蚁群算法在数据挖掘中的应用研究

张惟皎　刘春煌　尹晓峰
(铁道科学研究院电子所,北京 100081)
E-mail:wjzhang@rails.com.cn

摘　要　蚁群算法是一种新型的模拟进化算法,在求解复杂的组合优化问题中获得成功并表现出良好的性能。文章介绍了蚁群算法在分类和聚类两个重要的数据挖掘任务中的应用研究情况,阐述了算法的基本原理及特性,最后总结了蚁群算法在数据挖掘应用中尚待解决的问题。

关键词　蚁群算法　数据挖掘　聚类　分类

文章编号 1002-8331-(2004)28-0171-03　**文献标识码** A　**中图分类号** TP274

Application Research on Data Ming Using Ant Colony Algorithm

Zhang Weijiao　Liu Chunhuang　Yin Xiaofeng
(Institute of Computering Technology, China Academy of Railway Sciences, Beijing 100081)

Abstract: Ant colony algorithm is a novel simulated evolutionary algorithm which shows many promising characters and has been applied successfully to a range of different combinatorial problem.This paper respectively introduces the applications of ant colony algorithm to two major data mining tasks-classification and clustering, addresses the basic principle and characteristics of algorithms.Finally the problems to be solved mentioned.

Keywords: ant colony algorithm, data mining, clustering analyse, classification

1　前言

数据挖掘是在海量的数据中寻找模式或规则的过程。数据挖掘是多学科领域,其核心学科涉及机器学习、统计学和数据库。数据挖掘强调的是发现知识,获得的知识类型包括关联规则,分类、回归、聚类、依赖模型等。大型数据库为知识的产生和验证提供了丰富的相对可靠的数据资源,从数据库中发现的知识更有助于决策支持和过程控制。

蚁群算法(ant colony algorithm)是由意大利学者 Dori-go, Maniezzo 等人在 20 世纪 90 年代初首先提出来的[1]。蚁群算法在若干领域已取得了成功的应用,尤其是在组合优化问题的解决上获得了比较理想的效果,如旅行商问题(TSP)、QAP 问题、Job-shop 调度等。此外在一些实际问题的解决中也取得一定进展,如大规模集成电路综合布线以及网络数据包的路由。蚁群算法是一种新型的模拟进化算法,其在数据挖掘中的应用正逐步引起人们的关注。目前,人工蚁群在知识发现的过程中主要用于发掘聚类模型和分类模型,该文将分别讨论蚁群算法在发掘分类模型和数据聚类中的研究及应用。

2　基于蚁群算法的聚类分析

聚类分析是将一组对象分成若干个群体,每个群体构成一个簇,使得簇内的对象尽可能具有最大的相似性,不同簇之间的对象尽可能有最大的相异性。目前,聚类方法主要有 k 均值法,模糊聚类、神经网络聚类、基于遗传算法的聚类、小波变换聚类以及将这些算法有效结合而形成的改进方法。随着蚁群算法研究的兴起,人们发现在某些方面采用蚁群模型进行聚类更加接近实际的聚类问题。基于蚁群算法的聚类方法从原理上可分为两种:一种是基于蚁堆形成原理来实现数据聚类,文献[2]对这种方法进行了详细的描述;另一种是运用蚂蚁觅食的原理,利用信息素(pheromone)来实现聚类分析[3]。

2.1　基于蚁堆原理的聚类分析

蚁堆聚类现象的基本机制是死蚂蚁通过工蚁的搬运实现相互间吸引:小的蚁堆通过吸引工蚁积攒更多的死蚂蚁来逐渐变大,这种正反馈会导致蚁堆逐渐越积越大。根据这种机制对数据进行聚类,数据对象在空间的分布状态将影响聚类的结果,其基本思想如下:

