

基于最近邻搜索算法分组式 P2P 网络拓扑模型

贾晓倩, 刘方爱

(山东师范大学 信息科学与工程学院, 山东 济南 250014)

摘要:为了在 P2P 环境中实现资源的更快更精确搜索, 引入兴趣相似度计算方法, 提出一种基于最近邻搜索算法的分组式 P2P 网络拓扑模型。在这个模型中, 采用余弦相似性方法计算共享资源的相似程度; 相似程度较高的节点形成朋友节点进行逻辑连接, 兴趣相近的节点聚集成一个小组, 结合缓存机制实现共享资源的高效搜索。模拟实验查询结果表明, 兴趣相似度 Sim 值越大资源搜索越精确。模型中相似度的引入增强了 P2P 网络中资源定位的准确率, 提高了搜索效率。

关键词:分组式 P2P 网络; 兴趣域; 相似度; 缓存机制; 搜索策略

中图分类号: TP391

文献标识码: A

文章编号: 1673-629X(2010)11-0100-05

A Topology Model Based on Nearest Neighbor for P2P Group Networks

JIA Xiao-qian, LIU Fang-ai

(School of Information Science & Engineering, Shandong Normal University, Jinan 250014, China)

Abstract: In order to search the resource of P2P environment faster and more accurate, introduces the similarity calculation method, and proposes a topology model based on nearest neighbor for P2P group networks. This model use the similarity cosine to calculate the similarity degree of sharing resources; High degree of similarity of the nodes make a logical connection, friend node in the node gathered near a group, combining the resources sharing caching mechanism high search. Simulation results show that the bigger the similarity Sim values, the more accurate the search of resources. The introduction of similarity enhances accuracy rate of P2P networks positioning, improves the searching efficiency.

Key words: P2P group networks; interest-domain; similarity; caching mechanism; search strategy

0 引言

P2P 系统^[1]是一种新的计算模式, 它认为系统中的各个节点是逻辑对等的, 弱化甚至取消了服务器的角色。每一个对等体可以充分利用其他对等体的各种资源。数据信息不需要集中管理, 而是分布在各个节点上, 使得整个系统的数据容量理论上没有任何限制, 较好地改善了性能瓶颈的问题。

如何高效地索引、查找、定位以及访问 P2P 系统上的共享资源是 P2P 网络实现的最关键问题。目前, P2P 搜索技术中最重要的研究成果是基于 DHT 的结构化搜索算法^[2]和基于 Small World 理论的非结构化搜索算法^[3]。基于 DHT 的 P2P 系统采用相容散列函数, 根据精确关键词进行对象的定位与发现。这种系

统的网络拓扑以及对等体的位置都有严格的控制, 搜索机制是利用分布式哈希表将请求发送到目标对等体, 一个明显的缺陷就是只支持单关键字查询和精确匹配, 查全率比较低。

非结构化 P2P 系统的网络拓扑没有严格的控制, 对等体可以随意加入或离开, 不存在中央服务器存储资源索引项, 数据存储的位置不容易精确地定位。这种松散式的结构容易支持关键字的查询, 但搜索效率以及查询范围有限。现有研究表明, 按共享兴趣组建的共享网络反映了人类社会中的一些“小世界”现象——良好的聚合性质和短路径性质, 即每个人只需要很少的中间人就和全世界的人建立起联系。这些性质可被利用来提高共享的效率。

文中利用系统内节点的兴趣相似度计算方法提出一种最近邻搜索算法, 提高查找用户最近邻居的准确度, 更好地定位网络中的共享资源。以“小世界模型”为理论依据, 结合最近邻搜索算法, 构建一种基于最近邻搜索算法的分组式 P2P 网络拓扑模型 NNGP2P (Nearest Neighbor Grouping P2P networks)。

收稿日期: 2010-03-26; 修回日期: 2010-06-18

基金项目: 山东省自然科学基金(Y2007G11)

作者简介: 贾晓倩(1985-), 女, 山东聊城人, 硕士研究生, 研究方向为网格计算、互联网络; 刘方爱, 博士, 教授, 博士生导师, 研究方向为并行处理、互联网络、网格计算等。

