# 一种有效的用于数据挖掘的动态概念聚类算法*

郭建生, 赵 奕, 施鹏飞

(上海交通大学 图像处理与模式识别研究所,上海 200030)

E mail: jsguo@jcinfo.com.cn

**摘要**: 概念聚类适用于领域知识不完整或领域知识缺乏时的数据挖掘任务.定义了一种基于语义的距离判定函数,结合领域知识对连续属性值进行概念化处理,对于用分类属性和数值属性混合描述数据对象的情况,提出了一种动态概念聚类算法 DDCA(domain-based dynamic clustering algorithm).该算法能够自动确定聚类数目,依据聚类内部属性值的频繁程度修正聚类中心,通过概念归纳处理,用概念合取表达式解释聚类输出.研究表明,基于语义距离判定函数和基于领域知识的动态概念聚类的算法 DDCA 是有效的.

**关键词**: 数据挖掘;动态概念聚类;语义距离;领域知识

**中图法分类号**: TP18,TP311 **文献标识码**: A

数据挖掘(data mining)为知识发现提供手段,可以从巨量的数据集合中抽取隐含的、先前未知的、对决策有潜在价值的规则[1].当挖掘任务面临缺少领域知识或领域知识不完整的数据集合时,采用聚类分析技术,可以将无标识数据对象自动划分为不同的类,并且可以不受人的先验知识的约束和干扰,从而获取属于数据集合中原本存在的信息.机器学习领域里被称为概念聚类的分析技术,是用描述对象的一组概念取值复合表达式将数据划分为不同的类,而不是基于几何距离来实现数据对象之间的相似性度量.概念聚类的特点在于能够对输出的不同类确定其属性特征的覆盖(称作外延),并对聚类结果给予概念解释(称作内涵)[2].根据对概念属性范化和特化(generalization/specification)处理的程度不同,可以得到概念的多个层次描述.另外,概念聚类还适用于处理增量式数据挖掘,当有大量新的数据加入时,不需要对原有的数据重新进行聚类处理.

利用概念聚类的数据分析算法和应用已有大量的研究.概念聚类系统 CLUSTER/2[2]可以将所观察到的对象按预先选定的概念集实现可解释的分类,属于示例学习系统.CLUSTER/2 追求精确的描述,其学习过程需要反复迭代,不适用于处理大数据量的数据挖掘.CLARANS[3]对于大数据的样本采用基于随机抽样技术来确定可能的聚类,以减少数据的处理量.显然,该方法的信息丢失也是不能确定的量.BIRCH 算法[4]采用一种聚类特征树 CFT 技术,旨在寻求表达亚聚类(subcluster)在内存中处理的方法,较好地改善了算法的时间复杂性,但却难以处理数据噪音.DBSCAN[5]用测定样本数据在多维空间上的分布密度来实现聚类,可以处理噪音数据.但 DBSCAN 难以处理介于两个密度中心邻域边界上对象的类属.COBWEB[6]被认为是一种增量式概念形成系统,可以通过对新样本数据的聚类计算,实现对前一次聚类结果的自动修正.Z. Huang 等人[7]针对现实世界里的数据对象往往包含有数值的和非数值的两类属性混合的情况,研究了一种 $k$-原形($k$-

* 收稿日期: 1999-07-27; 修改日期: 2000-02-01

基金项目: 国家自然科学基金资助项目(69835010)

作者简介: 郭建生(1953—),男,河南郑州人,在职博士生,高级工程师,主要研究领域为期货交易计算机应用,金融数据挖掘;赵奕(1972—),女,江苏锡山人,博士生,主要研究领域为模式识别和智能系统,数据挖掘;施鹏飞(1940—),男,上海人,教授,博士生导师,主要研究领域为图像分析,模式识别,智能系统.

prototypes)混合属性聚类算法,是在 $k$-means 算法的距离公式中附加了符号比较计算项,解决了混合有非数值属性的聚类问题.但初始聚类个数 $k$ 是随机确定的,输出结果受 $k$ 值选择和数据排列顺序的影响,且不能有效地解释聚类输出.SBAC(similarity based agglomerative system)系统[8]参照基于生物学分类方法提出一种可以统一处理数值属性和概念属性的相似度量的框架.利用差别矩阵实现聚类,当数据量较大时,系统的空间开销太大.

本文结合对"证券投资客户行为规律"的研究,提出一种结合数据库操作的处理混合属性的概念聚类挖掘算法:结合领域知识对连续属性值进行概念化处理,通过设定相似性阈值自动确定聚类划分的数目,利用属性值的语义距离[5]判定数据对象的相似程度,依据属性不同取值的频繁程度实现聚类中心的动态调整.通过对聚类输出的概念范化处理,得到相应聚类的覆盖描述.聚类的输出可以方便地用于生成层次概念树,输出带有解释的特性判别规则.实验表明,本文提出的算法 DDCA(domain-based dynamic clustering algorithm)是有效的.

