、0.9，进行 30 次实验，分别记录每种组合所挖掘的规则数、平均支持度及平均置信度。实验结果如表 4 所列。

表 4 鸢尾属植物不同 MinSupport 和 MinConfidence 所挖掘的规则数、平均支持度及平均置信度

MinConfidence \ MinSupport		0.01	0.03	0.05	0.07	0.09	0.11
0.5	规则数	102	75	73	56	41	37
	平均置信度/%	92.00	93.41	93.23	92.85	92.69	92.22
	平均支持度/%	11.39	15.351	15.16	17.79	21.35	22.58
0.6	规则数	99	75	73	56	41	37
	平均置信度/%	92.97	93.41	93.23	92.85	92.69	92.22
	平均支持度/%	11.67	14.86	15.16	17.79	21.35	22.58
0.7	规则数	94	72	70	53	39	35
	平均置信度/%	94.29	94.42	94.26	94.19	93.87	93.51
	平均支持度/%	11.67	14.70	15.01	17.74	21.23	22.51
0.8	规则数	79	63	61	46	33	29
	平均置信度/%	97.79	97.23	97.13	97.12	97.36	97.41
	平均支持度/%	11.24	13.71	14.03	16.49	19.86	21.22
0.9	规则数	68	52	50	30	27	24
	平均置信度/%	99.81	99.76	99.75	99.90	99.87	99.85
	平均支持度/%	10.44	13.19	13.56	16.05	19.14	20.28

由于样本每组类别为 50 个，因此规则的最大支持度不超过 50/150=33.33%，最大置信度不超过 50/50=100%。从表 4 可知：

(1)每组实验结果均有较高的支持度和置信度。当最小置信度仅有 0.5 时，其平均置信度仍在 90%以上。同时平均支持度都在 10%以上，即平均每条规则有 15 条以上的数据支持。

(2)最小置信度影响平均置信度和规则提取数，最小支持度影响平均支持度和规则提取数。当最小支持度为 0.01 时，随着最小置信度升高，提取的规则数目减少，平均置信度增加。当最小置信度为 0.5 时，随着最小支持度升高，提取规则数目减少，平均支持度增加。

(3)最小支持度不直接影响平均置信度，最小置信度不直接影响平均支持度。当最小置信度为 0.5 时，随着最小支持度升高，平均置信度发生波动，但是稳定在一定的值附近。当最小支持度为 0.01 时，平均支持度也同样波动。

(4)关联规则随支持度和置信度稀疏分布。当最小支持

度在 0.05 和 0.07 时，挖掘到的规则数目相差较大，同样的置信度在 0.7 和 0.8 时，挖掘的规则也有较大的数量差距。

MmPSO 算法所挖掘的规则都是模糊编码表示，需要对编码进行释意才可以知道所提取的规则含义。提取的部分关联规则释意如下：

0001=>1 (support:33.33% confidence:100%)

瓣宽小=> Setosa

3230=>3 (support:8% confidence:100%)

萼长大;萼宽中;瓣长大=> Verginica

2210=>1 (support:8% confidence:100%)

萼长中;萼宽中;瓣长小 => Setosa

2122=>2 (support:6.67% confidence:90.9091%)

萼长中;萼宽小;瓣长中;瓣宽中=> Versicolor

4. 实验比较

GA、PSO、MmPSO 算法的比较实验中相同参数设置如表 5 所列。

表 5　实验 1 相同参数设置

参数名	MinSupport	MinConfidence	迭代次数	初始种群
参数取值	0.03	0.9	300	(50,60)

其它条件及参数设置如下：

1)PSO 算法参数设定与 MmPSO 算法相同；

2)GA 算法中的交叉操作：适应度前 20%的子代进行随机多次单点交叉取最优后代；

3)GA 算法中的变异操作：变异概率分段给出，适应度前 20%的 $P=0.01$，中间 70%的 $P=0.02$，最后 10%的 $P=0.04$。

根据实验设定，将 PSO、GA 和 MmPSO 算法分别对样本集进行模糊规则提取，记录各算法在每次迭代中挖掘到的规则总数。由于实验数据规模较小，可将所有满足条件的 FAR 挖掘出来，大于阈值 MinSupport 和 MinConfidence 的规则有 52 条。实验结果由图 4 可见，GA 算法约迭代 300 次内可挖掘到 52 条规则，PSO 算法约迭代 100 次内即可挖掘 52 条规则；而 MmPSO 算法只需 15 次左右，便可以挖掘出这 52 条规则。可见 MmO 的加入，提高了算法的挖掘效率，同时也保证了算法的准确性。