1 兴趣相似度的计算方法

最近邻搜索的核心是计算两个用户的兴趣相似度。

定义:兴趣域^[4]是由向量表示,表征节点的兴趣和对各个兴趣类别的感兴趣程度。一个兴趣域 D 表示: $D = (d_1, d_2, d_3, \dots, d_n)$ 其中, $(1) d_k (k \in (1, n))$ 是非负实数,代表一个兴趣类别(如:实况转播,文本文件,Video,SOAP等), d_k 的值衡量节点对此兴趣类别的感兴趣程度, d_k 的取值可以有多种算法。(2) 兴趣域向量的长度 n 取决于兴趣类别的数量。

在兴趣相似度的计算方法中,兴趣域用来记录节点对资源的兴趣爱好。

网络中的资源不仅有文档、文件,还有音频、视频和 RPC 接口。由于用户对各种资源的感兴趣程度不等,可以根据兴趣的不同定义一个分类树,也可称为兴趣树^[5,6],该树有以下几个特点:

- (1) 树中层次越高的节点代表的资源类型越广泛;
- (2) 每一层的节点代表的资源类型不能相互交叉;
- (3) 为便于实现,限定树的每个叶子节点到根节点的距离相同,即所有叶子节点都处于同一层,若按实际情况创建的兴趣树的某些叶子节点不在同一层,则为该节点添加与其类型一样的叶子节点,直到达到最低层为止。

在该分类树中添加共享资源时,广度优先遍历这棵兴趣树,如果资源属于树中某个节点所描述的类型,且该节点是叶子节点,则把其兴趣域的值增加 1,这样资源的添加和兴趣的更新就完成了。如图 1 所示是一个兴趣树结构的例子。

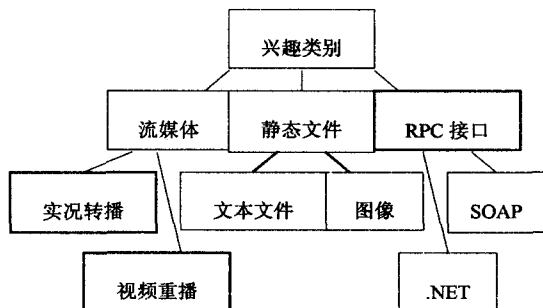


图 1 兴趣树结构的一个例子

对网络中每个共享资源按照兴趣树形式进行分类,节点共享的每一类资源的数目作为兴趣类别的值。例如:节点 A 所在 P2P 网络中共规划了 4 个兴趣类别:实况转播,文本文件,图像,SOAP;节点 A 共享了 13 个和实况转播有关的资源,6 个和图像有关的资源,则节点 A 的兴趣域为(13,0,6,0)。

向量空间模型^[7]将节点兴趣域映射为一个特征向量: $V(d) = (t_1, \omega_1(d); \dots; t_n, \omega_n(d))$, 其中 $t_i (i$

$= 1, 2, \dots, n)$ 是一列互不相同的词条项,每一个词条项可属于不同的兴趣类别; $\omega_i(d)$ 为 t_i 在 d 中的权值。在信息检索中词条的权值计算方法常用 TF-IDF 函数,下面是一种基于词条信息熵^[8]的 TF-IDF 公式:

$$\omega_i(d) = \frac{tf_{ik}(d) \log(N/n_k + 0.1)}{\sqrt{\sum_{k=1}^n (tf_{ik}(d))^2} \sqrt{\sum_{k=1}^n (\log(N/n_k + 0.1))^2}} * \log\left(\frac{1}{I_k(p) + \phi_k(1)} + 1\right)$$

其中 $tf_{ik}(d)$ 为词条 k 在文档 d 中出现的频率, N 为训练文档的总数, n_k 为训练文档集中出现词条 k 的文档数, $I_k(p)$ 是指词条 k 的信息熵, $\phi_k(1)$ 指根据词条文档总数计算出的该词条的次小信息熵的值 - $(\frac{n_k - 1}{n_k} \log(\frac{n_k - 1}{n_k}) + \frac{1}{n_k} \log(\frac{1}{n_k}))$ 。