## 1 概念属性及其相似性

从现实世界里获取的对象往往是用数值型属性和符号型属性混合描述的.需要研究将数值类属性值进行概念化处理的方法和定义基于语义距离的概念属性相似性的测度.距离函数用来确定数据对象之间的相似关系,可以认为是领域知识的一种表达方式[3].

### 1.1 概念属性特征及其分类

给定一个有限的 $n$ 维的离散向量空间 $U=D_1\times D_2\times\ldots\times D_m$,其中 $D_j$ 是有限的符号集($j=1,2,\ldots,m$).称 $|U|$ 为集合 $U$ 的元素的个数,表示了集合 $U$ 的尺度.

**定义 1.** $\forall u\in U$ 称为 $U$ 的实例,是以符号的向量形式描述的对象 $u=\langle a_1,a_2,\ldots,a_m\rangle$,其中属性值 $a_j\in A_j\subseteq D_j$,$j=1,2,\ldots,m$.在数据库表 $T$ 中记为元组 $S_i$,$i=1,2,\ldots,n$.

**定义 2.** 具有相同描述结构的对象集合构成向量空间的子集 $T\subset U$.$T$ 库有限个元组 $S_i$ 的集合表示,称为数据库表 $T=\{S_1,S_2,\ldots,S_n\}$.表 $T$ 被认为是论域 $U$ 中感兴趣的子集.

设待分类的 $n$ 个对象(称为元组 $S_i$)集合组成关系表 $T=\{S_1,S_2,\ldots,S_n\}$.元组 $S_i$ 由 $m$ 个属性 $A_j$ 描述,$j=1,2,\ldots,m$.属性 $A_j$ 的值域取自有限个可以互相区别的符号(称为概念)组成的域.用 $Dom(A_j)=\{a_1,a_2,\ldots,a_m\}$ 表示属性 $A_j$ 的值域.$Dom(A_j)$ 的尺度记为 $|Dom(A_j)|$;属性域中元素之间的结构关系因描述对象的不同而不同,可以分为线性(linear)的、范畴(categorical)的和层次结构(hierarchy)的.其值域分别是有序的、无序的和图序的集合,分别记为 $D_l$,$D_c$ 和 $D_h$.

$D_l$ 表示有序概念域,$A_j\subseteq D_l$ 表示属性值取自有序概念集合.如客户存款额 $D_l=\{$'无'、'少'、'较少'、'中等'、'较多'、'多'、'特别多'、'极多'$\}$;

$D_c$ 表示无序概念域,$A_j\subseteq D_c$ 表示属性值取自无序概念集合.如股票的市场表现 $D_c=\{$'领涨股'、'超跌股'、'庄家股'、'投机股'、'消息股'$\}$;

$D_h$ 表示结构化的概念域,属性一般是树状层次的.父结点的概念是对其子结点概念的归纳和概括.对于每一个概念层,属性也可以分为有序的和无序的两种.每个层次中节点的值分别形成有序或无序的集合.如客户在一段时间内的股票交易量是有序的层次结构,如图 1 所示.而上市的证券交易品种可以认为是无序层次结构,如图 2 所示.属性从数值到概念的变换反映了领域知识,表达了用户的兴趣.

### 1.2 语义距离

设感兴趣的数据对象满足定义 2,元组 $S_i\in T(i=1,2,\ldots,n)$ 之间的相似性以语义距离最小来

度量:

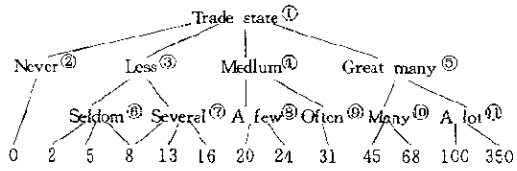



Fig. 1 Linear hicrarchy structure
图1 有序层次结构

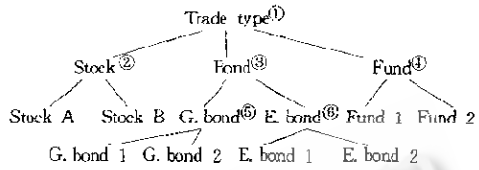



Fig. 2 Categorical hierarchy structure
图2 无序层次结构

(1) 符号属性的语义距离:

$$d_c(a_{ik},a_{jk})=\begin{cases}0 & a_{ik}=a_{jk}\\ 1 & a_{ik}\neq a_{jk}\end{cases},\quad i\neq j. \tag{1}$$

$d_c(a_{ik},a_{jk})$表示当不同的两个元组 $S_i$ 和 $S_j$ 的第 $k$ 个属性值取自对应的符号(范畴)概念域 $D_k$ 里元素 $a_{ik},a_{jk}\in A_k(i\neq j)$时的语义距离. 当两个属性值相同时,$d_c(a_{ik},a_{jk})$为 0,反之为 1.