假设所有的数据对象都随机地分布在二维的与数据库成比例伸缩的网格空间。处于网格中的两个对象 o_i 和 o_j 之间的距离(相异度)$d(o_i,o_j)$用它们之间的欧氏距离计算。如果 o_i 和 o_j 是同一类对象(即 o_i 和 o_j 相似),则 $d(o_i,o_j)=0$,反之 $d(o_i,o_j)=1$,这样可得到二进制的相异度矩阵。设若干个蚂蚁在二维网格上不断运动并反复执行拾起(pick)或放下(drop)对象的操作,如果蚂蚁在时刻于位置 r 发现对象 o_i,那么在 r 处与 o_i 相似的对象的局部密度为:

$$f(o_i)=\max\left\{0,\frac{1}{s}\sum_{o_j\in Neigh_{s\times s}(r)}\left[1-\frac{d(o_i,o_j)}{\alpha\left(1+\frac{(v-1)}{v_{\max}}\right)}\right]\right\}$$

$f(o_i)$是对象 o_i 领域中出现其它对象 o_j 时 o_i 的平均相似度度量。参数 α 为相异度比例,位置 r 的领域 $Neigh_{s\times s}$ 的面积为 $s\times s$。蚂蚁的速度 v 均匀分布于$[1,v_{\max}]$(v 是一个蚂蚁在一个时间单元内沿给定的网格轴线行走的网格单元数)。

基金项目:铁道部科技研究开发计划项目资助(编号:2001X008-B)

作者简介:张惟皎(1972-),女,博士研究生,主要研究方向为大型信息系统,数据挖掘,计算智能。刘春煌(1946—),男,研究员,博士生导师,主要研究方向大型信息系统,数据仓库,系统集成。

这样，在每次循环中，蚂蚁拾取或放下一个对象都遵循以下原则：如果一个蚂蚁没有搬运对象，那么它会随机地从邻近的单元拾取一个对象，拾起概率为 $p_p(o_i)=\left(\frac{k_1}{k_1+f(o_i)}\right)^2$；如果一个蚂蚁有搬运对象，那么它会随机地选择邻近的空单元放下该对象，或者，如果搬运对象与邻近的对象相似则放下搬运对象，放下概率为 $p_d(o_i)=\begin{cases}2f(o_i), f(o_i)<k_2\\ 1, f(o_i)\geq k_2\end{cases}$。$k_1$，$k_2$都是阈值常量。

上述方法实际上是一种基于网格和密度的聚类方法。高维数据空间首先要映射到某一低维网格空间以便处理，映射要确保簇内距离小于簇间距离，同时网格的精细度将影响聚类结果的精细度。蚂蚁拾起或放下对象受其局部相似密度 $f(o_i)$影响，局部相似密度大，拾起概率 $p_p(o_i)$小，数据对象不易从该簇移走，同时放下概率 $p_d(o_i)$大，对象倾向于留在该簇；反之亦然。同时，蚂蚁的运动速度也影响拾起或放下对象的倾向，不同速度的蚂蚁可以同时形成各种规模的聚类：快速的蚂蚁可以在较大的范围内形成粗糙的聚类，而慢速的蚂蚁可以在小范围内更精确地聚集对象。另外，蚂蚁还可以在聚类过程中保留“短期记忆”(short-term memory)，记录下前几次操作的对象，并将当前操作的对象放置到与其中最相似的对象的位置上，避免多次随机搜索。基于蚁堆的聚类算法不必预先指定簇的个数，并且能构造任意形状的簇。

2.2 基于蚂蚁觅食原理的聚类分析

蚂蚁的觅食过程可分为搜索食物和搬运食物两个环节。每个蚂蚁在运动过程中都会在其经过的路径释放信息素，蚂蚁能够感知信息素的存在及其强度。经过蚂蚁越多的路径其信息素越强，同时信息素自身也会随着时间的流逝而挥发。蚂蚁倾向于朝着信息素强度高的方向移动，某一路径上走过的蚂蚁越多，后者选择该路径的概率就越大，整个蚁群的行为表现出信息正反馈现象。基于蚂蚁信息素痕迹的聚类分析基本思想如下：