对于两个节点 P_i 和 P_j , 它们之间的共享资源兴趣相似度可以采用余弦相似性方法计算。设两节点在 n 维向量类上的兴趣域为向量 i 和 j , 则其相似度计算为

$$\text{sim}(i, j) = \cos(i, j) = \frac{\sum_{k=1}^n \omega_k(d_i) * \omega_k(d_j)}{\sqrt{\sum_{k=1}^n (\omega_k(d_i))^2} \sqrt{\sum_{k=1}^n (\omega_k(d_j))^2}}$$

其中 $\omega_k(d_i)$ 和 $\omega_k(d_j)$ 分别为词条 k 在向量 i 和 j 中的权值。

寻找最近邻居的目标就是对每一个用户节点 q , 在整个用户空间中查找用户集合 $P = (p_1, p_2, p_3, \dots, p_n)$, 使得 q 不属于 P , 并且 p_1 与 q 的相似度 $\text{sim}(q, p_1)$ 最高, p_2 与 q 的相似度 $\text{sim}(q, p_2)$ 次之, 以此类推。

2 根据兴趣相似分组 P2P 网络

P2P 网络是一种构建在 Internet 之上的逻辑网络。文中以“小世界模型”为理论依据,结合兴趣相似度计算方法,构建一种基于最近邻搜索算法的分组式 P2P 网络拓扑模型 NNGP2P(Nearest Neighbor Grouping P2P networks)。

2.1 P2P 网络的初始化拓扑结构

(1) 首先,将 P2P 网络基于兴趣进行分组(如图 2 所示),划分成 N 个兴趣组 (Interest Similarity Group, ISG), 组的个数作为参数可以根据网络的规模进行设置,将整个网络空间区域按照一定比例映射到这 N 个组上,这样每个组就对应一个相应的物理空间范围。

(2) 网络分组后,为每个小组分配一个超级节点 S_i , 且假定这些节点是一定存在的(非超级节点可以自由加入或者离开网络)。

(3) 每个 S_i 节点维护与分组相关的信息包括:(a)

物理坐标范围, 定义为 $\text{Area}_i (1 \leq i \leq N)$; (b) 空闲节点 IP 地址列表, 定义为 $\text{freelist}_i (1 \leq i \leq N)$; (c) 存储了数据的节点 IP 地址列表, 定义为 $\text{loadlist}_i (1 \leq i \leq N)$; (d) 每个 S_i 节点设置一个 cache 缓存^[9], 用来保存该组各节点的历史查询记录。