(2) 有序属性值的语义距离:

$$d_l(a_{ik},a_{jk})=|a_{ik}-a_{jk}|. \tag{2}$$

$d_l(a_{ik},a_{jk})$代表不同的两个元组 $S_i$ 和 $S_j$ 的第 $k$ 个属性值取自对应的有序概念集合 $D_k$ 里的元素 $a_{ik}$,$a_{jk}\in A_k(i\neq j)$时的语义距离. 例如:存款额 Deposit$=\{a_1,a_2,a_3,a_4,a_5,a_6\}=\{$'少','较少','中等','较多','多','极多'$\}$,语义距离 $d_l(a_1,a_3)=2$,即'少'和'中等'的语义距离为 2,语义距离 $d_l(a_1,a_1)=0$,表示'较少'和'较少'的语义距离是 0.

(3) 两个元组 $S_i,S_j\in T$ 之间的语义距离 $d(S_i,S_j)$定义为

$$d(S_i,S_j)=\mathrm{Dis\_o}(S_i,S_j)+\mathrm{Dis\_c}(S_i,S_j)=\sum_{k=1}^{k_o}\beta_k|a_{ik}-a_{jk}|+\sum_{k=1}^{k_c}\beta_k d_c(a_{ik},a_{jk}). \tag{3}$$

$\mathrm{Dis_o}(S_i,S_j)=\sum_{k=1}^{k_o}\beta_k|a_{ik}-a_{jk}|$定义为 $S_i$ 和 $S_j$ 之间所有有序属性值的语义距离的累加和,$k_o$ 是元组 $S$ 的有序属性的个数. $\mathrm{Dis_c}(S_i,S_j)=\sum_{k=1}^{k_c}\beta_k d_c(a_{ik},a_{jk})$定义为 $S_i$ 和 $S_j$ 之间所有无序属性值的语义距离累加和,无序属性的个数为 $k_c$. 其中 $\beta_k$ 是权重因子向量.

### 1.3 聚类判定准则 $J$

设 $\boldsymbol{Y}=\{Y_1,Y_2,\ldots,Y_p\}$为 $p$ 个聚类中心集合.$Y_i=(y_1,y_2,\ldots,y_m)$是由 $k_o$ 个有序属性字段和 $k_c$ 个无序属性字段组成的有 $m(=k_o+k_c)$个向量表示的聚类中心,与待分类的元组 $S_i\in T$ 有相同的属性描述结构. 用聚类判定准则 $\boldsymbol{J}$ 表示 $S_i$ 到 $\boldsymbol{Y}$ 的 $p$ 个聚类中心的语义距离的最小值:

$$\boldsymbol{J}=\min\sum_{l=1}^{p}d(S_i,Y_l),\ i=1,2,\ldots,n, \tag{4}$$

$d(S_i,Y_l)$是元组 $S_i$ 与聚类中心 $Y_l$ 之间各属性语义距离的累加和,即

$$d(S_i,Y_l)=\sum_{k=1}^{k_o}\beta_k|a_{ik}-y_{lk}|+\sum_{k=k_o+1}^{k_c}\beta_k d_c(a_{ik}-y_{lk}). \tag{5}$$

$\beta_k$ 是权重矢量,$k=1,2,\ldots,m$ 是元组属性的个数. 用于调整各属性对聚类的贡献率,可以由领域知识确定.

### 1.4 调整聚类中心的判决函数 $J_l$

设 $A_j$ 是表 $T$ 的第 $j$ 个概念属性的值域的集合,$p(A_j=a_{jk}|l)$是第 $l$ 个聚类划分中第 $j$ 个属性取值 $a_{jk}$的概率.调整聚类中心的判决函数 $J_l$ 用下式表达:

$$J_l=\{(\max\ p(A_j=a_{jk}|l))|_{j=1}^{m}\},\ l=1,2,\ldots,p. \tag{6}$$

即只要求得 $J_l$,便可以确定第 $l$ 个聚类的各属性字段($k=1,2,\ldots,m$)上出现频繁程度最大的概念值集合构成聚类的中心 $Y_l$,$l=1,2,\ldots,p$.

### 1.5 相似性度量

由上述聚类距离判决函数 $\boldsymbol{J}$ 可以分别计算元组 $S_i\in T$ 到各聚类中心 $Y_l$ 的语义距离.两个元组的语义距离越小,表示元组之间越相似.对于所有待分类的元组 $S_i$,利用公式(5)计算与所有 $p$ 个聚类中心的相似程度.当判定得到最小距离时,便找到该元组的类属,将 $S_i$ 划分到具有最小语义距离值的那个类.

## 2 动态聚类算法

采用动态概念聚类算法实现对"证券投资客户行为规律"的研究,可以利用上述语义距离的定义判定数据库中元组 $S_i$(客户行为的特征描述)之间的相似关系,将最相似的那些元组划分为同类.用调整聚类中心的函数动态地确定聚类划分的聚类中心 $\boldsymbol{Y}=\{Y_1,Y_2,\ldots,Y_p\}$,再利用定义的语义距离划分数据库表 $T$ 中的所有元组到最近的聚类中心所代表的那个类中.任何一个聚类划分中发生元组的增加或减少,都相应地调整其聚类中心的概念取值.这个过程直到系统趋于稳定为止.对于大数据集合,动态聚类挖掘的输出,还需要进行归纳处理.