将数据视为具有不同属性的蚂蚁，聚类中心既是蚂蚁所要寻找的“食物源”，那么数据聚类过程就可以看作是蚂蚁寻找食物源的过程。假设数据对象为 $X=\{X|X_i=(x_{i1},x_{i2},\cdots,x_{im}),i=1,2,\cdots,N\}$，算法首先进行初始化，将各个路径的信息素 $\tau_{ij}(0)=0$ 置为 0，设置簇半径 r，统计误差 ε，代表对象 M 等参数。计算对象 X_i 到 X_j 之间的加权欧式距离 d_{ij}，计算各路经上的信息素 $\tau_{ij}(t)=\begin{cases}1, d_{ij}\leq r\\ 0, d_{ij}>r\end{cases}$，对象 X_i 合并到 X_j 的概率为：

$$p_{ij}=\frac{\tau_{ij}^{\alpha}(t)\eta_{ij}^{\beta}(t)}{\sum_{s\in S}\tau_{sj}^{\alpha}(t)\eta_{sj}^{\beta}(t)}$$

其中，$S=\{X_s|d_{sj}\leq r, s=1,2,\cdots,j,j+1,\cdots,N\}$。如果 $p_{ij}(t)$大于阈值 p_0，那么将 X_i 合并到 X_j 的领域。

上述聚类方法中代表对象 M 对运行效率和聚类结果影响较大，代表对象的选择方法也比较多，可以根据情况尝试不同的方法避免算法陷于局部最优。算法虽然不需要事先给定聚类的个数，但由于簇半径是预置的，所以聚类的规模受到限制。

这两种方法的不同之处主要在于蚂蚁个体间的通信介质不同。在第一种方法中，蚂蚁之间是通过对象的空间分布状态达到相互作用的目的；而在第二种方法中，蚂蚁的通信介质是其在所经路径上留下的信息素。另外，文献[4]在蚁堆聚类的基础上提出了基于人工蚂蚁的混合聚类新方法，交替使用蚁群算法聚类和均值算法修正误差，得到“高质量”的聚类。

3 基于蚁群算法的分类模型

数据分类是数据挖掘的另一个重要主题，它是在数据库对象集合中寻找属性，并根据分类模式将其划分为不同类别的过程。分类过程利用历史数据记录自动推导出对给定数据的分类树。分类器构造方法有统计学方法、机器学习法、神经网络、决策树等。从知识发现的观点来看，分类规则的表达方式形如 IF < 条件 > THEN < 类 >，规则前件(IF 部分)包含一组条件集合，一般由逻辑连接符连接；规则结论(THEN 部分)定义了样本的预测类，这些样本的预测属性满足规则前件所定义的所有条件。

将蚁群算法引入分类规则的发现，是利用蚁群觅食原理在数据库中进行搜索，对随机产生的一组规则进行选择优化，直到数据库能被该组规则覆盖，从而挖掘出隐含在数据库中的规则，建立最优的分类模型[5]。蚁群算法搜索的初始条件为发现规则的集合为空，且训练集包含所有的训练样本。蚂蚁搜索一次要完成规则生成、规则剪枝、信息素更新三个任务。一次搜索生成一条规则，并且将这条规则加入发现规则集合，同时将该条规则所覆盖的训练样本从训练集中删除。如果未覆盖训练样本的数目大于用户定义的阈值，即最大未覆盖样本数，就反复执行上述过程，最终算法将得到一组最优分类规则集合。算法描述如下：

```
训练集={所有训练样本};
WHILE(训练集中样本数> 最大未覆盖样本数)
t=1;
j=1;
将所有路径的信息素初始化为相同的值;
REPEAT
    ①规则生成:Ant_t从一个空规则集开始,递增地建立规则 R_t;
    ②规则剪枝:对规则 R_t 进行剪枝;
    ③信息素更新:更新所有路径的信息素
      增加 Ant_t 经过的路径的信息素,减少其它路径的信息素;
IF(R_t=R_{t-1})   /* 更新收敛测试 */
THEN j=j+1;
      ELSE j=1;
    NED IF
    t=t+1;
  UNTIL(t≥蚂蚁总数)OR(j≥测试蚂蚁收敛的规则数目)
  从 R_t 选择最佳规则 R_best;
  将规则 R_best 加入发现规则列表;
  训练集=训练集-{R_best 当前覆盖的样本集};
END WHILE
```