(4) 为网络设置一个管理节点 T , 用来维护所有的 S_i 节点的 IP 地址以及 Area_i 信息。

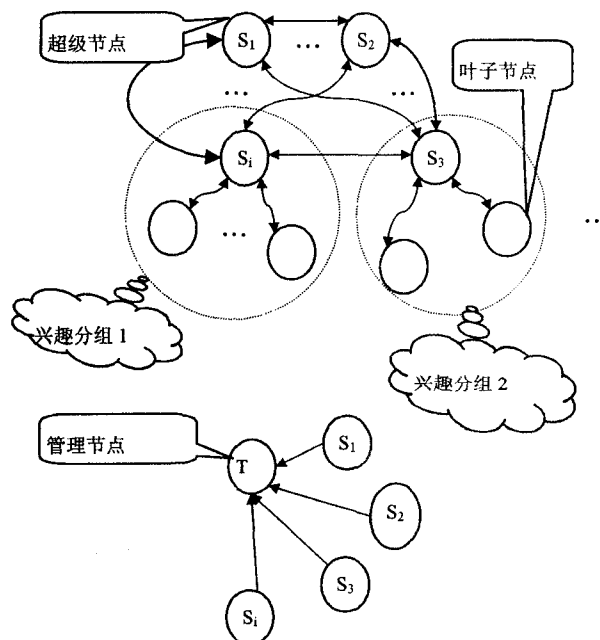


图2 NNGP2P网络拓扑模型

由于很多网络用户的兴趣比较广泛, 在 NNGP2P 网络中要求每个节点至少属于一个分组, 也可同时属于多个兴趣组; 模型中限制一个节点最多同时隶属兴趣组的数量(测试中要求一个节点最多同时属于 3 个兴趣组)以避免一个节点同时属于所有的兴趣组。每个分组选出一个综合性能(带宽、磁盘容量、CPU 处理速度等)最高的节点充当超级节点(SuperNode), 由它负责维护与兴趣分组相关的信息, 以及连接其它兴趣组的 SuperNode。网络中所有的 SuperNode 组成了一个小型的对等网, 相互之间可以进行直接访问, 以获取彼此的路由信息。在网络的最上层设置一个管理节点 T , 维护所有超级节点的 IP 地址以及其物理空间范围, 以适当调整兴趣组之间的负载均衡。

2.2 节点的加入

当一个新节点要加入网络时, 首先向最邻近的 S_i 提出加入请求。 S_i 一旦接受了新节点的加入请求, 就根据请求节点的路由信息为其在路由表内创建路由表项, 并与其建立一个逻辑链接。若新节点没有共享文件, 则将其 IP 地址放入 freelist_i ; 如果有共享资源, 则将其 IP 地址放入 loadlist_i 列表中, 同时修改网络中相应超级节点 S_i 和管理节点 T 的列表信息。属于同一个组

的彼此邻近的节点称为邻居节点(如图 3 所示)。

在保证消息沿着尽量短的距离转发的基础上, 综合考虑消息转发给最有可能存储查询资源的节点, 则查询消息要转发给相似度最高的节点。网络根据节点兴趣相似度计算方法, 将兴趣接近的节点逻辑上进行链接, 建立朋友节点。

2.3 捷径路线的建立

捷径路线即位于不同对等组中且具有较高兴趣相似度的节点间的相互访问。捷径路线上的节点称作捷径节点。捷径节点上保存一个快捷查询兴趣表 IST (Interest Shortcut Table), 该表保存与本节点具有较高兴趣相似度的节点信息。IST 表由 (Interest, flag, Num, address) 字段组成, 其中 Interest 代表节点查询的兴趣; flag 是快捷链接的标志, 取值为 0 或 1; Num 代表查询成功的次数, 开始赋初值 0; address 代表快捷链接的节点位置。

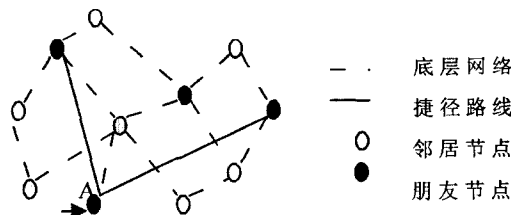


图3 NNGP2P网络中兴趣组节点的划分

2.4 节点的退出

P2P 网络中节点的加入和退出是动态进行的, 维护网络节点的正常退出也是 P2P 网络管理的一部分。节点的退出包括叶子节点的退出和超级节点的退出。

对于叶子节点 A 要离开它所属的兴趣组 ISG_i , 则 A 需要向它所在的分组所有节点发送广播消息, 所在兴趣组的 S_i 收到广播后, 首先删除其自身路由表中与 A 相关的路由信息, 以及 S_i Cache 中保存的关于 A 的历史查询信息; 然后再发送一个删除消息给其它兴趣组的超级节点, 通知它们检查各自的 S_i Cache, 查找与 A 相关的查询信息, 将其删除。若 A 节点资源为空, 则直接在 freelist_i 中删除其 IP 地址; 若为有资源节点, 则先将 A 中资源传送到其邻居节点, 然后在 loadlist_i 中删除其 IP。

对于超级节点 S_i , 为了防止单点失效情况的发生, 在 NNGP2P 网络中引入备用超级节点策略。在每个兴趣组中根据节点性能的高低进行排序, 设置多个备用超级节点 SSN (Spare Super Node), 每个组的超级节点要定期将该备用超级节点的路由表发送到管理节点 T 上进行备份。当 S_i 退出网络之前, 首先向管理节点发出退出请求, 管理节点接收到请求后从备份中选择兴趣组中的次超级节点 S_{i-1} 作为该组的超级节点。