### 2.1 数值属性值的概念化分段

为了能解释挖掘输出结果和有效缩减聚类空间,需要将数据库中的连续数值属性值作概念划分,以形成有序的概念值.数值属性值的分段可以根据数据分布形态选择处理方法.本文利用一种改进的基于数据分布密度的方法,将不连续分布的数值属性值映射为等价的概念值算法 CGA (conceptual generalization algorithm):考虑数据库表 $T$ 中有 $m$ 个属性 $A_j(j=1,2,\ldots,m)$,其中部分 $A_j$ 在实数域上取值.算法 CGA 将指定属性 $A_j$ 的取值映射为与原属性值等价的概念取值.

**算法 2.1.** CGA 可以简要描述如下:

(1) 对表 $T$ 进行抽样:(当表 $T$ 很大时,有利于减少算法的时间复杂性)得到表 $T_s\subseteq T$;抽样应考虑样本的随机性和数量;

(2) 对于需要处理的属性 $A_j$:依据领域知识指定期望得到的概念个数 $G$ 和精度因子 $\alpha$($\alpha$ 一般取 5~10),确定分割 interval=(upmost-lowest)/($G\times\alpha$);将属性 $A_j$ 的值域分割为 $n$($=G\times\alpha$)个小区间 $\text{seg}_k$;分别统计落入各区间里的数据个数 $s\_\text{count}_k$;

(3) 求概念分段:按照顺序依次累加 $s\_\text{count}_k$ 的值到 $\text{sum}_j$,每当 $\text{sum}_j$ 接近于指定的阈值 total_cnt/$G$(total_cnt=$|T_s|$)时,就产生一个概念分段(下限为 $\text{sum}_j$ 的第 1 个 $\text{seg}_k$ 的下限,上限为最后一个 $\text{seg}_k$ 的上限).将这个概念分段顺序存入概念表 $T_c$;直到将属性 $A_j$ 取值的所有统计区间 $s\_\text{count}_k$ 处理完毕为止.

对于数据库表中的符号属性,其概念取值的值域一般由领域专家或知识工程师直接给出.在处理数值属性包含有<0 的分布时,需要考虑分段的不对称性和零概念的定义.

### 2.2 动态概念聚类算法 DDCA

算法分为两个部分:确定聚类数目 $p$ 的算法 DNCP(decide number of clusters procedure)和动态概念聚类算法 DCP(dynamic clustering procedure).

确定聚类数目 $p$ 的算法 DNCP 可以描述为:任选一个无类属的元组 $S_i \in T$,用给定的相似度阈值 $\mathrm{Sim_{th}}$ 为半径做一个超球,统计落入球内元组 $S_j (i \neq j)$ 的个数 count,满足计数阈值 $N_{\mathrm{th}}$ 且与其他类中心的距离足够远时,聚类数目 $p$ 加 1.

**算法 2.2.** DNCP

输入:关系表 $T$,相似度的阈值 $\mathrm{Sim_{th}}$,类间最小距离 $L_{\mathrm{th}} (>=\mathrm{Sim_{th}})$,类内最少元组个数 $N_{\mathrm{th}}$,权重向量 $\beta_{kj}$;

输出:表达 $p$ 个聚类中心 Cluster_C 的表 $Y_i$.

方法:

```
p=1; ClusterCount[p]=1;
for (T 中每个元组 S_i) {                    // 测试 S_i 与所有其他 j(=n-i)个 S_j 的相似度.
    for (其余 n-i 个 S_j) {                 // 仅取未被划归到其他类中的元组计算相似度;
        D=Distance_o(S_i,S_j)+Distance_c(S_i,S_j);
        if (D<Sim_th) {
            将元组 S_i 划归到聚类 ω_p;
            Y_p={(max p(A_j=a_jk|p))|_{j=1}^{m}}; // 确定聚类中心属性取值
            ClusterCount[p]--;
        }
    } p++;
 }p--;
for (聚类中心表的 p 个 Y_i) {
    D_y=d(Y_i,Y_j);          // 计算聚类中心 Y_i 相互之间的语义距离;
    if (D_y<L_th){merge(ω_i,ω_j);p--;}
 }
for (所有 p 个聚类 ω_p) {          //查验每个聚类内的元组个数 ClusterCount[p];
    if (ClusterCount[p]<N_th){DeleteCluster(ω_p);p    ;}
 }
```

**定理 1.** 算法 DNCP 在给定相似度阈值 $\mathrm{Sim_{th}}$、类间最小距离 $L_{\mathrm{th}} (>=\mathrm{Sim_{th}})$ 和类内最少元组个数 $N_{\mathrm{th}}$ 后,能够按照聚类准则 $J$ 对表 $T$ 中的元组进行相似程度判定,至多在 $O(n)$ 时间内输出确定的 $p$ 个聚类中心 Cluster_C.