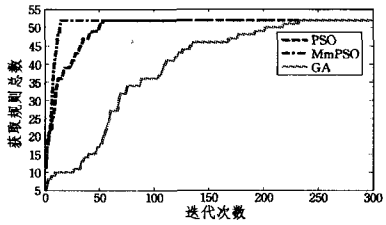


图 4　实验 1 中各算法提取规则总量

5.2　实验 2

1. 实验数据及预处理

第 5.1 节的实验说明了 MmPSO 算法的有效性和准确性，下面提高实验数据的维度，验证 MmPSO 算法在稍大维度中同样有效。实验数据为 UCI 数据集(地址：archive.ics.uci.edu/ml/datasets/Glass+Identification)中的玻璃分类样本。本文对这些样本稍做处理，去掉不需要的属性，同时因源样本中的玻璃种类不全，取其中 3 种类别数量较大的进行挖掘。最终共取 175 条数据，9 个属性，每条样本都记录了 8 个玻璃的构成元素数据和所对应的玻璃种类，数据的各字段及其对应类型如表 6 所列。

表 6　玻璃类别数据字段及数据类型

字段	类型	字段	类型	字段	类型
Na	Float	Si	Float	Fe	Float
Mg	Float	K	Float	Ca	Float
Al	Float	Ba	Float	Type	Int

3 种玻璃类别分别为：

(1)building windows float processed；

(2)building windows non float processed；

(3)headlamps。

根据第 3 节中所述的模糊划分方法，对这 9 个字段分别进行模糊编码，下面不再对样本编码结果及挖掘到的规则内容进行赘述。

2. 实验及结果

GA、PSO、MmPSO 算法的比较实验中相同参数设置如表 7 所列。因种群维度增大近一倍，样本中的排列组合数达到 19 万条以上，故将初始随机种群数修改为 $N\in(500,600)$，MinConfidence 设定为 0.8，其它参数设置与实验 1 相同。

表 7　实验 2 相同参数设置

参数	MinSupport	MinConfidence	迭代次数	初始种群数
取值	0.03	0.8	300	(500,600)

根据以上参数设置，分别用 GA、PSO、MmPSO 算法对玻璃分类样本进行规则提取。实验结果如表 8 所列，MmPSO 经 300 次迭代后，提取的关联规则总条数达 110 多条，远远超过 GA 和 PSO 挖掘到的约 40 条规则，同时保持了与 GA 和 PSO 相近的平均支持度和平均置信度，即 MmPSO 算法保持了与其它群体智能算法相近的准确性。

表 8　玻璃分类样本各算法挖掘规则结果

算法	MmPSO	PSO	GA
挖掘规则数	约 110 多条	约 40 条	约 40 条
平均支持度	5.03%	5.8%	4.9%
平均置信度	96.38%	93.08%	97.84%

图 5 显示了这 3 种算法每次迭代所提取规则总数间的关系。由图 5 可知：

1)MmPSO 每次挖掘的规则数目都高于 GA 和 PSO 算法；

2)GA 起步较晚，需进行多次迭代，然后逐渐地提取出所要规则。PSO 则易陷入局部最优，后期每次迭代所提取规则数明显减少，而 MmPSO 则在较早阶段提取出较多规则，同时保持了较好的规则提取优势。

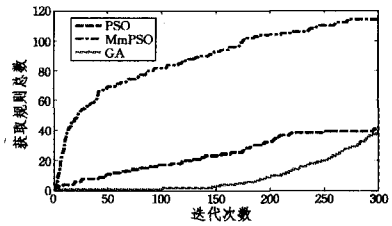


图 5　实验 2 中各算法提取规则总量

实验结果表明，MmPSO 算法在较高维度数据集环境中，与其他同类挖掘算法相比，其效率和准确性有一定优势。

5.3　实验 3

1. 实验数据及预处理

第 5.2 节的实验说明了 MmPSO 算法在增加维度后仍然有良好的挖掘效率和准确性，下面再在实验 2 的基础上增加实验的数据量，验证 MmPSO 算法在稍大数据量和稍大维度中同样有效。实验数据来自 UCI 数据集（地址：archive. ics. uci. edu/ml/datasets/Concrete+Compressive+Strength）中的水泥抗压强度样本，该样本共 1030 条数据，9 个属性字段，即 8 个影响抗压强度因素及所对应的水泥抗压强度值，数据的各字段及其对应的数据类型如表 9 所列。根据本文第 3 节所述的模糊划分方式，将这 9 个字段分别进行模糊编码。