搜索的核心操作是第一步——规则生成，即当前蚂蚁反复地一次将一个条件加入当前的局部规则。假设规则条件 $term_{ij}$ 的形式为 $A_i=V_{ij}$，A_i 是第 i 个属性，V_{ij} 是 A_i 域中的第 j 个值，$term_{ij}$ 加入当前局部规则的选择概率为：

$$P_{ij}=\frac{\eta_{ij}\cdot\tau_{ij}(t)}{\sum_{i=1}^{a}x_i\cdot\sum_{j=1}^{b_i}(\eta_{ij}\cdot\tau_{ij}(t))}$$

其中，a 为属性总数；如果属性 A_i 没有被当前蚂蚁使用，则 x_i 置为 1，否则置为 0；b_i 是第 i 个属性域中值的个数。η_{ij} 对应于 $term_{ij}$ 问题相关的启发函数的值，其计算公式如下：

$$\eta_{ij}=\frac{\log_2 k-H(W|A_i=V_{ij})}{\sum_{i=1}^{a} x_i\cdot\sum_{j=1}^{b_i}(\log_2 k-H(W|A_i=V_{ij}))}$$

$H(W|A_i=V_{ij})$是与 $term_{ij}$ 相关的熵。η_{ij} 的值越高，$term_{ij}$ 启发函数就越适合分类，其被选择的可能性就越大。

$\tau_{ij}(t)$是在循环 t 中 $term_{ij}$ 的信息素，即当前蚂蚁所经路径在位置 i,j 的信息素。第一个蚂蚁开始搜索时所有路径的信息素都是相同的。每条路径最初的信息素是与所有属性值的数量成反比，其定义如下式：

$$\tau_{ij}(t=0)=\frac{1}{\sum_{i=1}^{a} b_i}$$

根据选择概率选择 $term_{ij}$ 将其加到当前局部规则有两个约束：(1)当前局部规则不包含属性 A_i；(2)加入任何条件都会使规则覆盖的样本数量小于用户定义的阈值，即每条规则应覆盖的最小样本数。

信息素更新包括两个方面：(1) 增加 Ant_i 经过的路径的信息素，即增加已用条件的信息素，$\tau_{ij}(t+1)=\tau_{ij}(t)+\tau_{ij}(t)\cdot Q,\forall i,j\in R$，$Q$ 是规则质量[6]；(2)未用条件信息素的减少，将每个信息素 τ_{ij} 的值规格化，即将每个 τ_{ij} 除以 τ_{ij} 之和，$\forall i,j$。

基于蚁群算法的分类模型在启发函数和信息素更新中都引入熵，与传统决策树算法中只在生成树的过程中将熵作为仅有的启发函数不同，这使得的规则建构过程更具健壮性。而且由于信息素更新提供的正反馈信息可以帮助纠正熵度量的缺陷造成的错误，因此不易陷入搜索空间的局部最优。另外，决策树算法中熵度量是局部的启发度量，一次只考虑一个属性，因此易受属性间相互作用的影响，而信息素是基于规则整体性能的，信息素更新能更好地处理属性间的相互作用。从演化计算的角度来看，蚁群分类模型和遗传算法分类模型的基本原理是一样的，但是典型的 GA 规则发现采用随机的方式创建种群，没有任何启发方式，相比较蚁群分类器的收敛速度在某些情况下比 GA 分类器更快。