证明:设表 $T$ 由有限个元组 $S_i$ 组成 $(i=1,2,\ldots,n)$,元组 $S_i$ 的每一个属性值取自有限的概念符号值域,当给定适当的一组阈值 $\mathrm{Sim_{th}}$,$L_{\mathrm{th}}$ 和 $N_{\mathrm{th}}$ 时,由于算法是基于相似度来确定表 $T$ 中存在的不同分布的类的数目,所以算法 DNCP 的输出是确定的.

设表 $T$ 中的元组 $S_i$ 是可分类的.由算法描述可以明显地看出,时间复杂性由测试 $S_i$ 与所有其他 $j(=n-i)$ 个 $S_j$ 的相似度部分确定.首先是 $n-1$ 次计算比较,以确定 $S_i$ 与所有其他 $n-1$ 个 $S_j$ 的相似度;并将那些属于第 $p(=1)$ 类的 $v$ 个元组从 $T$ 中剔除;在第 $i$ 次计算比较时,只需将 $S_i$ 与所有其他 $j=n-v-i+1$ 个 $S_j$ 相比较.故算法 DNCP 的时间复杂度至多为 $O(nm)$.其中 $n$ 是数据库表 $T$ 的尺度 $n=|T|$,$m$ 是属性的个数.实际上,相对于数据挖掘中所处理的数据集合 $T$ 的尺度而言,$m$ 可以忽略不计. □

### 2.3 动态聚类算法 DCP

动态聚类算法 DCP 循环地将数据库表 $T$ 中的每一个元组 $S_i$ 分别与聚类中心表 $Y_i$ 中的 $p$ 个

聚类中心计算比较语义距离(相似度),将元组 $S_i$ 划归到概念上最接近的聚类 $\omega_p$ 中;每增加一个元组,就要修改调整获得新加入元组的类 $\omega_p$ 的聚类中心.如果元组 $S_i$ 取自其他的类 $\omega_q$,则在减少元组的类 $\omega_q$ 的同时也要调整其聚类中心 $y_l$ 的属性取值.这个过程直到某个计算比较循环中不再发生元组在类之间进行调整为止.

**算法 2.3.** DCP(dynamic clustering procedure)

输入:数据库表 $T$;$p$ 个聚类中心表 $Y_l$;

输出:标记有类别的 $p$ 个聚类表 $T[p]$.

方法:

```
for (T 中每个元组 S_i) {
    设 minDis 为第 i 个元组 S_i 与任意一个聚类中心 Y_1 的语义距离;
    for (λ=1,λ<=|Y|,λ++) {
        设 distn 为第 i 个元组 S_i 与第 λ 个聚类中心 Y_λ 的语义距离;
        if (distn<minDis) {                    // 分别判定元组 S_i 到各聚类中心的距离
          minDis=distn;
          T cluster=S_i; member=λ;             //求得的元组 S_i 的类属
        }
    }
    T cluster(cluster)=member;                 // S_i 与 Y_λ 有最大相似
    Y_λ=(max p(A_j=a_jk|λ))_{j=1}^{m};         // 调整第 λ 个类的聚类中心
}
/* 以下不断调整所有元组的类属并调整新的 Y_λ,直到没有可调整的元组(move=0)时为止 */
repeat
  move=0;                                      // 元组类属调整标志
  for (T 中每个元组 S_i) {
        设 minDis 为第 i 个元组 S_i 与任意一个聚类中心 Y_1 的语义距离;
        for (λ=1,λ<=|Y|,λ++) {
            设 distn 为第 i 个元组 S_i 与第 λ 个聚类中心 Y_λ 的语义距离;
            if (distn<minDis)
            minDis=distn; member=λ;
        }
  }
     if (元组 S_i(cluster)<>member) {          // 发现第 i 个元组 S_i 到聚类中心 Y_λ 距离更近
            move=1;
            q=S_i(cluster);          //取出原类属标记,
            S_i(cluster)=member;
            调整加入新元组的聚类中心 Y_member={(max p(A_j=a_jk|member))|_{j=1}^{m}};
            调整减少元组的聚类中心 Y_q={max p(A_j=a_jk|q))_{j=1}^{m}};
     }
}
until move=0.
```

**定理2.** 算法 DCP 的时间复杂性为 $O(cnp)$.其中 $n$ 是表 $T$ 的尺度 $n=|T|$;$p$ 是输出的聚类数;$c$ 是常数,取决于 repeat-until 的循环次数.

证明:首先,算法 DCP 是收敛的.因为在算法 DNCP 中指定了不同聚类间的最小距离 $L_{th}$ 大于给定的最小相似的语义距离 $Sim_{th}$,本算法重新划分任一元组 $S_i \in T$ 到最相似的聚类 $\omega_p$,聚类中心的调整只能是向 $S_i$ 的方向调整,而减少元组的聚类中心向偏离 $S_i$ 的方向调整.在下一轮比较最相似的循环过程中,$S_i$ 只能是与聚类 $\omega_p$ 更相似.算法 DCP 遍历数据库表 $T$ 中所有 $n$ 个元组 $S_i$,分别