表 9　水泥抗压强度数据字段及数据类型

字段	类型	字段	类型
Cement	Float	Water	Float
Blast Furnace Slag	Float	Superplasticizer	Float
Fly Ash	Float	Coarse Aggregate	Float
Fine Aggregate	Float	Age	Float
Concrete compressive strength	Float		

2. 实验及结果

GA、PSO、MmPSO 算法比较实验中参数与实验 2 的相同，分别对水泥抗压强度样本进行规则提取。实验结果如表 10 所列，其中可以看到 MmPSO 在经 300 次迭代后，提取的规则数目约 240 条，远超过其它两种算法所提取的数目，同时 MmPSO 算法所挖掘的规则仍同样保持了与 GA 和 PSO 算法相近的平均支持度和平均置信度。

表 10　水泥抗压强度样本各算法挖掘规则结果

算法	MmPSO	PSO	GA
挖掘规则数	约 240 条	约 100 条	约 110 条
平均支持度	9.25%	9.72%	8.91%
平均置信度	93.29%	89.89%	91.47%

图 6 显示了这 3 种算法每次迭代所提取规则总数间的关系，并同样可以得到图 5 类似的结论。由实验可知，在较大数据量和较高数据维度环境下，MmPSO 算法仍能保持较好的挖掘效率。

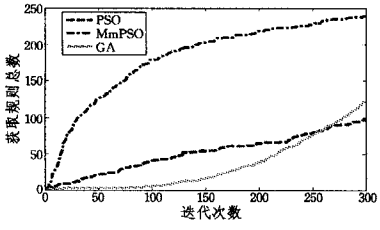


图 6　实验 3 中各算法提取规则总量

结束语　本文针对事务数据库中连续型数值较难划分的问题，用隶属函数进行模糊区间划分，并提出一种新的多变异粒子群优化算法用于对给定事务数据库的模糊划分结果进行模糊关联规则挖掘。通过加入多变异算子，增大粒子逃出局部最优解的几率，扩大算法的搜索范围。实验结果表明，MmPSO 算法在不同维度和样本数量的情况下，与其它基于群体智能技术的关联规则挖掘算法相比，具有更好的准确性和规则提取能力。

参 考 文 献

[1] Agrawal R, Imielinski T, Swami A. Mining association rules between sets of items in large databases[C]// Buneman P, Jajodia S, eds. Proc of the 1996 ACM SIGMOD Int'l Conf. on Management of Data. New York: ACM Pressa, 1993: 207-216

[2] Agrawal R, Srikant R. Fast algorithms for mining association roles[C]// Proc. of the Int'l Conf. on Very Large Data Bases (VLDB). Santiago, 1994: 487-499

[3] Agrawal R , Sharfer J . Parallel Mining of Association Rules [J]. IEEE Transactions on Knowledge and Data Engineering, 1996, 8(6): 962-969

[4] Park J S, Chen M, Yu P S. Efficient Parallel Data Mining for Mining Association Rules[C]// ACM International Conference on Information and Knowledge Management. 1995: 31-36

[5] 李玲娟，张敏. 云计算环境下关联规则挖掘算法的研究[J]. 计算机技术与发展，2011，21(2)：43-46，50

[6] Wang Xiao-li, Mabu Shin-go, Zhou Hui-yu, et al. Time Related Association Rules Mining with Attributes Accumulation Mechanism Applied to Large-scale Traffic System[C]// SICE Annual Conference 2010. August 2010: 2637-2641

[7] Lan Guo-cheng, Chen Chun-Hao, Hong Tzung-pei, et al. A Fuzzy Approach for Mining General Temporal Association Rules in a Publication Database[C]// 2011 11th International Conference on Hybrid Intelligent Systems (HIS). 2011: 611-615