4 总结

虽然蚁群算法研究历史较短，但是已经显示出大量有趣的特征，包括灵活性、健壮性、分布性和自组织性等。这些特征使得蚂蚁算法非常适合本质上是分布、动态和需要内部容错的问题求解。数据挖掘实质上是知识的求解过程，即通过各种技术和方法获得知识，优化知识结构。将蚁群算法运用于数据发掘还存在一些问题，需要进一步研究：

(1)如何将现实的挖掘任务转换成蚁群求解的问题空间，并用适当的方式表达。如何定义“人工蚂蚁”以及蚂蚁间的非直接通信方式(如路径上的信息素、对象的分布状态等)的选择。

(2)如何建立正反馈机制，定义启发函数，递增地进行问题求解，并且使得到的解与问题定义中现实世界的情况相对应。

(3)基于蚁群的算法要初始化大量的参数，这些参数的选择会对算法的性能产生较大的影响，但其选取的方法和原则目前尚无理论上的依据，只能通过多次实验调优，因此参数的最佳设置原则还有待进一步研究。

(4)蚁群算法的搜索时间较长，如何将蚁群算法与遗传算法、免疫算法等优化算法相结合，改善和提高算法性能，以适应海量数据库的知识发现。

因而如何在数据挖掘中运用蚁群算法快速、高效地获得高质量的知识越来越受到人们的关注，逐渐成为近期的研究热点。(收稿日期：2004 年 6 月)

参考文献

1.Dorigo M,Maniezzo V,Colorni A.Ant System:Optimization by a Colony of Cooperating Agents[J].IEEE Trans On System,Man,and Cybernetics,1996;26(1):29~41

2.E Lumber,B Faieta.Diversity and adaption in populations of clustering ants[C].In:J-A Meyer,S W Wilson Eds.Proceeding of the Third International Conferrence on Simulation of Adaptive Behavior:From Animals to animates,MIT Press/Bradford Books,Cambridge,MA,1994:501~508

3.杨新斌，孙京诰，黄道.一种进化聚类学习新方法[J].计算机工程与应用，2003;39(15):60~62

4.N Monmarché.On data clustering with artificial ants[C].In:Data Mining with Evolutionary Algorithms,Research Directions-papers from the AAAI Workshop ed.Menlo Park,CA:AAAI press,1999:23~26

5.Rafael S Parpinelli,Heitor S Lopes,Alex A Freitas.Data mining with a ant colony optimization algorithm[J].IEEE Trans On Evolution Computing,2002;6(4):321~332

6.H S Lopes,M S Coutinho,W C Lima.E Sanchez,T Shibata,L Zadeh Eds.A evolutionary approach to simulate cognitive feedback learning in medical domain:Genetic Algorithm and Fuzzy Logic System:Soft Computing Perspectives[M].Singapore:World Scientific,1998:193~207

(上接 170 页)

2.K Ramamritham.Real-time databases[J].International Journal of Distributed and Parallel Databases,1993;1:199~226

3.J R Haritsa,K Ramamritham.Real-time data in the new millennium.Real-time System,2000

4.J Huang,J A Stankovic.An integrated real-time data management architecture for industrial control system.TR93-08,1993

5.张志檩主编.实时数据库原理及应用[M].中国石化出版公司出版，2001

6.杨庆，王堃，王宏安等.企业级工控实时数据库研究与实现[J].计算机工程与应用，2001;37(13):68~70

7.杨庆.流程工业 CIMS 中的实时数据库系统研究[D].硕士论文.中科院软件研究所，2001

8.张峰.面向对象时态数据库系统研究与开发[D].硕士论文.中科院软件研究所，2001

9.Bristol E H.Swinging door trending:Adaptive Trend Recording[C].In:ISA National Conference Proceedings,1990:749~753