计算 $J=\min\sum_{i=1}^{p}d(S_i,Y_i)$,即元组 $S_i$ 与 $p$ 个聚类中心比较相似程度,每循环一次(repeat-until)的时间代价是 $O(np)$. 设只要一遍循环就能确定所有元组的正确类属划分,算法至少要循环两遍,此时 $c=2$. 从以上算法收敛的证明可知,$c\ll n$. □

### 2.4 概念层次生成

概念层次生成算法 HCGP(hierarchical concept generate procedure)对算法 DCP 的结果进行概念归纳. 动态概念聚类的执行结果,使得所处理的数据对象得到极大的约减. 概念层次生成的算法可以考虑在内存中高效实现. 设有 $p$ 个聚类 $\omega_i$ 组成规则集合表 $T_i\subset T$,$T_i$ 中的元组表示原子规则,每个元组有 $m$ 个属性. 在算法 HCGP 中,表 $T_i$ 记为规则集 $R\_set0$.

**算法2.4.** HCGP

输入:DCP 产生的聚类输出 $R\_set0$,相似计数阈值 $\kappa_\lambda$;

输出:概念层次表 HCT.

方法:

```
candidt_set=R_set0;
while (candidt_set≠∅) {
    for (所有规则 R_i,R_j∈candidt_set) {        // i≠j,i,j=1,2,...,n
        sim_valu=simlr_juge(R_i,R_j);           // 返回规则 R_i,R_j 的相同属性的计数值,
        将 sim_valu 填写到相似矩阵表 sim_arrry;
    }
    for (sim_arrry 中的元素 C_ij) {             // C_ij=—sim_valu
            if (C_ij>=相似计数阈值 κ_λ) {
                    new_rule=new_rule∩merger (R_i,R_j);
                    delete(candidt_set,R_i,R_j);
            }
    candidt_set=candidt_set∪new_rule;    λ++; //λ:相似计数阈值 κ_λ 指针
```

表1给出了由算法 DCP 输出的部分规则表,表示了在指定的时间段内客户的交易行为规律. 当相似度阈值 $\kappa_1=5$时,得到归纳结果(1,2),(3,4),(5,6,8)和(7,8). 当相似度阈值 $\kappa_2=3$时,得到更高层次的归纳概念(1,2,3,4)和(5,6,7,8). 图3表示了算法 HCGP 产生的概念层次结构. 遍历该结构,可以得出某些证券投资客户交易行为的规律.

**Table 1** The rules set output $R$-set0 from DCP

**表1** 算法 DCP 产生的聚类输出 $R\_set0$

| # | Stk_number① | Trd_mode② | Trd_types③ | Strength④ | Frequency⑤ | Assets⑥ | Custm_type⑦ | Loss⑧ | Samples⑨ |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 3..5 | A | A∨Z | 1..5 | 1..3 | 少 | A | 5..15 | 11 |
| 2 | 3..5 | B | A∨C | 1..5 | 1..3 | 少 | A | 5..15 | 18 |
| 3 | 3..5 | A | A | 5..12 | 3..6 | 多 | A | 5..15 | 23 |
| 4 | 3..5 | B | A | 5..12 | 3..6 | 中 | A | 5..15 | 15 |
| 5 | 5..8 | A | A | 1..5 | 5..10 | 中 | A | 15..25 | 8 |
| 6 | 5..8 | C | A | 5..12 | 6..10 | 中 | A | 15..25 | 28 |
| 7 | 5..8 | C | G | 1..5 | 1..3 | 少 | B | 5..15 | 31 |
| 8 | 5..8 | A | G | 1..5 | 1..3 | 中 | B | 15..25 | 16 |

①股票数量,②交易类型,③交易品种,④交易力度,⑤交易频度,⑥资产,⑦客户类型,⑧亏损,⑨样本数.

(1) [客户类=散户][交易股票文数=3~5][频度不大于3次][手数不大于5手]→[资产=小][可信度=30%];

(2) [资产=小][客户类=散户][交易类型为 $A$ 或 $B$]⇒[亏损率在5～15%][可信度=90%];

(3) [参与国债交易的客户]⇒[客户类别=$B$]∧[交易频度=少][可信度=34%].