[8] Prasanna K, Seetha M. Mining High Dimensional Association Rules by Generating Large Frequent K-Dimension Set [C]// 2012 International Conference on Data Science & Engineering (ICDSE). 2012: 58-63

[9] 朱玉，张虹，孔令东. 基于人工免疫的多维关联规则挖掘及其应用研究[J]. 计算机科学，2009，36(8)：1104-1106

[10] Wang Shyue-liang, Hong Tzung-pei, Tsai Yu-chuan, et al. Multitable Association Rules Hiding[C]// 2010 10th International Conference on Intelligent Systems Design and Applications. 2010: 1298-1302

[11] 毛宇星，陈彤兵，施伯乐. 一种高效的多层和概化关联规则挖掘方法[J]. 软件学报，2011，22(12)：2965-2980

[12] 陈申燕，曹曼. 多层关联规则挖掘算法的研究及应用[J]. 计算机工程与设计，2010，31(4)：885-888

[13] 何军，刘红岩，杜小勇. 挖掘多关系关联规则[J]. 软件学报，2007，18(11)：2752-2765

[14] 崔建，李强，杨龙坡. 基于垂直数据分布的大型稠密数据库快速关联规则挖掘算法[J]. 计算机科学，2011，38(4)：216-220

[15] 向卓元，李颖. 基于改进型遗传算法的关联规则挖掘方法[C]// Proceedings of 2010 International Conference on Management Science and Engineering(MSE 2010). Volume 4, 2010: 265-267

[16] 刘丛林，张忠林，曾庆飞. PSO 算法在关联规则挖掘中的应用[J]. 兰州交通大学学报，2010，29(3)：96-99

[17] 李呈林，陈水利. 基于 PSO 的加权关联规则挖掘算法[J]. 集美大学学报：自然科学版，2007，12(1)：52-58

[18] 程灿，梁军，张超英. 基于遗传粒子群算法的频繁项集挖掘算法[J]. 现代计算机：专业版，2009，1：15-18

[19] 王晓敏. 基于微粒群算法的关联规则挖掘方法及应用[D]. 济南：山东师范大学，2010

[20] Kennedy J, Eberhart R C. Panicle swarm optimization [C]// Proc of the IEEE International Conference on Neural Networks. Piscataway, NJ: IEEE Service Center, 1995: 1942-1948

[21] 程舒通，徐从富. 关联规则挖掘技术研究进展[J]. 计算机应用研究，2009，26(9)：3210-3213

[22] 钟文亮，王惠森，张军，等. 带启发性变异的粒子群优化算法[J]. 计算机工程与设计，2008，29(13)：3042-3046

[23] 赫然,王永吉,王青,等. 一种改进的自适应逃逸微粒群算法及实验分析[J]. 软件学报,2005,16(12):2036-2044
[24] 邹晓峰,陆建江,宋自林. 基于模糊分类关联规则的分类系统[J]. 计算机研究与发展,2003,40(5):651-656
[25] 李锦泽,叶晓俊. 关联规则挖掘算法研究现状[C]//第18届计算机技术与应用学术会议(CACIS). 2007:216-220
[26] Robinson J, Rahmat-Samii Y. Particle swarm optimization in electroma gnettics[J]. IEEE Transactions on Knowledge and Data Engineering,2004,52(2):397-407
[27] 陈翔,顾庆,王子元,等. 一种基于粒子群优化的成对组合测试算法框架[J]. 软件学报,2011,22 (12):2879-2893
[28] Ratnaweera A, Halgamuge S K, Watson H C. Self-organizing hierarchical panicle swarm optimizer with time-varying acceleration coefficients[J]. IEEE Trans Evol Comput,2004,8(3):240-255
[29] 蔡昭权,黄翰. 自适应变异综合学习粒子群优化算法[J]. 计算机工程,2009,35(7):170-171,202

(上接第208页)

1)在建立参数名Trie和参数值Trie后,首先对输入的网页文本数据预处理,根据逗号、句号等分隔符将文本分成长字符串;

2)通过参数名Trie和参数值Trie分别对输入的长字符串进行词语识别;