S_i 将其维护的叶子节点路由信息表以及其他超级节点的相关信息发送给 S_{i-1} , S_{i-1} 收到路由信息后向 S_i 发送确认消息,待 S_i 收到确认消息后,管理节点删除 S_i 的 IP 和其物理区域, S_i 正式退出网络。

3 NNGP2P 网络搜索策略

每一个 S_i 设置一个 cache 保存该组查询资源频率较高的历史查询记录;同时每一个对等节点也设置一个 cache,cache 中保存该节点的历史查询记录、该节点的兴趣域表、邻居节点相关信息表 L 以及朋友节点的相关信息表 M ,搜索策略主要流程如图 4 所示。

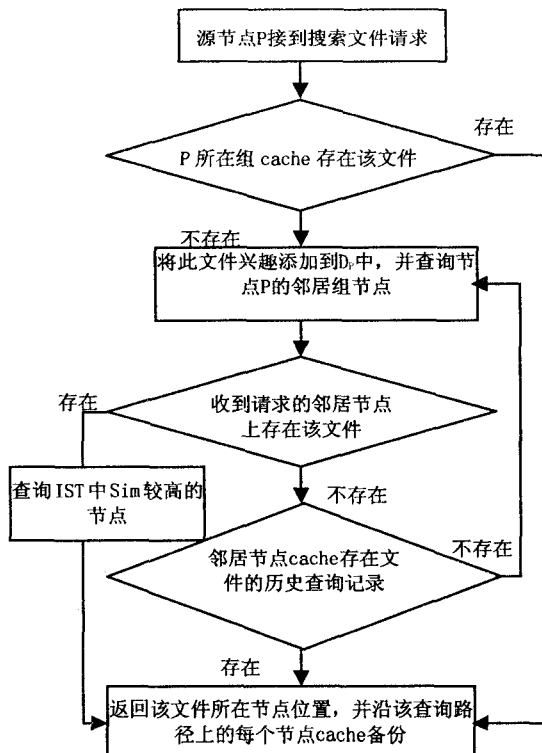


图 4 基于最近邻搜索策略的主要流程

其搜索的步骤主要包括:

(1)源节点 P 接到共享文件搜索请求后,先查找节点 P 所在的组 cache 历史查询记录,如果有相应的历史查询记录则转(4),否则转(2);

(2)将搜索请求的资源兴趣添加到节点 P 的兴趣域中,并查询对等节点 P 的邻居节点列表;如果收到信息的相邻对等节点上查找到该文件则查找 IST 中兴趣相似度较高的朋友节点,返回最接近搜索资源的文件(转(4)),否则转(3);

(3)查询该邻居节点中 cache 的历史查询记录,如果有相应的历史查询记录则转(4)否则转(2);

(4)返回该文件所在节点位置,并沿该查询路径上的每个节点进行 cache 备份。

其中,对等网中相邻对等节点一般是以 IP 包生存时间(Time to Live, TTL)或者节点查询的路由跳数为衡量标准的;类似生成树算法,节点向除原路径外其他相邻节点转发搜索信息;查询网络上所有节点后,搜索结束。

算法描述如下:

算法 1:兴趣组内的资源搜索算法

```

Status InsideGroupSearch ( InterestSimilarityGroup ISG, Document Doc, Node P)
{
  Route = Request (ISG, P, Doc); //发送路由请求
  IF (Route)
  {
    for (i=0; i<length(Route); i++)
    {
      IF (Search (Route[i], Doc))
      {
        Get (Route[i], Doc); Find = 1;
      }
      Else (! Find)
      {
        Compute (Sim (I, j)); //计算兴趣组内两节点共享资源的相似度
        Get (Route[j], Doc); Find = 1;
      }
    }
  }
  Else (! Find)
  {
    Flooding (P, Doc); //洪泛式搜索
  }
}
  