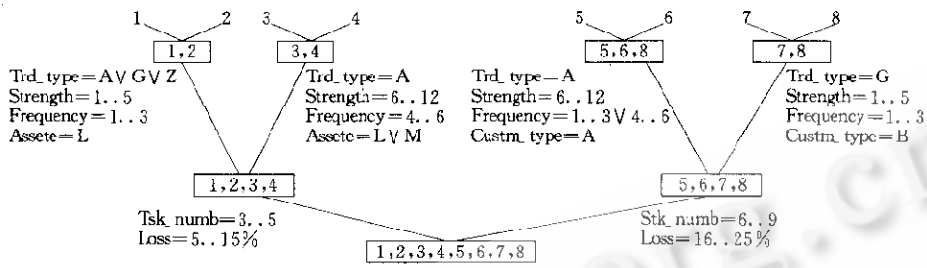



Fig. 3 The hierarchical concept from HCGP

图3 算法HCGP输出的层次概念

## 3 实验分析

为了验证算法 DDCA 的可行性,我们分别选用模拟数据和真实数据测试了某些阈值的选择对算法的执行时间和聚类输出的影响.

### 3.1 算法执行时间与数据库表规模|$T$|的关系

实验表明,表 $T$ 每增大10倍时,算法的执行时间就增加约10倍.用随机样本发生器生成模拟数据的规则是:在20维的向量空间中随机产生50个聚类点,每点有20个坐标,在各聚类点的每个坐标点处分别按正态分布产生[1,10000]上的整数,组成各聚类点处的2000个记录(每个点的各维取不同的分布参数);得到100K 个记录数据作为 Samples.然后以平均随机抽样的方法,分别抽取5K,10K,50K,100K,500K 和1M 记录作为数据子样.分别对数据子样各执行算法10遍,记录算法的执行时间.计算得到执行算法的时间均值,时间方差和数据规模的对应关系见表2.实验表明,算法的效率是可以满足挖掘要求的.

Table 2 The relation between running time and data size

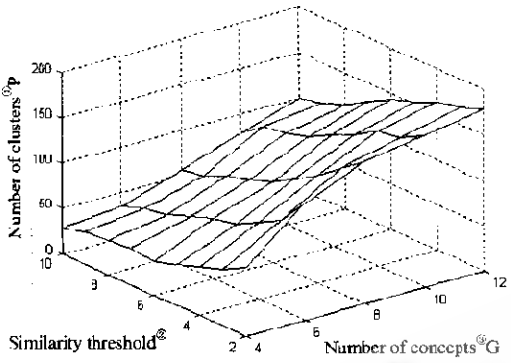
表2 数据规模与执行时间对照表

| Record size① | 5K | 10K | 50K | 100K | 500K | 1000K |
|---|---|---|---|---|---|---|
| Average running time②(s) | 15.2 | 35.32 | 135.1 | 337.3 | 1 426 | 3 652 |
| Covariance of time③ | 3.51 | 4.1 | 3.26 | 8.5 | 10.4 | 23.9 |

①记录数,②平均执行时间,③执行时间均差.

### 3.2 连续属性值概念分段对聚类数目的影响

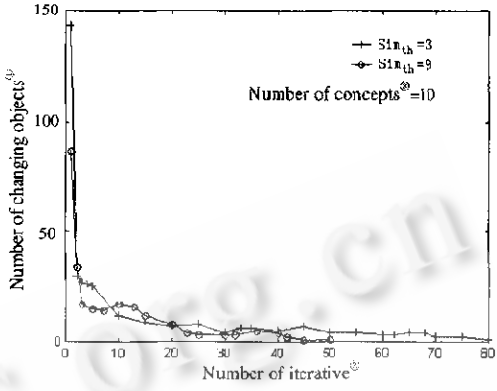
算法 DDCA 的特点是,先将数据库表中的那些连续属性取值作概念化处理,通过定义语义距离来实现聚类.属性值域的大小直接影响到聚类结果.图4表示了对于同一个数据样本(5K 个记录,17个属性),当取不同的概念分段数目 $G$ 和相似度阈值 $\mathrm{Sim_{th}}$ 时所得到的不同聚类个数 $p$ 的关系图.$x$ 轴表示概念分段数目 $G$,$y$ 轴表示相似度阈值 $\mathrm{Sim_{th}}$,$z$ 轴为聚类数目 $p$.关于连续属性值概念分段阈值 $G$ 对聚类误差的影响,可以理解为:当给定数据库表 $T$ 时,增大 $G$,则产生的概念属性值域的元素就会增加,表示用户关心的概念粒度较小,所得到的聚类数目有所增加;反之,表示用户关心的概念粒度较大,所得到的聚类数目有所减少.相似度阈值 $\mathrm{Sim_{th}}$ 减小,元组间的相似性要求严格,则分类数目增加;反之,分类数目减少.




①聚类数目,②相似度阈值,③概念分段数目.

Fig. 4 The relation with granularity, similarity threshold and number of clusters

图4 概念粒度和相似度阈值对聚类的影响




①调整元组个数,②循环次数,③概念分段数.