3)循环处理每个长字符串,对提取获得的规则字符串进行约束性判断,如果不符合判断条件,则丢弃;否则作为提取串保存至数据库作进一步分析。

系统可以很方便地扩展成过滤系统。例如过滤垃圾邮件,只要将垃圾邮件中经常出现的字符串作为规则即可。

通过设计实验,针对本文的提取方法进行验证,实验1采用的两段输入文本如下:

输入文本1:单相控制隔离变压器符合EN61558标准接触保护(VBG4)绝缘等级T40变压器MCT输入电压230/400V AC输出电压24V绝缘等级T 40/B

提取串:输入电压:230/400V

输出电压:24V

绝缘等级:T 40/B

输入文本2:此款上衣胸围86cm肩宽38cm适合平时穿160/84的mm,颜色黑色。

提取串:胸围:86cm

肩宽:38cm

颜色:黑色

实验1中根据所需提取串的特征,所建立的参数名表、参数值表以及约束条件构建如下:

参数名表(输入电压,输出电压,绝缘等级,胸围,肩宽,颜色)

参数值表(v,b,cm,黑色)

约束条件(输入电压:v),(输出电压:v),(绝缘等级:B),(胸围:cm),(肩宽:cm),(颜色:黑色)

实验1的运行结果表明,本方法能够正确、准确地提取所需要的提取串。

实验2通过加大输入文本的数据量,来测试提取串的正确率和召回率,所获取的实验结果数据如表2所列。

表2 提取串的准确率

类型	测试数目	正确率	召回率
品牌词	1000	95.7%	93.1%
类别词	10000	96.3%	94.6%
服务项目	10000	97.1%	95.4%

通过表2的数据可以看出,本方法能够达到的正确率和召回率均满足B2B商务搜索引擎的需求。

结束语 本文提出了一种基于双数组Trie树的规则串提取方法,用于提取B2B垂直搜索引擎中产品规格的信息。通过对B2B系统中的参数名、参数值等规则串构建规则库,并生成双数组Trie树,实现在一次扫描过程中获取规则串;通过规则库中的约束条件判断,对候选提取串进行过滤,以提高提取准确率;在双数组Trie存储方面,优先处理分支结点最多的子树,以提高存储效率。实验表明,该方法能够降低传统规则串查找的算法复杂度,查找规则串的时间复杂度是O(n)。提取算法已经应用到实践中,能够提高网站的数据质量。本文的进一步研究内容包括:对参数值的进一步约束以及约束性自动判断方面。

参 考 文 献

[1] Curran K, Glinchey J M. Vertical Search Engines [J]. ITB Journal,2008,16:22-28
[2] 雷育生. 基于垂直网站的网络信息支持系统研究[J]. 计算机应用研究,2005,7:105-107
[3] Aoe J. An Efficient Digital Search Algorithm by Using a Double-Array Structure [J]. IEEE Transactions on Software Engineering,1989,15(9):1066-1077
[4] Aoe J, Morimoto K, Sato T. An Efficient Implementation of Trie Structures [J]. Software Practice and Experience,1992,22(9):695-721
[5] Karoonboonyanan T. An Implementation of Double-Array Trie [OL]. http://linux.Thai.net/thep/datrie/datrie.html,2003
[6] 王思力,张华平,王斌. 双数组Trie数算法优化及应用研究[J]. 中文信息学报:人工智能及识别技术,2006,20(5):24-30
[7] 赵欢,朱红权. 基于双数组Trie数中文分词研究[J]. 湖南大学学报,2009,36(5):77-80
[8] 刘燕兵,刘萍,谭建龙,等. 基于存储优化的多模式串匹配算法[J]. 计算机研究与发展,2009,46(10):1768-1776
[9] 刘群,张华平,俞鸿魁,等. 基于层次隐马模型的汉语语法分析[J]. 计算机研究与发展,2004,41(8):1421-1429
[10] Dorji T C, Atlam E-S. New methods for compression of MP double array by compact management of suffixes[J]. Information Processing & Management,2010,46(5):502-513
[11] Schubert P, Legner C. CAKES-NEGO: Causal knowledge-based expert system for B2B negotiation[J]. Expert Systems with Applications,2011,35(1):459-471
[12] Rosenzweig E D, Timothy M. Through the service operations strategy looking glass: Influence of industrial sector, ownership, and service offerings on B2B e-marketplace failures[J]. Journal of Operations Management,2011,29(1):33-48