```

算法 2:兴趣组外的资源搜索算法

```

Status OutsideGroupSearch ( InterestSimilarityGroup ISG, Document Doc, Node P)
{
  R = Send (ISG, SuperNode, P); //向相邻兴趣组超级节点发送查询消息
  Route = Request (R, P, Doc); //发送路由请求
  IF (Route)
  {
    for (i=0; i<length(Route); i++)
    {
      IF (Search (Route[i], Doc))
      {
        Get (Route[i], Doc); Find = 1;
      }
    }
  }
  IF Else (! Find)
  {
    Compute (Sim (I, j)); //计算不同兴趣组间两节点共享资源的相似度
    Get (Route[j], Doc);
  }
  Else (! Find)
  {
    Flooding (P, Doc); //洪泛式搜索
  }
}
  
```

4 仿真测试与性能分析

为了验证 NNGP2P 的搜索效率,进行系统模拟和实验数据分析,将实验结果与非结构化 Gnutella 泛洪搜索机制进行对比。从以下两方面进行分析:(1)平均查询跳数的比较;(2)查询结果中兴趣相似度对资源搜索精确度的影响。

4.1 仿真环境设置

仿真中,使用 Peersim^[11,12]作为网络模拟实验平台,客户节点总数设定为 1000 个。每个节点初始包含资源 100 个,最大资源数为 300 个;每个节点的最大邻居数为 10,通过计算查询词条与路由上节点共享资源的兴趣相似度,进行资源定位,形成逻辑上的兴趣覆盖网。TTL 的最大值设为 10;查询精确度定义为要查找资源与满足条件资源的兴趣相似度的值,该值越大查询越精确。实验时,随机选取节点发起查询请求,总共 2000 次,过程中每隔 20 次就记录一下查询精确度。

图 5 显示了 NNGP2P 网络与 Gnutella 在平均查询跳数上的比较。实验表明通过引入兴趣组缓存机制, NNGP2P 网络的搜索提高了查询效率,减少了查询跳数。

假设信息查询实例为 I , I 对应的相关资源集合为 R , 则用 $|R|$ 表示集合中的资源总数。若用给定的搜索策略对实例 I 进行处理,得到一个结果集合为 A , 则用 $|A|$ 表示集合的文档数目。那么查准率是指搜索到的相关资源数与检索出资源总数的比值,即查准率 = $(|R| \cap |A|) / |A|$ 。图 6 是兴趣相似度对资源搜索精确度的影响,从图中可以看出 Sim 值越大,查询的精确度越高。

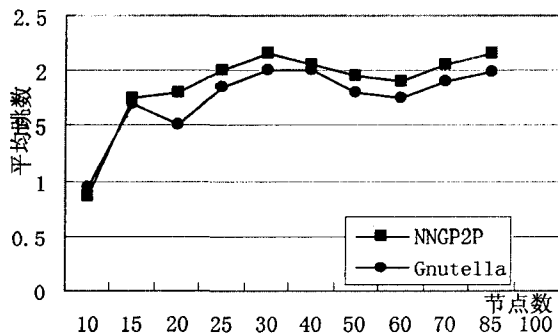


图 5 两种结构的平均跳数比较

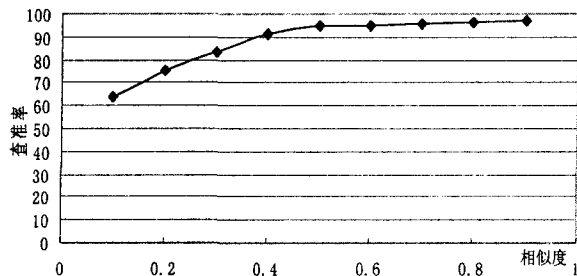


图 6 相似度对查准率的影响

4.2 性能分析

将 P2P 系统根据节点兴趣度的不同进行分组,每个分组进行各自的信息管理。利用兴趣的相似程度不同对随机网络进行分组,在一定程度上提高了共享资源查询的命中率,并且通过计算合适的权值,可以确保

兴趣相似度的计算更全面、更准确。但是在相似度计算过程中存在大量的权值计算和余弦夹角的计算,可能对性能存在一定的影响。

引入兴趣组缓存机制,极大地减少了搜索消息的数量,基本上一跳就能查询到所需资源,但是存储历史查询记录的 cache 表会随着查询不断增加而变大,相应 cache 表的维护开销也增大,为了控制 cache 表的大小,cache 表中一般只会保存使用频繁的一些查询。当节点资源负载过重时,还需要对各个分组的节点进行负载调整^[10],以降低整个系统的负载。