Fig. 5 The relation between similarity and convergence

图5 相似度阈值对动态调整次数的影响

### 3.3 动态聚类算法 DDCA 的收敛特征

动态聚类算法 DDCA 的时间复杂性与动态调整元组的循环次数 $c$ 有关,而 $c$ 取决于相似度阈值 $Sim_{th}$. 实验表明,算法 DDCA 的收敛速度是较快的,动态调整次数 $c \ll |T| = n$. 图5给出了当对真实数据(5K)进行聚类分析时,属性值分段数为10,分别选用不同的相似度阈值 $Sim_{th}$,对聚类过程中的动态调整元组的个数和循环次数 $c$ 的影响.

## 4 结 论

研究高效的、结合背景知识的聚类分析方法,是数据挖掘技术在数据库中发现知识的一个重要的研究方面. 概念聚类更适用于领域知识不完整或领域知识缺乏时的数据挖掘任务. 它不仅可以用于直接将大数据集合划分为有意义的模式类,而且还可以用于其他数据挖掘方法所产生结果的后处理. 因为作为数据挖掘的输出其本身也往往构成了一个巨大的规则空间. 本文结合"证券投资客户行为规律",研究并提出了一种基于背景语义的距离函数和快速概念聚类算法. 研究了利用领域知识确定的各个阈值对算法的影响,实验表明,算法是有效和可行的. 进一步的研究是将算法作通用性改进,寻求背景知识的通用表达模式和维护技术,以便使概念聚类挖掘算法可以用于其他的应用领域.

## References:

[1] Fayyad, M., Piatetsky-Shapiro, G., Smyth, P. From data mining to knowledge discovery: an overview. In: Fayyad, M., Piatetsky-Shapiro, G., Smyth, P., eds. Advances in Knowledge Discovery and Data Mining. Menlo Park, CA: AAAI Press, 1996. 1~36.

[2] Michalski, R., Stepp, R. Automated construction of classification: conceptual clustering versus numerical taxonomy. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1983,5(4):396~409.

[3] Ng, R., Han, J. Efficient and effective cluster method for spatial data mining. In: Bocca, J., Jarke, M., Zaniolo, C., eds. Proceedings of the 20th International Conference of Very Large Data Bases. San Francisco, CA: Morgan Kaufmann Publisher, 1994. 144~155.

[4] Yan, W., Larson, P. Eager aggregation and lazy aggregation. In: Dayal, U., Gray, P., Nishio, S., eds. Proceedings of the 21st International Conference of Very Large Data Bases. Los Altos, CA: Morgan Kaufmann Publisher, 1995. 345~

357.

[5] Easter, M., Kriegel, H. P., Sander, J., *et al.* A density-based algorithm for discovering clusters in large spatial databases with noise. In, Simoudis, E., Han, J., Fayyad, U., eds. Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining. Menlo Park, CA: AAAI Press, 1996. 226~231.

[6] Fisher, D. Knowledge acquisition via incremental conceptual clustering. Machine Learning, 1987,2(2):461~465.

[7] Huang, Z. Clustering large data sets with mixed numeric and categorical values. In: Lu, H., Motoda, H., Liu, H., eds. Proceedings of the 1st Pacific-Asia Conference on Knowledge Discovery and Data Mining. Singapore: World Scientific, 1997. 21~34.

[8] Li, C., Biswas, G. Unsupervised clustering with mixed numeric and nominal data: a new similarity based agglomerative system. In: Lu, H., Motoda, H., Liu, H., eds. Proceedings of the 1st Pacific-Asia Conference on Knowledge Discovery and Data Mining. Singapore: World Scientific, 1997. 35~48.

[9] Cheng, Ji-hua, Guo, Jian-sheng, Shi, Peng-fei. Multi Strategy approach to mining interesting rules. Chinese Journal of Computers, 2000,23(1):47~51 (in Chinese).

**附中文参考文献:**

[9] 程继华,郭建生,施鹏飞.挖掘所关注规则的多策略方法研究.计算机学报,2000,23(1):47~51.

## An Efficient Dynamic Conceptual Clustering Algorithm for Data Mining*

GUO Jian-sheng, ZHAO Yi, SHI Peng-fei

*(Institute of Image Processing and Pattern Recognition, Shanghai Jiaotong University, Shanghai 200030, China)*

E-mail: jsguo@jcinco.com.cn

**Abstract:** Conceptual clustering analysis is suitable to discover the knowledge in database with incomplete or absent domain background information. It is difficult for original conceptual clustering method to deal with the data objects described by numerical attribute values. A new criterion function based on semantic distance is proposed in this paper, and a novel domain-based dynamic conceptual clustering algorithm (DDCA) is also presented. With the discretization of the continuous attribute values, it works well on the datasets that are described by mixed numerical attributes and categorical attributes. The algorithm automatically determines the number of clusters, modifies the demoid according to the frequency of the attribute values within each cluster and gives out the interpretations of the clustering with the conceptual complex expression. The experiments demonstrate that the semantic based criterion function and the dynamic conceptual clustering algorithm are effective and efficient.

**Key words:** data mining; dynamic conceptual clustering; semantic distance; domain knowledge

* Received July 27, 1999; accepted February 1, 2000
Supported by the National Natural Science Foundation of China under Grant No. 69835010