5 结束语

通过引入兴趣相似度的计算方法,结合缓存机制设计了一种基于最近邻搜索算法的分组式 P2P 网络模型,其中遵循了 P2P 系统在网络拓扑结构上的“小世界”特性,利用网络中节点各自的 cache 和相应的兴趣相似度的计算提高了系统的搜索效率,减小了系统的开销。下一步的工作是改进在节点加入和退出兴趣组的情况下对整个搜索效率的影响,并给出理论上的分析;对权值的选取和计算可以引入语义因素,通过神经网络学习技术进行修正。

参考文献:

- [1] Steinmetz R, Wehrle K. P2P 系统及其应用[M]. 北京:机械工业出版社,2008.
- [2] 李运娣,冯 勇. 基于 DHT 的 P2P 搜索定位技术研究[J]. 计算机应用研究,2006,23(10):226-228.
- [3] Manfredi S, di Bernardo M, Garofalo F. Small - World effects in networks: An engineering interpretation. ISCAS[EB/OL]. 2004-04-04. http://icceexplore.ieee.org/xpls/abs_all.jsp?arnumber=1329130.
- [4] 杨 舰,吕智慧,钟亦平,等. 一种基于兴趣域的高效对等网络搜索方案[J]. 计算机研究与发展,2005(5):804-809.
- [5] 周晓波,周 健,卢汉成,等. 一种基于层次化兴趣的非结构化 P2P 拓扑形成模型[J]. 软件学报,2007(12):3131-3138.
- [6] 陈素萍,谢丽聪. 一种文本特征选择方法的研究[J]. 计算机技术与发展,2009,19(2):112-115.
- [7] 黄萱菁,夏迎炬,吴立德. 基于向量空间模型的文本过滤系统[J]. 软件学报,2003(3):435-442.
- [8] 周炎涛,唐剑波,王家琴. 基于信息熵的改进 TFIDF 特征选择算法[J]. 计算机工程与应用,2007,43(35):156-158.
- [9] 李春洪,冯国富,顾铁成,等. 一种无“热点”的覆盖网协同缓存策略[J]. 软件学报,2008(3):744-754.
- [10] 熊 伟,谢冬青,焦炳旺,等. 一种结构化 P2P 协议中的自适应负载均衡方法[J]. 软件学报,2009(3):660-670.

(下转第 108 页)

有意义的规则。可以这样理解,若顾客甲在购买 A 的同时购买了 B,而顾客乙在购买 B 的同时购买了 C,则顾客甲有可能需要购买 C;

② 当 $A \Rightarrow B$ 是规则时,从宏观分析的角度,可以认为 $B \Rightarrow A$ 也是规则,因为它们都表示 A 与 B 之间的关联关系,宏观分析可以忽略微观细节;

③ 当 $A \Rightarrow B$ 、 $A \Rightarrow C$ 、 $A \Rightarrow D$ 是规则时, $A \Rightarrow BCD$ 可能也是有意义的规则,即:如果停止 A 的销售,可能同时对 B、C、D 的销售带来影响;或购买 A 可能同时需要购买 B、C、D;或购买 A 后,可选择性的需要购买 B、C 或 D;

④ 当 $A \Rightarrow B$ 、 $C \Rightarrow B$ 、 $D \Rightarrow B$ 是规则时, $ACD \Rightarrow B$ 可能也是有意义的规则,即:哪些商品的销售可能促进 B 的销售。

因此,从宏观分析的角度,可以把关联规则视为一个等价关系,借助等价关系的概念,可以把单项关联规则集按等价类划分,每一个等价类,就是具有某种关联关系的项集。基于这种思想的关联规则挖掘算法如下:

输入:交易数据库 D,支持度阈值 \min_sup ,置信度阈值 \min_conf ,隐含度阈值 $\min_implicit$

输出:单项关联规则集 R 及其等价类

方法:

```

R =  $\phi$ ;
 $L_1 = \text{find\_frequent\_1-itemsets}(D)$ ;
 $C_2 = \text{Apriori\_genC}_2(L_1)$ ;
for each transaction  $t \in D$  {
     $C_t = \text{subset}(C_2, t)$ ;
    for each  $c \in C_t$  c.count ++;
}
 $L_2 = \{c \in C_2 \mid c.\text{count} \geq \min\_sup\}$ ;
for each frequent 2-itemsets  $c \in L_2$  {
     $R_L = \phi$ ;
    for each  $x \subset c$ 
        if ( $\text{conf}(x \Rightarrow c - x) > \min\_conf$ )
            then  $R_L = R_L \cup \{x \Rightarrow c - x\}$ ;
     $R = R \cup R_L$ ;
}

```

return R 及其等价类;

5 结束语

提高数据挖掘算法效率的一条有效途径就是,针对具体的应用,尽量将规则的判断条件与挖掘过程结合起来,可以显著减少无效规则的产生,从而提高挖掘的速度和效率。文中讨论的隐含度削减、置信度削减方法也可以与其它 Apriori 改进算法结合起来,比如

AprioriTid、Partition、DHP 等算法。不过对交易数据库的削减不可取,破坏交易数据库将会带来灾难性的后果,得不偿失,更不可能每挖掘一次就复制一个交易数据库副本。文中最后介绍的单项关联规则分析方法,需要在应用中检验。

参考文献:

- [1] 陈安,陈宁,周龙骧,等.数据挖掘技术及应用[M].北京:科学出版社,2006:50-75.
- [2] Han Jiawei, Kamber M. Data Mining Concepts and Techniques [M]. 北京:机械出版社,2005:149-161.
- [3] 陈伟. Apriori 算法的优化方法[J]. 计算机技术与发展, 2009, 19(6): 80-83.
- [4] 袁万莲,郑诚,翟明清. 一种改进的 Apriori 算法[J]. 计算机技术与发展, 2008, 18(5): 51-53.
- [5] 朱孝宇,王理冬,汪光阳. 一种改进的 Apriori 挖掘关联规则算法[J]. 计算机技术与发展, 2006, 16(12): 89-90.
- [6] 朱其祥,徐勇,张林. 基于改进 Apriori 算法的关联规则挖掘研究[J]. 计算机技术与发展, 2006, 16(7): 102-104.
- [7] 胡吉明,鲜学丰. 挖掘关联规则中 Apriori 算法的研究与改进[J]. 计算机技术与发展, 2006, 16(4): 99-101.
- [8] 程玉胜. Apriori 算法中频繁项集挖掘实现研究[J]. 计算机技术与发展, 2006, 16(3): 58-60.
- [9] 陈文庆,许棠. 关联规则挖掘 Apriori 算法的改进与实现[J]. 微机发展(现更名:计算机技术与发展), 2005, 15(8): 155-157.
- [10] Agrawal R, Srikant R. Fast Algorithms for Mining Association Rules in Large Database[C]//Proceeding of the 20th International Conference on Very Large Databases. Santiago, Chile: [s. n.], 1994: 487-499.
- [11] Agrawal R, Imielinski T, Swami A. Mining association rules between sets of items in large Databases[C]//Proceedings of the ACM SIGMOD Conference on Management of data (ACM SIGMOD'93). Washington, USA: [s. n.], 1993: 207-216.
- [12] 刘北林,刘莉. 商场超市仓储管理[M]. 北京:化学工业出版社,2008:106-108.
- [13] 姚昆遗,邹炜. 超市经营管理实务[M]. 沈阳:辽宁科学技术出版社,2004:63-70.

(上接第 104 页)

- [11] Joseph S. An Extendible Open Source P2P Simulator[J]. P2P Journal, 2003(12): 105-123.
- [12] Merugu S, Srinivasan S, Zegura E. P-sim: A Simulator for Peer-to-Peer Networks[C]//In proceedings of the 11th IEEE international Symposium on Modeling, Analysis, Simulation of Computer and Telecommunications Systems (MAS-COTS'03). [s. l.]: [s. n.], 2003: 213-218.